NetIQ® AppManager®

Installation Guide

September 2017

Legal Notice

For information about NetIQ legal notices, disclaimers, warranties, export and other use restrictions, U.S. Government restricted rights, patent policy, and FIPS compliance, see https://www.netiq.com/company/legal/.

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

Intended Audience

This guide provides information to ensure a successful installation of AppManager components. This guide is intended for system administrators and users responsible for installing all or part of the AppManager Suite software.

Other Information in the Library

The library provides the following information resources:

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.

Upgrade and Migration Guide

Provides information about upgrading from a previous version of AppManager.

Module management guides

Provide information about installing and monitoring specific applications with AppManager.

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
Website:	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
Website:	www.netiq.com/support

Contacting Documentation Support

Our goal is to provide documentation that meets your needs. The documentation for this product is available on the NetIQ website in HTML and PDF formats on a page that does not require you to log in. If you have suggestions for documentation improvements, click **comment on this topic** 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

NetIQ Communities, 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, NetIQ Communities 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.

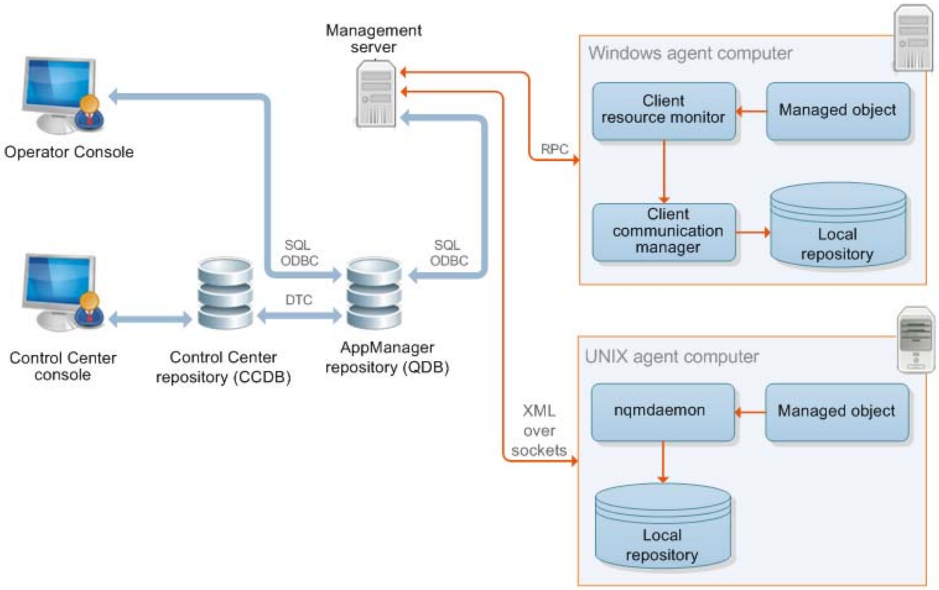
1 Introduction to AppManager

AppManager is a client/server application that helps monitor and manage a broad spectrum of IT environments. Before you install AppManager, it is important to understand the crucial components of the AppManager architecture. Understanding how AppManager works helps you develop a workable implementation plan and ensures successful deployment.

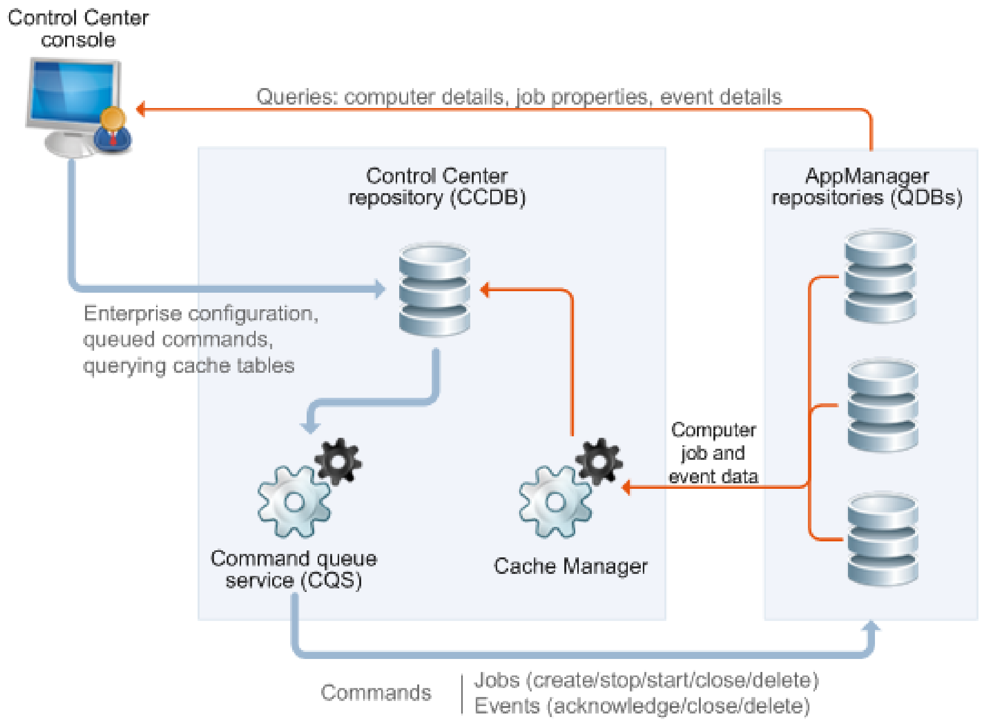
Understanding the AppManager Architecture

AppManager uses a scalable, flexible, tiered architecture that allows components to communicate efficiently and allows you to distribute process load across multiple components.

The following graphic illustrates the overall AppManager architecture and how components interact, including both the Operator Console and the Control Center console.



The following graphic illustrates the architecture of the Control Center console.


Control Center console
Queries: computer details, job properties, event details
Control Center repository (CCDB)
AppManager repositories (QDBs)
Enterprise configuration, queued commands, querying cache tables
Computer job and event data
Command queue service (CQS)
Cache Manager
Commands
Jobs (create/stop/start/close/delete)
Events (acknowledge/close/delete)

For more information about the options for distributing AppManager components across multiple computers, see "Implementation Guidelines" on page 35 and the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

Understanding AppManager Components

The AppManager flexible, tiered architecture consists of required and optional components. You can install components on one computer or on multiple computers.

The following table describes the AppManager components.

Component	Description	Required/Optional
AppManager repository (QDB)	SQL Server database that stores management information, such as jobs, events, data, and Knowledge Scripts	Required
Management server	Windows service called the NetIQ AppManager Management Service (`NetIQms`) that manages event-driven communication between AppManager agents and the QDB	Required
Task Scheduler service	Windows service that schedules SQL Server jobs for QDBs and the Control Center repository (CCDB)	Required

Component	Description	Required/Optional
Agent	AppManager software you deploy in your environment that schedules and runs jobs to manage third-party products and enables communication between AppManager components When you run the setup program to install the agent on 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 When you use Control Center to deploy agents to remote computers, the AppManager for Microsoft Windows module is not automatically deployed. You must also deploy the module to the computers where you deploy the agent. These components reside locally on the agent computer. For UNIX or Linux computers, the agent is a daemon and the supporting files and directories that provide data persistence (equivalent to the local repository) and access to system statistics (equivalent to modules). AppManager uses the NetIQ UNIX agent, which can be used for other NetIQ products. For more information about how to install the UNIX agent, see *AppManager for UNIX and Linux Servers Management Guide*, available on the AppManager Modules Documentation page (http://www.netiq.com/documentation/appmanager-modules).	Required
Report-enabled agent	Optional supplement to the agent that allows you to create and configure reports on selected computers in your environment You discover report-related elements on agent computers to enable different types of reporting. For more information about enabling the agent reporting capability, see Chapter 8, “Installing Agent Components,” on page 85.	Optional
Control Center repository (CCDB)	SQL Server database that stores information Control Center collects from the QDBs it manages, user preferences, security settings, and management group definitions	Required

Component	Description	Required/Optional
Control Center and Deployment services	Control Center components that include: ◆ Command queue service (CQS), a Windows service that retrieves commands from the CCDB and sends them to the appropriate QDBs ◆ Cache Manager, a child process of the command queue service running on each QDB that runs Control Center queries ◆ Deployment Service, which allows you to install agents and monitoring modules on remote computers If the service is across a firewall and you do not want to open additional ports to allow direct communication with the CCDB, you can configure it to use the Deployment Web Service for communication with the CCDB. During installation, choose the option that indicates a firewall is active between the Deployment Service and the CCDB. ◆ Deployment Web Service, which distributes deployment packages to the Deployment Service For Deployment Services that are across a firewall, the service can also provide a communication proxy to the CCDB so that you do not have to open additional ports to allow direct communication between the Deployment Service and the CCDB.	Required
NetIQ AppManager Integration Adapter	Allows NetIQ Aegis to communicate with AppManager through its repositories (QDBs and CCDBs), and includes Aegis workflow activities specific to AppManager You can install the NetIQ AppManager Integration Adapter (AppManager adapter) on any computer with network access to the NetIQ Resource Management Namespace Provider service and the repository with which you want NetIQ Aegis to communicate. For more information about installing the AppManager adapter, see the *NetIQ AppManager Integration Adapter Installation Guide*, available on the Aegis Documentation page (http://www.netiq.com/documentation/aegis).	Optional

Component	Description	Required/Optional
Control Center console	Windows interface that connects to the CCDB and allows you to run jobs on the systems and applications you manage across multiple QDBs The console provides a single user interface for managing most administrative functions and offers more powerful monitoring and deployment capabilities than the Operator Console. You can use the Control Center console to deploy agents and modules to remote computers. The Control Center console also provides access to the Chart Console, a Windows interface that allows you to generate and view charts of QDB data.	Required
Operator Console	Windows interface that allows you to view and control the jobs that monitor and manage your computers and server applications	Optional
UNIX Agent Manager console	Interface that allows you to use the UNIX Agent Manager to deploy and manage UNIX agents	Optional
Security Manager Console	Windows interface that allows AppManager administrators to control access to views and tasks in the Operator Console and manages application or computer-specific security information, such as SNMP community strings and passwords	Optional
Chart Console	Interface that allows you to generate and view charts of QDB data	Optional
Developer's Console	Tool for editing Knowledge Scripts and developing custom Knowledge Scripts	Optional

Monitoring in Different Environments

Computers on which you install AppManager agents become agent computers you can monitor. You run Knowledge Scripts to monitor agent computers. Knowledge Scripts help you collect data, monitor for events, and respond to events.

A job is an instance of a Knowledge Script running on an agent computer. Each time you run a Knowledge Script, you create a job. At a minimum, to create a job you must discover your agent computers and run Knowledge Scripts on those computers. For more information about discovering agent computers and running Knowledge Scripts, see the *Control Center User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

When you start a job, AppManager inserts a new record into the QDB and notifies the management server of the job request.

The agent communicates back to the management server any relevant output from the Knowledge Script. For network efficiency, the AppManager agent only communicates back to the management server when an event has occurred or data needs to be inserted into the repository database.

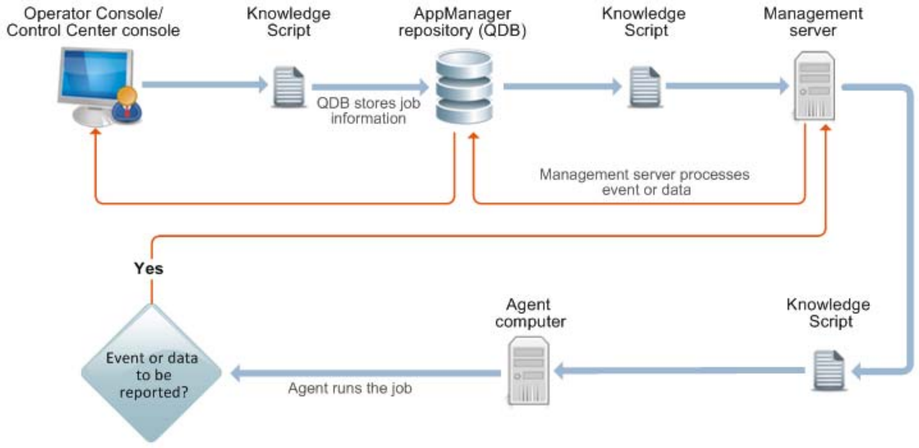
AppManager agents handle the scheduling and housekeeping of Knowledge Scripts, and initiate corrective actions and communication with the management server. The collection of performance and event data is facilitated through the use of software modules called **managed objects** that “plug into” the AppManager agent.

Knowledge Scripts use managed objects to access counters, event logs, queries, application programming interfaces (APIs), and other sources to gather statistics, metrics, and other properties of specific application elements. On Windows computers, managed objects are COM/OLE objects in the form of dynamic link libraries (`.dll` files). On UNIX and Linux computers, managed objects are Perl modules, in the form of dynamic shared libraries.

Using these native sources of information, managed objects collect raw statistics and information, such as current CPU utilization or database lock activity, and pass that information to the Knowledge Script jobs. Knowledge Scripts then provide the rules for what to do with this raw information. The Knowledge Scripts run under the control of the AppManager agent. On Windows agent computers, the Knowledge Scripts invoke the managed objects through the standard COM/OLE interface. On UNIX and Linux agent computers, the Knowledge Scripts invoke the managed objects through the standard Perl module interface.

Monitoring in a Windows Environment

When you start a job on a Windows computer, the Control Center console notifies the CCDB that you have requested a Knowledge Script to run (the Operator Console contacts the QDB directly). The command queue service (CQS) updates the appropriate QDB with information about the job properties and the QDB, with updated with job information, communicates with the management server (`NetIQms`). The management server then sends the Knowledge Script to the appropriate agent computers you want to monitor by contacting the AppManager agent (`NetIQmc`). The following diagram illustrates this process.



As the agent runs a job, it uses the associated Knowledge Script to gather information. Knowledge Scripts gather information a variety of ways. For example, a Knowledge Script might check the value of performance counters, read log files, execute queries, or access system tables.

In Windows environments, modules allow Knowledge Script jobs to run and gather information. A module is AppManager software that resides on the agent computer to enable management of a particular third-party product. During setup, you select modules to install based on the servers and applications you want to monitor. For more information about modules, see “Installing Modules” on page 90.

Each time a Knowledge Script runs, it evaluates information the module returns to determine whether the management server needs to insert events or data into the QDB. If so, the `NetIQmc` service notifies the `NetIQccm` service, which then notifies the management server to upload the information to the QDB. This triggers the CCDB to update with the latest information as well.

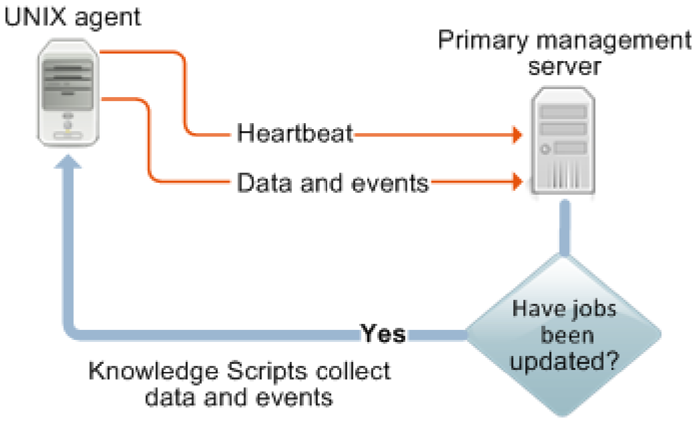
If the `NetIQccm` service cannot communicate with the management server, it writes the data to the local repository. Upon reconnection, the `NetIQccm` service uploads data from the local repository to the management server.

Monitoring in a UNIX or Linux Environment

If you are monitoring UNIX or Linux servers, the AppManager agents you install are called NetIQ UNIX agents (UNIX agents). You can use the UNIX agent with several NetIQ products, and you install the UNIX agent using NetIQ UNIX Agent Manager.

Every 30 seconds, UNIX agents send a heartbeat message to the management server to indicate they are working properly. Each heartbeat message also requests new or updated job information.

When the UNIX agent contacts the management server, the management server determines whether any of the Knowledge Script jobs for the agent computer have been added or updated. If you changed job properties or added new jobs since the last heartbeat interval, the management server delivers the revised job information to the UNIX agent. If there is no change to the Knowledge Script job the agent computer is running, the management server simply acknowledges the heartbeat and waits for the next heartbeat. The following diagram illustrates this communication flow.

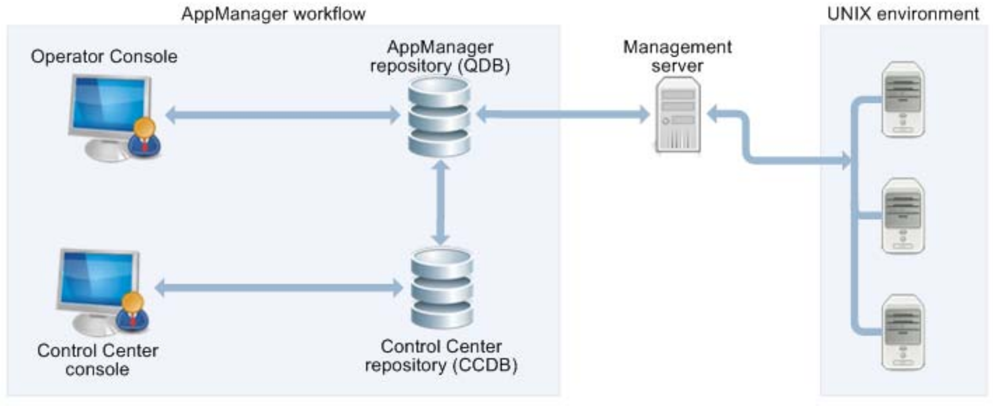

UNIX agent
Primary management server
Heartbeat
Data and events
Have jobs been updated?
Yes
Knowledge Scripts collect data and events

After it receives a job from the management server, the UNIX agent runs the job to access log files, system tables, or other data providers and retrieves the information requested.

Each time the Knowledge Script job runs, it determines whether events or data need to be inserted into the QDB. If an event condition is detected or a data point collected, the UNIX agent communicates with the management server to upload the information to the QDB.

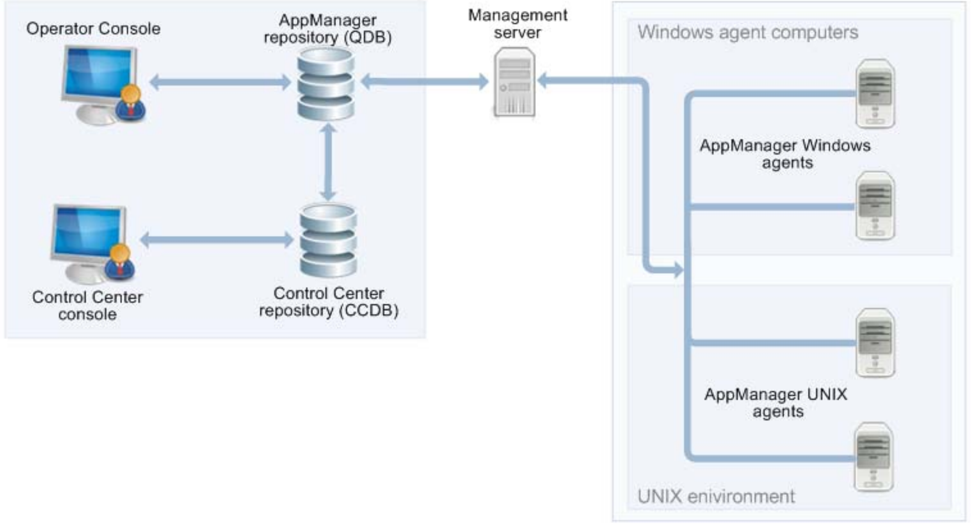
If the UNIX agent service cannot communicate with the management server, the agent writes the data to the `db` directory on the UNIX or Linux computer. When connectivity is reestablished, the UNIX agent uploads any data stored locally to the management server.

The management server inserts events and data from the UNIX agent into the standard AppManager workflow. You can use the Control Center console or the Operator Console to see events stored in the QDB. The following diagram illustrates this communication flow.



Working with Both Windows and UNIX Computers

Although slight differences in communication exist for Windows agents and UNIX agents, the AppManager workflow is the same in a heterogeneous monitoring environment. The following figure illustrates the basic relationship between AppManager components and the UNIX and Linux environment.



In an environment with both Windows computers and UNIX or Linux computers, a single management server can communicate with:

- Multiple Windows agents
- Multiple UNIX agents
- A combination of Windows and UNIX agents

You can also install multiple management servers in your environment to distribute processing and to provide failover support for Windows, UNIX, and Linux computers.

For information about:

- Configuring a management site to use multiple management servers, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).
- Monitoring in a UNIX or Linux environment, see the *AppManager for UNIX and Linux Servers Management Guide*, available on the AppManager Modules Documentation page (http://www.netiq.com/documentation/appmanager-modules).
- Installing and configuring a UNIX agent, see the *AppManager for UNIX and Linux Servers Management Guide*, available on the AppManager Modules Documentation page (http://www.netiq.com/documentation/appmanager-modules).

2 Planning to Install AppManager

This chapter helps you plan your AppManager installation.

Getting Started

You can use the AppManager setup program to perform the following types of installations:

- **Evaluation:** A stand-alone configuration with all AppManager components, with the possible exception of the QDB and CCDB, on one computer. An Evaluation installation includes full functionality and expires 30 days from the installation date. For information about performing an Evaluation installation, see the *Trial Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).
- **Production:** A full-fledged AppManager deployment. The information in this chapter helps you plan a Production installation.

Planning Checklist

Use the following checklist to plan your AppManager installation and record information you will need during the installation.

	Step	Reference
❑	Determine the accounts you will need to install components and ensure the accounts have the required permissions. ◆ AppManager installation account to run the setup program. ◆ QDB creation account for the setup program to log in to the SQL Server to create the QDB. ◆ QDB owner account to serve as `db_owner` of the QDB. ◆ Management server service (`NetIQms`) account. ◆ Management server QDB connection account for the management server to connect to the QDB. ◆ Task Scheduler service account. ◆ Agent services (`NetIQmc` and `NetIQccm`) account. ◆ CCDB creation account for the setup program to use to log in to the SQL Server to create the CCDB. ◆ CCDB owner account to serve as `db_owner` of the CCDB. ◆ Command queue service account. ◆ Deployment Service account. ◆ Deployment Web Service account.	"Reviewing Required Accounts and Permissions" on page 24
❑	Determine where to install AppManager components. Ensure components can connect to and communicate with each other.	◆ "Understanding Network Connection Requirements" on page 27 ◆ "Understanding Firewall Considerations for Control Center Components" on page 28 ◆ "Reviewing AppManager Port Usage" on page 28
❑	Estimate the initial QDB size. Initial size estimate:	"Sizing the QDB" on page 30

	Step	Reference
❏	Assess your security needs. ◆ SQL Server security mode: ◆ AppManager roles for the Operator Console: ◆ AppManager user groups and permission sets for the Control Center console: ◆ Security level for communication between the management server and agent computers: ◆ Security information needed for Knowledge Scripts:	"Assessing Your Security Requirements" on page 32
❏	Evaluate the environment you want to monitor and review implementation scenarios and recommendations. Consider the following factors that influence the number of QDBs and management servers you will need to accomplish your monitoring objectives: ◆ Number of Windows computers to monitor: ◆ Number of UNIX computers to monitor: ◆ Number and type of application and database servers to monitor: ◆ Hardware types to monitor: ◆ Specific components to monitor—for example, operating systems, email servers, clustered applications, and so on:	"Implementation Guidelines" on page 35
❏	Develop a deployment plan.	*Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager)
❏	Ensure that the computers where you plan to install components meet system requirements.	Chapter 4, "System Requirements," on page 45

Reviewing Required Accounts and Permissions

The AppManager setup program requires access to various user accounts during installation. The following table lists the required accounts.

Account	Requirements	Other Considerations
AppManager installation	◆ Windows user account ◆ Member of the local Administrators group	None
QDB creation	◆ Windows or SQL Server user account During installation, if you select Windows authentication, the setup program uses the Windows account under which you are currently logged in to access the SQL Server to create the QDB. If you select SQL Server authentication, the setup program uses the SQL Server user account you specify to access the SQL Server. ◆ `sysadmin` SQL Server role	Before installation, use Microsoft SQL Server Management Studio or SQL Server Enterprise Manager to verify the account or the group to which the account belongs has the `sysadmin` role. Before SQL Server 2008, Microsoft provided a `BUILTIN\Administrators` account to automatically grant the `sysadmin` role to any Windows account in the local Administrators group. SQL Server 2008 does not include this account by default, and it is possible to manually remove the account in earlier versions of SQL Server.
QDB owner	◆ Windows or SQL Server user account ◆ Meets password requirements on the SQL Server that will host the QDB ◆ Does not already exist in the SQL Server If the account already exists, it might create a conflict unless the database administrator sets the login properties for the account to use a fully-qualified account name (domain\username).	◆ The account will be assigned the `sysadmin` SQL Server role. ◆ If you use a SQL Server user account, specify the same account for the management server to use to connect to the QDB. ◆ If you specify a user name that includes certain special characters, you will not be able to log in to the Operator Console using the QDB owner account. The special characters are: \ / * ? : < > | “
Management server service (`NetIQms`)	◆ Windows local system or Windows user account ◆ Windows user account must be a member of the local Administrators group, have the right to log on as a service, and have the right to log in to the SQL Server that hosts the QDB	◆ If the service will use the Windows local system account, the management server must use a SQL Server user account to connect to the QDB. ◆ If you use Windows Authentication security mode for Microsoft SQL Server, the service must use a Windows user account.
Management server QDB connection	Windows or SQL Server user account	◆ If you use a Windows user account, the management server uses the `NetIQms` account to connect to the QDB. ◆ If you use a SQL Server user account as the QDB owner account, specify the same account for the management server to connect to the QDB.

Account	Requirements	Other Considerations
Task Scheduler service	- Windows local system or Windows user account with the right to log on as a service - Windows user account must be a member of the local Administrators group - Accounts must have access to the SQL Server instances that host the repositories the service will manage	Specify a Windows user account if the SQL Server that will host the repositories the service will manage uses Windows authentication. If you want to use the local system account, the repositories must use SQL authentication. When you add repositories to the service, if you select to use Windows authentication for the service to connect to the repositories, the service will use this account to make the connection.
Agent services (`NetIQmc` and `NetIQccm`)	Windows local system or Windows user account with the right to log on as a service When the agent is on the same computer as the management server, the account requires administrator permissions on the management server.	In the following situations, specify a Windows user account: - You will install the agent on the management server. - The agent will monitor a SQL Server that uses Windows authentication. - You plan to install a module, such as AppManager for Microsoft Exchange Server or AppManager for Microsoft Active Directory, that requires the agent services to run under a Windows user account. For information about the permissions and memberships a module requires, see the management guide for the module. If you are unsure whether a module requires a Windows user account for the agent services, NetIQ Corporation recommends installing the agent using the Windows local system account. If necessary, you can use the Services application in Control Panel to change the account after installation. If you change the account, change it for both agent services. - You plan to enable the agent to generate reports. For more information about the agent reporting capability, see "Understanding Agent Reporting Capabilities" on page 86. - You plan to enable the MAPI mail option. For more information about the MAPI mail option, see "Understanding MAPI Mail Settings" on page 87.

Account	Requirements	Other Considerations
CCDB creation	◆ Windows or SQL Server user account During installation, if you select Windows authentication, the setup program uses the Windows account under which you are currently logged in to access the SQL Server to create the CCDB. If you select SQL Server authentication, the setup program uses the SQL Server user account you specify to access the SQL Server. ◆ `sysadmin` SQL Server role	Before installation, use Microsoft SQL Server Management Studio or SQL Server Enterprise Manager to verify the account or the group to which the account belongs has the `sysadmin` role. Before SQL Server 2008, Microsoft provided a `BUILTIN\Administrators` account to automatically grant the `sysadmin` role to any Windows account in the local Administrators group. SQL Server 2008 does not include this account by default, and it is possible to manually remove the account in earlier versions of SQL Server.
CCDB owner	◆ Windows or SQL Server user account ◆ Meets password requirements on the SQL Server that will host the QDB ◆ Does not already exist in the SQL Server If the account already exists, it might create a conflict unless the database administrator sets the login properties for the account to use a fully-qualified account name (domain\username).	The account will be assigned the `sysadmin` SQL Server role.
Command queue service	◆ Windows user account ◆ Member of the local Administrators group An account with Domain Administrator privileges is not sufficient unless it is also a direct member of the local Administrators group. ◆ Has the right to log on as a service ◆ Has the right to log in to the SQL Server that hosts the CCDB	◆ The account will be granted permissions in each managed QDB. ◆ The command queue service, Deployment Service, and Deployment Web Service can use the same account. ◆ The command queue service will run under this account and use it to connect to the CCDB. ◆ When you install the service, the account under which you run the installation program must have administrative privileges on the SQL Server that hosts the CCDB. Otherwise, the installation program will not be able to establish a connection with the CCDB and the installation will fail.

Account	Requirements	Other Considerations
Deployment Service	◆ Windows user account ◆ Member of the local Administrators group An account with Domain Administrator privileges is not sufficient unless it is also a direct member of the local Administrators group. ◆ Has the right to log on as a service ◆ Has the right to log in to the SQL Server that hosts the CCDB ◆ If using special characters ($%*&) or reserved characters (period or comma) in the password, enclose the password in quotation marks. For example, "P@$$word123". Otherwise, the service will not start.	◆ The account will be granted appropriate permissions in each managed QDB. ◆ If a firewall is present between the Deployment Service and the CCDB, the Deployment Service uses the Deployment Web Service account. For more information about how the Deployment Service connects to the CCDB, see "Understanding Deployment Server Configuration" on page 73. ◆ The command queue service, Deployment Service, and Deployment Web Service can use the same account. ◆ The Deployment Service will use this account only to connect to the CCDB. It will not run under this account. The service will run under the local system account. ◆ When you install the service, the account under which you run the installation program must have administrative privileges on the SQL Server that hosts the CCDB. Otherwise, the installation program will not be able to establish a connection with the CCDB and the installation will fail.
Deployment Web Service	◆ Windows user account ◆ Member of the local Administrators group An account with Domain Administrator privileges is not sufficient unless it is also a direct member of the local Administrators group. ◆ Has the right to log on as a service ◆ Has the right to log in to the SQL Server that hosts the CCDB	◆ The command queue service, Deployment Service, and Deployment Web Service can use the same account. ◆ The Deployment Web Service will use this account only to connect to the CCDB. It will not run under this account. ◆ When you install the service, the account under which you run the installation program must have administrative privileges on the SQL Server that hosts the CCDB. Otherwise, the installation program will not be able to establish a connection with the CCDB and the installation will fail.

Understanding Network Connection Requirements

Consider your network configuration when determining where to install AppManager components. Typically, a basic AppManager setup requires the following network connections:

- TCP/IP and ODBC connectivity between the QDB and management server on separate computers
- TCP/IP and RPC connectivity between the management server and each agent computer

  The management server must resolve the names and IP addresses of agent computers, and agent computers must resolve the name and IP address of the management server.
- ODBC connectivity between report-enabled agent computers and the QDB
- TCP/IP and ODBC connectivity between the QDB and the Task Scheduler service

- TCP/IP and ODBC connectivity between the CCDB and the Task Scheduler service
- Microsoft Distributed Transaction Coordinator (DTC) connectivity between the CCDB and each managed QDB

  You can check DTC connectivity before you install Control Center components. For more information about checking connectivity, see "Using the Control Center Configuration Checker Utility" on page 79.
- ODBC connectivity between the Control Center console and the CCDB
- TCP/IP and ODBC connectivity between the Operator Console and the QDB

Understanding Firewall Considerations for Control Center Components

NetIQ Corporation recommends distributing Control Center components across computers to improve performance. Because the command queue service runs under a Windows user account and connects to QDBs and the CCDB using that account, ensure that no firewall is present between the command queue service and QDB computers and the command queue service and CCDB computers. On computers running Microsoft Windows Server 2008 or later, a firewall is enabled by default. At a minimum, you will need to open ports 1433 and 135. For information about other SQL Server ports you might need to open, see "Reviewing AppManager Port Usage" on page 28.

Reviewing AppManager Port Usage

AppManager components communicate with each other through default ports. Check for port restrictions specific to your site or firewall protections that might prevent you from using certain ports.

The following table lists the default ports that AppManager uses.

Components Communicating	Ports Used	Protocols	Service/process using port
Connections to QDB:			
Task Scheduler service to QDB	1433† 1433 is the default port. You can use any port.	TCP/IP	SQL
Management server to QDB	1433† 1433 is the default port. You can use any port.	ODBC	SQL
Control Center command queue service to QDB	1433† 1433 is the default port. You can use any port.	ODBC TCP/IP	SQL
Operator Console to QDB	1433† 1433 is the default port. You can use any port.	ODBC	SQL

Components Communicating	Ports Used	Protocols	Service/process using port
Connections to CCDB:			
Control Center command queue service to CCDB	1433† 1433 is the default port. You can use any port.	TCP/IP	SQL
Control Center console to CCDB	1433† 1433 is the default port. You can use any port.	TCP/IP	SQL
Connections required for deployments:			
Deployment Service to Windows agent computers for installation	135, 139‡ If 139 is not available and a firewall is not present, you can use port 445.	TCP/IP	
Deployment Service to CCDB Deployment Web Service to CCDB Proxy Deployment Web Service to CCDB	1433† 1433 is the default port. You can use any port.	TCP/IP	SQL
Deployment Service to Deployment Web Service	80*	HTTP	Background Intelligent Transfer Service (BITS)
Deployment Service running in proxy mode to proxy Deployment Web Service	443*	HTTPS	BITS
Connections between management servers and agents:			
Management server to Windows agents	9998 9998 is the default port. You can use any port.	TCP/IP	
Windows agents to management server	9999 9999 is the default port. You can use any port, but ensure that you also update the port information on the management server. For more information, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).	TCP/IP	

Components Communicating	Ports Used	Protocols	Service/process using port
UNIX agents to management server	9001 9001 is the default port. You can use any port, but ensure that you also update the port information on the management server. For more information, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).	TCP/IP	
Connections between UNIX Agent Manager and UNIX agents:			
UNIX Agent Manager to UNIX agents	2620 2620 is the default port. You can use any port.	TCP/IP	
UNIX Agent Manager console to UNIX Agent Manager	2222 2222 is the default port. You can use any port.	TCP/IP	
Miscellaneous connections			
Troubleshooter utility on Operator Console or Control Center console computer to agent	8996*	TCP/IP	

* Indicates a bidirectional port requirement

† Indicates additional port requirements

If you use a named SQL Server instance, or if you are not using the default SQL Server port (1433), additional port requirements include:

- The SQL Server browser port, UDP 1434. The SQL Server Browser service helps clients determine the associated SQL Server port to use. Once a client establishes a connection to the SQL Server running on the non-default port, it will not use the SQL Browser again unless the SQL Server port changes.
- The SQL Server port for the instance that is hosting the QDB and CCDB.
- If the SQL Server is clustered, include the physical address for each cluster node and the virtual IP address for each repository.

‡ Indicates that you need an additional port range. If you plan to remotely install agents and updates across a firewall, decide how many ports you want to allocate to DCOM processes on the agent computers.

Sizing the QDB

Consider the following factors when planning your QDB configuration:

- The number of events you expect to generate
- The number of data points you intend to collect and save for historical reporting or trend analysis

Because this information is difficult to estimate before you install AppManager, and changes as you expand and refine your deployment strategy, NetIQ Corporation recommends the following process as a starting point to size the QDB data and log files.

To estimate the initial size of the QDB data and log files:

1. Determine the number of agent computers you plan to monitor and multiply that number by 1/3 MB to account for the events and data each will generate, then multiply the result by the number of days you intend to keep data in the QDB. For example:

   Number of agent computers = 180 X 1/3 MB = 60 MB

   Number of days to retain data = 30

   Estimated QDB data file size = 60 MB X 30 = 1800 MB (1.8 GB)
2. Set the initial log file size to 512 MB, which is the default installation value.

Sizing the initial data and log files along these guidelines is a good starting point in most environments. It is roughly one-third of the size for a full deployment. Keep the data file and the log file on separate devices.

For more information about determining the number of QDBs for your environment, see "Implementation Guidelines" on page 35.

Accounting for Database Growth

The default size for the QDB data file is 2048 MB. This size is adequate for a small network with moderate monitoring activities. Larger AppManager deployments will need additional space.

Setting the initial size to about a third of what you think you will need avoids reserving space you will not use during the early stages of deployment, when you are unlikely to run a full set of Knowledge Script jobs or collect all the data you will eventually want to use. Instead, you will probably increase the number of agent computers and the number of jobs you run over time. In addition, maintenance operations, such as backup and restore, are easier if you create smaller data files and plan for growth rather than sizing the data file at the onset to handle your eventual database requirements.

Although SQL Server can dynamically increase the size of database files and memory, this is not a reliable method for managing database growth. The default setting for the size increase is only 1 MB, which can lead to multiple size increases and corresponding disk fragmentation. Letting the database grow dynamically can also cause fragmentation of the SQL space, which can severely impede performance. Instead, plan for periodic QDB maintenance. Plan to monitor the size of the QDB by running AppManager Knowledge Script jobs to check the size at regular intervals.

To estimate the potential growth of your QDB, assume it will grow at a rate of about 2 MB per server, per day, assuming 20 to 30 jobs running at 15 minute intervals and collecting data. NetIQ Corporation recommends installing NetIQ Analysis Center to manage and report on your data if you need to keep data for longer than 90 days. Typically, AppManager performance will start to deteriorate as the QDB surpasses 50 GB in size. If QDB size exceeds 100 GB, its impact on performance might be severe.

AppManager provides many options for managing the QDB and keeping it healthy. For example, you can configure the QDB to consolidate older data into daily, weekly, and monthly averages. For more information about managing QDBs, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

To achieve optimal SQL Server and AppManager performance, place the data and transaction logs on separate physical drives. You can select the locations of these logs during QDB installation.

Adjusting the Size of Other Databases

As you increase the amount of data you store, you might also need to increase the size of the `temp` database to handle queries that require temporary space. By default, the database is 8 MB. The more data you store and access, and the more you plan to use AppManager reporting capabilities, the more space you should set aside for the `temp` database.

You typically do not need to change the size of the other databases, such as the master database.

Assessing Your Security Requirements

NetIQ Corporation recommends using Windows groups to provide secure access to AppManager data and components. This approach entails using Windows administrative tools to create and manage user and group accounts and mapping those groups and users to SQL Server login accounts. You can then use the Control Center console or Security Manager (which defines security permissions for the Operator Console) to define who has access to particular repositories and what AppManager functions and components they are able to use.

To assess your security requirements before installing AppManager, determine the following information:

- SQL Server security mode

  NetIQ Corporation recommends using Windows authentication on the SQL Server that hosts the QDB. Using Windows authentication simplifies several tasks when setting up permissions for users or groups to access AppManager components and features. If the SQL Server uses mixed mode authentication, users can log in to the QDB using either Windows authentication or SQL Server authentication. Then, you must communicate with users about which login account they should use to access the QDB and also configure access permissions on two separate accounts for each user or group.

  For more information about the relationship between SQL Server security and AppManager, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager). For more information about using security modes, see the Microsoft SQL Server documentation.
- For the Operator Console, AppManager user roles and the AppManager-related rights to associate with each role

You use Security Manager to configure security for a QDB and control access to views and tasks in the Operator Console through AppManager roles. Every Operator Console user must be assigned a role. To help you get started, the Operator Console provides the following predefined roles:

Role	Default Rights
Administrator	All functional rights to perform all Operator Console activities and see all views. You can copy this role, but you cannot modify, delete, or rename it. Because you cannot modify this role, only the **Users** tab is available in the Properties pane when you select the Administrator role. Users with the Administrator role in AppManager must have the Microsoft SQL Server `db_owner` role for the QDB.
Read-Only User	Functional rights to start the Operator Console or Control Center console and see all views but not perform any AppManager activities. You can copy, modify, delete, or rename this role. Users with the Read-Only User role in AppManager must have the Microsoft SQL Server `public` role for the QDB.
Standard User	Functional rights to perform all basic Operator Console and Chart Console activities and to see the **Master** view. You can copy, modify, delete, or rename this role. Users with the Standard User role in AppManager must have the Microsoft SQL Server `public` role for the QDB.

Most organizations find it useful to modify the predefined roles or create custom roles before adding any Operator Console users. For more information about using Security Manager to manage Operator Console security, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

- For the Control Center console, AppManager user groups and permission sets

  You use the Control Center console to manage security for the CCDB. To configure security permissions in Control Center, you must add users and user groups, define permission sets, and then assign the user groups and permission sets to management groups.

  The Control Center console includes a set of default user groups you can use, modify, copy, or delete to help implement security for the console. The default user groups include the following:

  - Administrator
  - Executives and Stakeholders
  - NOC Tier 1
  - NOC Tier 2
  - Trusted Application Admins
  - Trusted Application Owners

  You cannot copy or delete the Administrator group.

  A permission set is a collection of operational and Knowledge Script permissions that defines a group of activities that can be performed and Knowledge Scripts that can be used in the Control Center console. To apply a permission set, you associate the permission set with a user group and a management group. Users belonging to that user group can perform the activities you define in the permission set for the associated management group.

The default permission sets are:

- AppManager Administrator
- Deny Management Group Access
- Event Operation
- Management Group Administration
- Monitoring Administration
- Monitoring Operation
- Read Only

For information about the specific operational permissions granted for each default permission set, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

A global permission set is a permission set associated with a user group that applies to all management groups managed by the Control Center console for the associated user group. Since global permissions apply to all management groups for the associated user group, they do not depend on association with a specific management group to take effect. The Control Center console provides a default set of global permissions:

User Group Name	Permission Set Name
Executives & Stakeholders	Read Only
NOC Tier 1	Event Operation
NOC Tier 2	Monitoring Operation
Trusted Application Admins	Monitoring Administration
Trusted Application Owners	Management Group Administration

For more information about managing Control Center security and the interaction between Control Center and Operator Console security, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

- Level of security appropriate for the management server and agent computers

  Agent installation offers extra security options to encrypt agent-to-management server communications, or to encrypt communications and require agents to authenticate the management server. In most cases, you do not need to use these extra options, which add some overhead to production servers and the management server.

  AppManager always encrypts passwords, so even without extra agent security options, only user names are sent as clear text over the network. If you require a password for access to a particular application, like SQL, the password is encrypted in a table. That encrypted password is sent to the agent, which records it locally, still encrypted. Only when a job executes will the password be unencrypted and used to gain access to the application.

  To secure communication between the management server and agent computers, choose either **Encrypted communications only** or **Authentication and encrypted communications** when you install the QDB and agents. For more information about the secure communication options, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

If you choose **Encrypted communications only** or **Authentication and encrypted communications** when you install the QDB and agents, AppManager implements FIPS-compliant algorithms. FIPS compliance does not affect unencrypted communications. For more information about AppManager FIPS compliance, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

Although you manage secure communication separately for Windows agents and UNIX agents, all management servers and agent computers in a management site should use the same level of security. For either platform, you cannot mix security levels. For example, you cannot set some Windows agent computers to use clear text or encryption while other Windows agent computers use authentication and encryption.

- Security-related information needed to run Knowledge Scripts (for example, community names, user account, or password information)

  For information about security requirements for running Knowledge Scripts, see the management guide for the applicable module. Typically, you need to enter this information when you install the module for an application that requires it. For example, AppManager for Microsoft Exchange Server requires a user account, profile, and mailbox alias name.

Implementation Guidelines

Because you can deploy AppManager in almost any scenario, there is no standard implementation formula that is applicable to all scenarios.

A management site comprises one QDB and one or more management servers. Typically, you install the QDB and management server on separate computers. The management server accesses the QDB every five seconds and manipulates QDB data, which results in dense network traffic and requires a highly-available connection between the management server and the QDB.

NetIQ Corporation recommends installing multiple management servers to distribute processing and communication and provide failover support for agent computers. Each agent computer needs at least one management server. If you install multiple management servers, explicitly designate a primary and secondary management server for each agent computer to communicate with. For more information about designating primary and secondary management servers, see Chapter 8, “Installing Agent Components,” on page 85 and the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

Workload is an important factor to consider when planning your implementation. The number of agent computers you plan to have in your environment and the number of jobs you plan to run on each agent computer should help you determine the number of management servers and QDBs you need.

NetIQ Corporation recommends always having two management servers per QDB to provide failover support. Designate one as primary and one as secondary and split the workload so that half of your agent computers use management server A as primary and management server B as secondary, and the other half use management server B as primary and management server A as secondary.

3 Staging the Deployment

This chapter describes typical deployment stages and goals for each stage.

Installing in a Lab Environment

NetIQ Corporation recommends initially installing AppManager in a lab environment. When you install AppManager in a lab environment, focus on the following goals:

- Uncovering potential conflicts between AppManager and other applications, such as firewalls

  For example, you might have special port requirements or restrictive account policies. If you uncover problems, you can search the AppManager Knowledge Base on the NetIQ website for information about resolving the problem, or contact Technical Support.
- Quantifying the resource usage requirements of AppManager components

  This allows you to safely test your assumptions and verify that the computers where you intend to install components during the actual deployment meet the requirements.
- Documenting network utilization between components

  Even when deploying in a test environment, setting up a realistic sample of scripts and distribution of components lets you assess your bandwidth and latency assumptions.
- Testing the distribution of AppManager agents to ensure you have reliable account information and permissions (for example, usable passwords and domain account names)
- Estimating the time required to install components and resolve installation issues

Deploying to a Pilot Group

Depending on the size of your organization, the importance of your monitoring needs, the expertise of your deployment team, and the resources available to you, the pilot deployment might involve a small but representative number of computers or all of the computers you intend to monitor. NetIQ Corporation recommends installing on enough computers to get a realistic view of the full-scale deployment. The pilot deployment should last from two to four weeks and reveal the following information:

- Problems that need immediate attention, such as computers that are low on disk space
- Environmental issues you need to address, such as insufficient privileges or instability
- How closely the computers you want to monitor conform to your expectations

During the pilot deployment, focus on the following goals:

- Running the recommended core set of Knowledge Scripts on agent computers

  For more information about working with Knowledge Scripts and jobs, see the *Control Center User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager). For more information about the recommended core set of Knowledge Scripts, see “Running the Recommended Core Knowledge Scripts” on page 38.

- Identifying and correcting problems with running the core set of jobs

  For example, you might find problems with the required accounts and permissions.
- Gaining experience viewing and responding to events

  For more information about how AppManager raises events and using the Control Center console to view and respond to them, see the *Control Center User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).
- Identifying normal operating values and adjusting thresholds for your environment

  For more information about identifying normal operating values, see "Collecting Data" on page 39.
- Gathering operational data, such as disk space and CPU utilization, for charting and reporting

  For more information about using AppManager to generate charts and reports, see the *Control Center User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

Running the Recommended Core Knowledge Scripts

During the pilot deployment, NetIQ Corporation recommends that you only run a core set of Knowledge Scripts and restrict the number of users allowed to perform activities such as acknowledging and closing events or starting and stopping jobs. Running only a core set of Knowledge Scripts prevents a large number of events from overwhelming your staff and allows you to understand the events the jobs generate, develop a methodology for responding to them, and troubleshoot issues.

In a typical environment, you run approximately 20 Knowledge Script jobs on each agent computer at regular intervals to ensure basic operational health and availability. You run additional jobs less frequently to diagnose problems or take corrective action. Although running around 20 jobs is typical, the core set of Knowledge Scripts you initially run might include fewer jobs.

NetIQ Corporation recommends initially running a core set of Knowledge Scripts from the General and NT Knowledge Script categories. The following table describes the recommended core set of Knowledge Scripts. For more information about using these Knowledge Scripts and setting parameters, see the *AppManager Knowledge Script Reference Guide*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

Knowledge Script	Description
General_EventLog	Monitors and filters information in the Windows Event Log and allows you to track log entries that match filtering criteria Initially, NetIQ Corporation recommends monitoring all logs for error events. You can further filter the log entries to include or exclude other criteria such as specific IDs, descriptions, user names, or computer names.
General_MachineDown	Detects whether the computer on which you run the script can communicate with one or more specified Windows computers and raises an event if communication attempts fail
NT_MemUtil	Monitors physical and virtual memory and the paging files and raises an event if a monitored metric exceeds the threshold
NT_DiskSpace	Monitors logical drives for disk utilization, the amount of free space available, and the percentage of disk growth

Knowledge Script	Description
NT_CpuLoaded	Monitors total CPU usage and queue length to determine whether the CPU is overloaded and raises an event when both the total CPU usage and CPU queue length exceed the thresholds
NT_LogicalDiskStats	Monitors logical disk reads, writes, and transfers per second, disk operation time, and queue length
NT_PhysicalDiskStats	Monitors physical disk reads, writes, and transfers per second, disk operation time, and queue length
NT_ServiceDown	Monitors whether specified Microsoft Windows services are stopped or started, and, optionally, starts any stopped service
NT_TrustRelationship	Tests the domain trust relationship from the computer on which you run the script to a specified domain and raises an event if a problem exists with the domain trust

Collecting Data

To identify normal baseline operating values before you set thresholds for events, set all Knowledge Scripts only to collect data (that is, not to raise events) and run reports for at least one week. From the reports, you can review the high, low, and average values for core statistics. You can configure several basic report Knowledge Scripts to create reports.

To create reports about your environment:

1. Install at least one report-enabled agent.

   For more information about enabling reporting capability for an agent, see "Understanding Agent Reporting Capabilities" on page 86.
2. Run the Discovery_ReportAgent Knowledge Script on the report-enabled agent computer.
3. In the **Report** view, click through tabs in the **Knowledge Script** pane to select the reports to run.

At the end of the collection period, evaluate the information to determine a baseline for a normal operating environment. After you complete your evaluation, remove the data you collected from the QDB. For information about removing data from the QDB, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

When you are ready to raise events, set only those Knowledge Scripts that address critical issues in your environment to raise events, and set the remaining Knowledge Scripts to collect data. You can employ this approach enterprise-wide or only on the computers you identify as needing immediate attention. To help tune your system later, track the frequency of events and the number of data points collected.

Based on the data you collect, you can adjust thresholds to more accurately reflect specific characteristics of your environment. If you see too many events, the thresholds might be too low for your environment, the intervals might be too short, or you might need to address critical resource issues.

Basic AppManager reporting provides detailed information about the computers in a single management site. When you expand your deployment to multiple management sites with multiple QDBs, you might want the more sophisticated reporting available with NetIQ Analysis Center.

Expanding the Scope of Your Deployment

When you feel comfortable with the core set of Knowledge Scripts and the stability of your environment, consider expanding your deployment. During the expansion stage, focus on the following goals:

- Deploying AppManager to additional computers

  Large or widely distributed organizations typically phase in a full AppManager deployment over a period of several weeks or even months. For example, if your organization is going to monitor a group of computers in the United States, Germany, and Spain, you might decide to deploy AppManager first in Germany, stabilize the environment there, and then expand the deployment to include computers in Spain and the United States. Or you might decide to expand the deployment to include the computers in Spain, allow time to uncover problems and stabilize that environment, and deploy to the computers in the United States later.

- Running additional Knowledge Scripts beyond the core set

  For more information about additional recommended Knowledge Scripts, see "Running Additional Knowledge Scripts" on page 40.

- Identifying reporting needs and generating recommended reports

  For more information about reporting needs to consider and reports to generate, see "Identifying Your Reporting Requirements" on page 41.

- Adding responsive and corrective actions to Knowledge Scripts

  AppManager Knowledge Scripts can automatically take corrective actions, notify selected people in response to certain events, and acknowledge events. To take advantage of Knowledge Script automation capabilities, you might need to install additional components, such as an agent that can send email responses to events. For more information about enabling agents to send email responses to events, see "Understanding MAPI Mail Settings" on page 87. For more information about responsive and corrective actions, see the *Control Center User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

Running Additional Knowledge Scripts

During the expansion stage, add Knowledge Scripts beyond the core set. The following table describes Knowledge Scripts that NetIQ Corporation recommends adding. For more information about using these Knowledge Scripts and setting parameters, see the *AppManager Knowledge Script Reference Guide*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager)

Knowledge Script	Description
General_AsciiLog	Monitors one or more ASCII text files for specific strings and messages
General_Counter	Monitors any System Monitor counter
NT_NetworkBusy	Monitors the traffic on network interface cards (NICs) and raises an event if the bandwidth utilization of the network interface exceeds the threshold
NT_PagingHigh	Monitors reads and writes per second to the pagefile and raises an event if the number of reads and writes per second exceeds the threshold

Knowledge Script	Description
NT_PrinterHealth	Monitors printer health and raises an event if the printer is paused, the queue length exceeds the threshold, or there is some other error such as a jammed printer
NT_PrinterQueue	Monitors printer queue length and raises an event if the number of queued jobs exceeds the threshold
NT_RunAwayProcesses	Detects runaway processes on the specified computer based on sustained high CPU usage and raises an event if a process exceeds the CPU usage threshold
NT_SystemUpTime	Monitors the number of hours a computer has been operational since it was last rebooted and raises an event if the computer was rebooted within the monitoring interval

Once you select a set of Knowledge Scripts for monitoring basic server health and key application resources, you can plan for and implement policy-based monitoring. For information about implementing monitoring policies, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

Identifying Your Reporting Requirements

When considering your reporting needs, determine the following information:

- Standard AppManager reports to generate and the Knowledge Scripts required to generate those reports
- Who should receive the reports and how frequently
- Whether to generate reports automatically on a scheduled basis or manually on demand
- Who will generate reports

  For example, you might want to restrict access to the **Report** view or assign Exchange reports to an Exchange administrator and SQL Server reports to your DBA group.
- Whether to format reports in table format, in charts, or both
- Whether to deliver reports through e-mail or a website

The following table describes report Knowledge Scripts that NetIQ Corporation recommends running to generate standard reports. For more information about using these Knowledge Scripts and setting parameters, see the *AppManager Knowledge Script Reference Guide*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

Knowledge Script	Description
ReportAM_EventSummary	Summarizes events per computer
ReportAM_SystemUpTime	Details the uptime and downtime of monitored computers
ReportAM_CompDeploy	Details the number of instances of each AppManager component installed on computers in an AppManager site
ReportAM_WatchList	Details the top or bottom N computers (by number or percent) generating the selected data streams
NT_Report_CPULoadSummary	Summarizes CPU usage and queue length for selected computers

Knowledge Script	Description
NT_Report_LogicalDiskUsageSummary	Summarizes the percentage of disk space used and the amount of free space (in MB) for selected computers

Reviewing and Refining the Deployment

Once basic monitoring is underway, it becomes easier to fine-tune thresholds and job intervals, articulate and automate event-response policies, and tailor event notification, data collection, and the user interface to suit your needs. As you refine your deployment, focus on the following goals:

- Managing events and event notification

  For more information about developing and refining your policies for handling events, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

- Handling data-collection and archiving data

  For more information about developing and refining your data handling policies, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

- Controlling communication between agent computers and management servers

  For more information about communication between agent computers and management servers, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

- Managing security and security roles within AppManager

  For more information about controlling access to tasks and configuring security settings, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

- Adding management servers and configuring primary and secondary management servers for agent computers

  For more information about setting up primary and backup management servers, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

- Organizing the computers in your network into meaningful groups

  For more information about using management groups to manage a group of computers, see the *Control Center User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

- Identifying and establishing Knowledge Script Groups, dynamic views, and monitoring policies for the computers in your environment

  NetIQ Corporation recommends implementing policy-based monitoring in a test environment before you implement it in your production environment. For more information about initiating policy-based monitoring, see the *Control Center User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

Extending AppManager

After you deploy AppManager to all or most of the computers in your environment, you will probably continue improving and streamlining your management process. To ensure reliability and optimal performance, it is important to manage key aspects of the AppManager environment itself. As you extend your deployment, focus on the following goals:

- Deploying reports automatically and designing requirements for customized reporting (for example, reports focused on Service-Level Agreements)

  For more information about configuring reports, see the *Control Center User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).
- Running Knowledge Scripts that are uniquely useful for your environment or troubleshooting specific problems

  For more information about the Knowledge Scripts available with AppManager, see the *AppManager Knowledge Script Reference Guide*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).
- Extending your base monitoring program
- Developing custom Knowledge Scripts

  For more information about creating new Knowledge Scripts, see the *Control Center User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).
- Designing more complex notification or resolution rules
- Integrating AppManager with other products
- Documenting your management and resolution policies and extensions to AppManager
- Maintaining the QDB and CCDB

  For more information about performing repository maintenance, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

4 System Requirements

This chapter describes AppManager system requirements.

Before running the AppManager setup program, review AppManager system requirements and verify that the computers where you plan to install AppManager components meet the requirements.

Although the AppManager pre-installation check program verifies most system requirements, you might need to manually verify some requirements. For the most recent information about third-party software requirements, see the AppManager Supported Products website.

Module Requirements

AppManager offers monitoring support for a large number of modules. Each module might have unique requirements. The module pre-installation check program verifies most module-specific requirements. If a computer does not pass the requirements to monitor an application with a module, that module does not appear in the list of available modules to install.

For more information about module-specific requirements, see the management guide for the module.

For the most recent information about supported product versions and unique requirements for monitoring third-party systems, see the AppManager Supported Products website.

General Requirements for Installing Components

The following table summarizes requirements for running the AppManager setup program.

Pre-install Check Verified?	Requirement
n	Valid Windows login account
y	Microsoft Windows Installer 3.1
n	Valid TEMP or TMP environment variable AppManager might require up to 80 MB of temporary disk space on the drive where the TEMP or TMP folder resides. For more information about determining where the folder resides, see “Understanding Space Considerations” on page 86.
n	Event Viewer closed during the installation If Event Viewer is open, it might cause locking contention with processes trying to write to the Windows Log. No Event Viewers (local or remote) should be viewing the computer where you plan to install AppManager.
n	All Control Panel applets, such as Services, closed
n	Network connectivity between computers For more information about connectivity requirements, see “Understanding Network Connection Requirements” on page 27.

QDB Requirements

The following table describes the requirements for installing the QDB.

Pre-install Check Verified?	Requirement
On the computer where you run the setup program:	
y	Microsoft .NET Framework 3.5 Service Pack 1
y	Microsoft XML parser 3.0 Service Pack 1
On the QDB computer:	
n	One of the following operating systems: ◆ Microsoft Windows Server 2016 Standard or Datacenter edition (GUI mode only) ◆ Microsoft Windows Server 2012 R2 ◆ Microsoft Windows Server 2012 Standard or Datacenter edition (GUI mode only) ◆ Microsoft Windows Server 2008 R2 Standard or Enterprise edition ◆ Microsoft Windows Server 2008 Standard or Enterprise edition (32-bit or 64-bit)
n	One of the following versions of Microsoft SQL Server: ◆ Microsoft SQL Server 2016 Standard or Enterprise edition ◆ Microsoft SQL Server 2014 Standard or Enterprise edition ◆ Microsoft SQL Server 2012 Standard or Enterprise edition (32-bit or 64-bit) ◆ For environments with all components on one computer, Microsoft SQL Server 2012 Express (32-bit or 64-bit) ◆ Microsoft SQL Server 2008 R2 Standard or Enterprise edition (32-bit or 64-bit) ◆ Microsoft SQL Server 2008 Standard or Enterprise edition Service Pack 1 or later (32-bit or 64-bit) ◆ For environments with all components on one computer, Microsoft SQL Server 2008 Express (32-bit or 64-bit)
n	1 GB of RAM
n	100 MB for the QDB data file The default value for the data file is 2048 MB. 50 MB for the QDB log file The default value for the log file is 512 MB. The autogrowth rate is set to 256 MB for each file. Allow enough free space to accommodate the autogrowth rate.
n	SQL Server configured to use the Named Pipes network protocol You can have other protocols in addition to the Named Pipes protocol. If the SQL Server is not configured for any network protocol, run the Microsoft SQL Server setup program before you install AppManager and, at a minimum, add the Named Pipes protocol.

Pre-install Check Verified?	Requirement
n	The following services started and set to run automatically under a domain account: ◆ For SQL Server without instances: `MSSQLServer` ◆ For SQL Server with instances: `MSSQL$`*`Instance_Name`*
n	SQL Server configured to run in **Windows authentication** or **mixed** security mode

Management Server Requirements

The following table describes the requirements for installing the management server.

Pre-install Check Verified?	Requirement
y	One of the following operating systems: ◆ Microsoft Windows Server 2016 Standard or Datacenter edition (GUI mode only) ◆ Microsoft Windows Server 2012 R2 ◆ Microsoft Windows Server 2012 Standard or Datacenter edition (GUI mode only) ◆ Microsoft Windows Server 2008 R2 Standard or Enterprise edition ◆ Microsoft Windows Server 2008 Standard or Enterprise edition (32-bit or 64-bit)
y	512 MB of RAM
n	70 MB of available disk space
n	Microsoft XML Parser 6.0 (`msxml6`) Service Pack 1
y	TCP port 9999 available If another application is using port 9999, reconfigure AppManager to use a different port. For more information about port requirements, see “Reviewing AppManager Port Usage” on page 28.
n	Static IP address (highly recommended) DHCP is supported, but you should not use it if a static IP address is available. If you use DHCP, run the AMAdmin_ConfigSiteCommType Knowledge Script on the agent computers and disable **Communication via IP address**. The agent service on the agent computer then communicates with the management server using the server hostname instead of the IP address. Resolving the hostname incurs more overhead on the agent computer.
n	Network connectivity between this computer and the QDB computer For more information about network connectivity, see “Understanding Network Connection Requirements” on page 27.

Pre-install Check Verified?	Requirement
n	Network connectivity between this computer and the agent computers For more information about network connectivity, see "Understanding Network Connection Requirements" on page 27.
n	For all users on this computer, regional settings set to **English (United States)**

Task Scheduler Service Requirements

The following table describes the requirements for installing the Task Scheduler service.

Pre-install Check Verified?	Requirement
y	One of the following operating systems: ◆ Microsoft Windows Server 2016 Standard or Datacenter edition (GUI mode only) ◆ Microsoft Windows 10 (64-bit) ◆ Microsoft Windows Server 2012 R2 ◆ Microsoft Windows Server 2012 Standard or Datacenter edition (GUI mode only) ◆ Microsoft Windows 8 (64-bit) ◆ Microsoft Windows 7 Business or Enterprise edition (64-bit) ◆ Microsoft Windows Server 2008 R2 Standard or Enterprise edition ◆ Microsoft Windows Server 2008 Standard or Enterprise edition (64-bit)
n	Network connectivity between this computer and the QDB computer For more information about network connectivity, see "Understanding Network Connection Requirements" on page 27.
n	Network connectivity between this computer and the Control Center repository (CCDB) computer For more information about network connectivity, see "Understanding Network Connection Requirements" on page 27.

AppManager Windows Agent Requirements

The following table describes the requirements for installing the AppManager Windows agent. The table does not include requirements for applications you want to monitor with modules. For more information about module requirements, see "Module Requirements" on page 45.

Pre-install Check Verified?	Requirement
y	One of the following operating systems: ◆ Microsoft Windows Server 2016 Standard or Datacenter edition (GUI mode only) ◆ Microsoft Windows Server 2016 Core ◆ Microsoft Windows 10 (32-bit or 64-bit) ◆ Microsoft Windows Server 2012 R2 ◆ Microsoft Windows Server 2012 Standard or Datacenter edition (GUI mode only) ◆ Microsoft Windows 8 (32-bit or 64-bit) ◆ Microsoft Windows 7 Business or Enterprise edition (32-bit or 64-bit) ◆ Microsoft Windows Server 2008 R2 Standard or Enterprise edition, including Itanium ◆ Microsoft Windows Server 2008 R2 Core ◆ Microsoft Windows Server 2008 Standard or Enterprise edition (32-bit or 64-bit), including Itanium ◆ Microsoft Windows Server 2008 Core (32-bit or 64-bit)
y	512 MB of RAM
n	55 MB of available disk space
n	For disk array subsystems, Performance Monitor for disk activities enabled if you plan to run disk-related Knowledge Scripts such as NT_LogicalDiskIO and NT_PhysicalDiskIO Run the program `%systemroot%\system32\diskperf.exe` with the `-y` switch. After enabling the disk counters, reboot your system.
n	SNMP service The service does not need to be running when you install AppManager; however, some Knowledge Scripts, such as SNMPGet, require the SNMP service to be installed and running.
n	Microsoft Exchange client Microsoft Exchange client is required if you select the setup option to enable MAPI mail on a computer that is not an Exchange Server. The client allows AppManager to initiate MAPI mail recovery actions. In addition to installing the Exchange client, create a mailbox for AppManager to use. For more information about using MAPI mail, see "Understanding MAPI Mail Settings" on page 87. If you install the AppManager for Microsoft Exchange module, with or without the MAPI mail option, you can select to have the setup program create an Exchange mailbox. This option does not require installing the Exchange client. However, if you do not select the option to create an Exchange mailbox, you must create a mailbox before running the setup program.

Pre-install Check Verified?	Requirement
y	TCP port 9998 available If another application is using port 9998, reconfigure AppManager to use a different port. For more information about port requirements, see "Reviewing AppManager Port Usage" on page 28.
n	Static IP address (highly recommended) DHCP is supported, but you should not use it if a static IP address is available.
n	Network connectivity between the agent computer and the management server For more information about network connectivity, see "Understanding Network Connection Requirements" on page 27.

The following table describes additional requirements your system must meet if you want to enable a Windows agent to generate reports.

Pre-install Check Verified?	Requirement
y	Internet Explorer 8.0 or later
n	Microsoft XML Parser 3.0 (`msxml3`) Service Pack 1

AppManager UNIX Agent Requirements

The *AppManager for UNIX and Linux Servers Management Guide*, available on the AppManager Modules Documentation page (http://www.netiq.com/documentation/appmanager-modules), describes the system requirements for UNIX agents.

For the most recent information about system requirements and platform support for UNIX agents and applications, see the AppManager Supported Products website.

CCDB Requirements

The following table describes requirements for installing the CCDB.

Pre-install Check Verified?	Requirement
On the computer where you run the setup program:	
y	Microsoft .NET Framework 3.5 Service Pack 1
y	Microsoft XML parser 3.0 Service Pack 1
On the CCDB computer:	

Pre-install Check Verified?	Requirement
n	One of the following operating systems: ◆ Microsoft Windows Server 2016 Standard or Datacenter edition (GUI mode only) ◆ Microsoft Windows Server 2012 R2 ◆ Microsoft Windows Server 2012 Standard or Datacenter edition (GUI mode only) ◆ Microsoft Windows Server 2008 R2 Standard or Enterprise edition ◆ Microsoft Windows Server 2008 Standard or Enterprise edition (32-bit or 64-bit)
n	One of the following versions of Microsoft SQL Server: ◆ Microsoft SQL Server 2016 Standard or Enterprise edition ◆ Microsoft SQL Server 2014 Standard or Enterprise edition ◆ Microsoft SQL Server 2012 Standard or Enterprise edition (32-bit or 64-bit) ◆ For environments with all components on one computer, Microsoft SQL Server 2012 Express (32-bit or 64-bit) ◆ Microsoft SQL Server 2008 R2 Standard or Enterprise edition (32-bit or 64-bit) ◆ Microsoft SQL Server 2008 Standard or Enterprise edition Service Pack 1 or later (32-bit or 64-bit) ◆ For environments with all components on one computer, Microsoft SQL Server 2008 Express (32-bit or 64-bit)
n	256 MB of RAM
n	512 MB of available disk space for the CCDB data file The default value for the data file is 1024 MB. 50 MB of available disk space for the CCDB log file The default value for the log file is 512 MB. The autogrowth rate is set to 256 MB for each file. Allow enough free space to accommodate the autogrowth rate.
n	Microsoft Distributed Transaction Coordinator (DTC), running as a service To ensure Control Center can find and use the DTC service, you might need to change some security settings. For more information about configuring DTC, see “Configuring DTC Security Settings” on page 80.
n	Network connectivity If you install the command queue service on a separate computer, both computers should reside on the same LAN. For more information about network connectivity, see “Understanding Network Connection Requirements” on page 27.
n	The following services started and set to run automatically under a domain account: ◆ For SQL Server without instances: `MSSQLServer` ◆ For SQL Server with instances: `MSSQL$`*`Instance_Name`*

Control Center Command Queue Service Requirements

The following table describes requirements for installing the Control Center command queue service.

Pre-install Check Verified?	Requirement
y	One of the following operating systems: ◆ Microsoft Windows Server 2016 Standard or Datacenter edition (GUI mode only) ◆ Microsoft Windows Server 2012 R2 ◆ Microsoft Windows Server 2012 Standard or Datacenter edition (GUI mode only) ◆ Microsoft Windows Server 2008 R2 Standard or Enterprise edition ◆ Microsoft Windows Server 2008 Standard or Enterprise edition (32-bit or 64-bit)
n	256 MB of RAM
n	64 MB of available disk space For large environments, NetIQ Corporation recommends 110 MB.
y	Microsoft .NET Framework 3.5 Service Pack 1
n	Network connectivity If you install the command queue service and the CCDB on different computers, both computers should reside on the same LAN. For more information about network connectivity, see “Understanding Network Connection Requirements” on page 27.

Control Center Deployment Services Requirements

The following table describes requirements for installing the Deployment Service and the Deployment Web Service.

Deployment Service	
Pre-install Check Verified?	**Requirement**

y	One of the following operating systems: ◆ Microsoft Windows Server 2016 Standard or Datacenter edition (GUI mode only) ◆ Microsoft Windows Server 2012 R2 ◆ Microsoft Windows Server 2012 Standard or Datacenter edition (GUI mode only) ◆ Microsoft Windows Server 2008 R2 Standard or Enterprise edition ◆ Microsoft Windows Server 2008 Standard or Enterprise edition (32-bit or 64-bit)
n	256 MB of RAM
n	64 MB of available disk space
y	Microsoft .NET Framework 3.5 Service Pack 1
y	Microsoft Background Intelligent Transfer Service (BITS) BITS is an operating system feature. For Microsoft Windows Server 2008: BITS version 3.0 For Microsoft Windows Server 2008 R2 or later: BITS version 4.0
n	Network connectivity If you install the Deployment Service and the CCDB on separate computers, both computers should reside on the same LAN. For more information about network connectivity, see “Understanding Network Connection Requirements” on page 27.
n	If the Deployment Service will run in proxy mode, Secure Sockets Layer (SSL) certificate signed by a certification authority The SSL certificate allows the Deployment Service to run in proxy mode to access the Deployment Web Service across a firewall. For more information about the SSL certificate, see “Understanding Deployment Server Configuration” on page 73.
Deployment Web Service	
Pre-install Check Verified?	**Requirement**
y	One of the following operating systems: ◆ Microsoft Windows Server 2016 Standard or Datacenter edition (GUI mode only) ◆ Microsoft Windows Server 2012 R2 ◆ Microsoft Windows Server 2012 Standard or Datacenter edition (GUI mode only) ◆ Microsoft Windows Server 2008 R2 Standard or Enterprise edition ◆ Microsoft Windows Server 2008 Standard or Enterprise edition (32-bit or 64-bit)
n	256 MB of RAM
n	64 MB of available disk space

y	Microsoft .NET Framework 3.5 Service Pack 1
y	Microsoft BITS BITS is an operating system feature. For Microsoft Windows Server 2008: BITS version 3.0 For Microsoft Windows Server 2008 R2 or later: BITS version 4.0 For more information about enabling BITS, see “Enabling BITS and BITS Server Extensions on the Deployment Web Server” on page 74.
y	Microsoft BITS Server Extensions BITS Server Extensions is an optional Microsoft Internet Information Services (IIS) component. For more information about enabling BITS, see “Enabling BITS and BITS Server Extensions on the Deployment Web Server” on page 74.
Pre-install Check Verified?	**Requirement**
y	Microsoft IIS If you are running Microsoft Windows Server 2008 or later, the Web Server (IIS) server role is required. To ensure the computer passes the ASP.NET pre-requisite, also install the IIS 6 Management Compatibility role service with the IIS 6 Metabase Compatibility component.
y	ASP.NET with v2.0.50727 Web Service Extension enabled Also install the IIS 6 Metabase Compatibility component of the IIS 6 Management Compatibility role service. IIS 6 Management Compatibility is a role service of the Web Server (IIS) server role.
n	Network connectivity If you install the Deployment Web Service and the CCDB on different computers, both computers should reside on the same LAN. For more information about network connectivity, see “Understanding Network Connection Requirements” on page 27.
n	If the Deployment Service will run in proxy mode, SSL certificate signed by a certification authority The SSL certificate allows the Deployment Service to run in proxy mode to access the Deployment Web Service across a firewall. For more information about the SSL certificate, see “Understanding Deployment Server Configuration” on page 73.

Control Center Console Requirements

The following table describes requirements for installing the Control Center console.

Pre-install Check Verified?	**Requirement**
y	One of the following operating systems: ◆ Microsoft Windows Server 2016 Standard or Datacenter edition (GUI mode only) ◆ Microsoft Windows 10 (32-bit or 64-bit) ◆ Microsoft Windows Server 2012 R2 ◆ Microsoft Windows Server 2012 Standard or Datacenter edition (GUI mode only) ◆ Microsoft Windows 8 (32-bit or 64-bit) ◆ Microsoft Windows 7 Business or Enterprise edition (32-bit or 64-bit) ◆ Microsoft Windows Server 2008 R2 Standard or Enterprise edition ◆ Microsoft Windows Server 2008 Standard or Enterprise edition (32-bit or 64-bit)
n	512 MB of RAM
n	1 GB of available disk space Depending on the number of agent computers, events you expect to generate, and jobs you expect to run in data collection mode, you might need to allow additional disk space for the local cache folder and paging files. For large environments, NetIQ Corporation recommends 110 MB.
n	256-color display monitor configured for at least 1024x768 display resolution
y	Microsoft .NET Framework 3.5 Service Pack 1

Operator Console Requirements

The following table describes requirements for installing the Operator Console.

Pre-install Check Verified?	**Requirement**
y	One of the following operating systems: ◆ Microsoft Windows Server 2016 Standard or Datacenter edition (GUI mode only) ◆ Microsoft Windows 10 (32-bit or 64-bit) ◆ Microsoft Windows Server 2012 R2 ◆ Microsoft Windows Server 2012 Standard or Datacenter edition (GUI mode only) ◆ Microsoft Windows 8 (32-bit or 64-bit) ◆ Microsoft Windows 7 Business or Enterprise edition (32-bit or 64-bit) ◆ Microsoft Windows Server 2008 R2 Standard or Enterprise edition ◆ Microsoft Windows Server 2008 Standard or Enterprise edition (32-bit or 64-bit)
y	512 MB of RAM
y	Internet Explorer 8.0 or later
y	Microsoft XML Parser 3.0 (`msxml3`) Service Pack 1

VMware Support

NetIQ Corporation supports virtualized installations of AppManager components using VMware vSphere. In a VMware vSphere environment, NetIQ Corporation supports non-clustered configurations and clustered configurations using Microsoft Cluster Service (MSCS). For more information about the AppManager components NetIQ Corporation supports on MSCS, see Appendix D, “Installing on Microsoft Cluster Service,” on page 125.

To install AppManager components in a virtual environment, the virtual environment must meet the requirements for the applicable components as described in this chapter.

While NetIQ Corporation expects AppManager components to function properly in a VMware vSphere environment, there might be performance implications that invalidate typical sizing and implementation recommendations. If you have questions about installing and using AppManager in a VMware vSphere environment, contact Technical Support.

If you install components on virtual hardware, when you view software inventory information, AppManager might identify the components by the name of the physical computer hosting the virtual computer instead of by the virtual computer name.

5 Installation Overview

This chapter provides an overview of the AppManager installation process and describes the recommended order for installing components.

Installation Checklist

The following checklist outlines the basic steps for installing AppManager and provides references to detailed information.

Step		Reference
□	Complete planning activities.	Chapter 2, "Planning to Install AppManager," on page 21
□	Determine the installation order.	"Installing Components in Order" on page 58
□	If necessary, install the required runtime libraries.	"Understanding Installation Methods" on page 58
□	Check for updates to module installation packages.	"Understanding the Check for Updates Utility" on page 60
□	Generate a pre-installation check report and resolve issues.	"Understanding the AppManager Pre-Installation Check" on page 61
□	Install AppManager core components.	◆ Chapter 6, "Installing a Management Site," on page 63 ◆ Chapter 8, "Installing Agent Components," on page 85 ◆ Chapter 7, "Installing Control Center Components," on page 71
□	Install additional core components and optional components according to your implementation plan.	◆ Chapter 6, "Installing a Management Site," on page 63 ◆ Chapter 8, "Installing Agent Components," on page 85 ◆ Chapter 9, "Installing the Operator Console and Console Programs," on page 91
□	Configure security for your environment.	*Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager)

Installing Components in Order

You can install AppManager components in any of these combinations:

- All components, except the UNIX agent, at once

  For information about installing UNIX agents, see the *AppManager for UNIX and Linux Servers Management Guide*, available on the AppManager Modules Documentation page (http://www.netiq.com/documentation/appmanager-modules).
- Individual components one at a time
- A few components at a time

If you install AppManager components one at a time, install them in the following order:

1. Install the QDB and ensure it is running before you install management servers, agents, and Control Center components.

   The QDB stores AppManager information such as events, data, and statistics and the other components interact with it.
2. Install at least one management server.

   Because the management server requires an agent on the same computer, the setup program automatically selects the agent components (the agent services and the AppManager for Microsoft Windows module) for installation when you select the management server.

   Because the management server enables communication between the QDB and agents, install at least one before you install additional agents.
3. Install the Task Scheduler service, which also installs the Task Scheduler Configuration Utility. You will use the utility to add QDBs and the Control Center repository to the service. The repositories will not function correctly until you add them to the service.
4. Install Control Center components.

   Control Center allows you to deploy agents and modules to remote computers.

   Because you need a Windows agent to enable Control Center to monitor the health of AppManager components, the setup program automatically selects the agent components (the agent services and the AppManager for Microsoft Windows module) for installation when you select Control Center and Deployment services.
5. Install additional agents to remote computers.
6. Install other components.

Understanding Installation Methods

You can install AppManager components interactively or silently from a command prompt. For more information about silently installing components, see Appendix B, “Performing a Silent Installation,” on page 99. Regardless of whether you choose to install components interactively or silently, ensure the installation path contains only ASCII characters.

To interactively install 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 install. During the setup process, the setup program automatically installs the runtime libraries required for the selected components. For more information about the pre-installation check, see “Understanding the AppManager Pre-Installation Check” on page 61.

You can run Windows Installer packages for individual components instead of using the AppManager setup program. If you use this method to install 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 installing.

  For more information about installing the runtime libraries, see "Installing Runtime Libraries" on page 59.

- If the computer from which you will run the Windows Installer package has the User Access Control (UAC) feature enabled, ensure the user who will perform the installation is authorized to run the package.

  For more information about ensuring the user is authorized, see "Running Windows Installer Packages When UAC is Enabled" on page 60.

Installing Runtime Libraries

If you choose to install components by running individual Windows Installer packages instead of the AppManager setup program, you must first manually install the required runtime libraries. Runtime support installation packages are located in the `Setup\Setup Files` folder of the AppManager installation package. The Windows Installer packages for the QDB and the Control Center and Deployment services also require Microsoft .NET Framework 3.5 Service Pack 1.

The following table lists the required runtime support packages for each AppManager component.

Component	Required runtime support packages
QDB	No runtime support required
Management server on all platforms	`NetIQ AppManager VC2008 SP1 Runtime Support x86.msi`
Agent on 32-bit platforms	◆ `NetIQ AppManager VC2005 SP1 Runtime Support x86.msi` ◆ `NetIQ AppManager VC2008 SP1 Runtime Support x86.msi`
Agent on 64-bit platforms	◆ `NetIQ AppManager VC2005 SP1 Runtime Support x86.msi` ◆ `NetIQ AppManager VC2008 SP1 Runtime Support x86.msi` ◆ `NetIQ AppManager VC2005 SP1 Runtime Support x64.msi` ◆ `NetIQ AppManager VC2008 SP1 Runtime Support x64.msi`
CCDB	No runtime support required
Control Center and Deployment services on all platforms	`NetIQ AppManager VC2008 SP1 Runtime Support x86.msi`
Control Center console on all platforms	`NetIQ AppManager VC2008 SP1 Runtime Support x86.msi`
Operator Console on all platforms	`NetIQ AppManager VC2008 SP1 Runtime Support x86.msi`

If you want to uninstall the runtime support packages later, you must use the Add or Remove Programs application in Control Panel. Uninstalling AppManager does not automatically uninstall the runtime support packages.

Running Windows Installer Packages When UAC is Enabled

To run a Windows Installer package on an operating system with the UAC feature enabled, you must either start the installation from a command prompt with the run as administrator option or turn off UAC.

To start the installation from a command prompt:

1. Right-click **Command Prompt** and select **Run as administrator**.
2. Change directory to the folder that contains the Windows Installer package.
3. Type the name of the Windows Installer package (for example, `NetIQ AppManager Management Server.msi`) and press **Enter**.

To turn off UAC:

1. Ensure you are logged on as a member of the local Administrators group.
2. Start the User Accounts application in Control Panel.
3. Click **Turn User Account Control on or off**.
4. Deselect **Use User Account Control (UAC)** to help protect your computer.
5. Click **OK**.
6. To apply the changes, restart your computer.

Understanding the Check for Updates Utility

Before you install components, you can compare the versions of AppManager modules in the installation folder with the versions available for download from the NetIQ website. If you maintain the installation files on a shared drive, the utility compares the installation packages on the shared drive to the versions available for download. This allows you to keep the installation folder current so that the latest module installation packages are available for subsequent installations.

To check for updates to module installation packages:

1. From the location where you saved the AppManager installation files, run `Setup.exe`.
2. In the left pane, click **Check for Updates**.
3. In the right pane, click **Check your version of AppManager modules**.

If later versions of module installation packages are available from the NetIQ website, the Check for Updates utility generates a report that lists the versions in the installation folder and the versions available for download. You can click the links provided in the report to download the module installation packages.

If the module installation packages in the installation folder are current, the Check for Updates utility displays the following message:

```
You have the current versions of AppManager modules. To install the current
versions, use the AppManager setup program or use Control Center to install modules
on remote computers.
```

Understanding the AppManager Pre-Installation Check

After you select the components to install from the AppManager setup program, the AppManager pre-installation check script verifies system requirements and generates a report that summarizes the results. You can view the report from the Confirmation window of the AppManager setup program. 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.

If a component fails the pre-installation check, the setup program does not allow you to install that component, but does allow you to install other components that passed the check.

The AppManager setup program does not perform a pre-installation check for the Knowledge Base or the NetIQ AppManager Integration Adapter. A pre-installation check report for the AppManager adapter is available from the AppManager adapter setup program.

To generate a pre-installation check report:

1. Using an account with local Administrator privileges, run `Setup.exe` from the location where you saved the AppManager installation files.
2. Click **Start Installation**.
3. On the Welcome window, select **Production**.
4. Select the components you plan to install and click **Next**.

   The AppManager pre-installation check script verifies system requirements for each component you selected and generates a report that summarizes the results.
5. To view the pre-installation check report, on the Confirmation window, click the link.

If your environment passed all requirements, you are ready to install components. If your environment did not pass all requirements, resolve issues and re-generate the pre-installation check report.

Canceling the Installation

Once the AppManager setup program launches the Windows Installer package for a component, to cancel installation of that component, click the **Cancel** button for the Windows Installer package. Clicking the **Cancel** button for the AppManager setup program will cancel installation of the components in the installation queue, but not installation of the component currently being installed.

Canceling installation of the QDB or CCDB does not undo changes that were already completed during installation. To undo changes that were already completed, use Microsoft SQL Server Management Studio to delete the QDB or CCDB. For more information about deleting the QDB or CCDB, see "Uninstalling the QDB or CCDB" on page 123.

Reviewing AppManager Log Files

Each AppManager component has at least a custom log and an MSI/InstallShield log associated with it. The custom log is usually named `nq*.log`. The MSI/InstallShield log is usually named the same as the installation package, with `.log` appended. The following table lists the log files for each component.

Component	Log Files	Default Location
AppManager suite	`nqAMInst_Setup.log`	`\NetIQ\Temp\NetIQ_Debug\`*`Computer_Name`*
QDB	`kscheckin.log`	`Users\`*`User_Name`*`\AppData\NetIQ\NetIQ_Debug\`*`Computer_Name`*
	`nqAMInst_QDB.log`	`\Windows\Temp`
	◆ `NetIQDatabaseManager.log` ◆ `NetIQDatabaseUtility.log`	`\`*`User_Profile_Temp_Folder`*`\NetIQ`
Management server	◆ `nqAMInstMS.log` ◆ `ms.log` ◆ `msrplib.log`	`\NetIQ\Temp\NetIQ_Debug\`*`Computer_Name`*
Agent	◆ `nqAMInstMC.log` ◆ `ccmtrace.log` ◆ `mctrace.log`	`\NetIQ\Temp\NetIQ_Debug\`*`Computer_Name`*
CCDB	◆ `nqCCDB_Install.log` ◆ `rplib.log`	`\NetIQ\Temp\NetIQ_Debug\`*`Computer_Name`*
	◆ `NetIQDatabaseManager.log` ◆ `NetIQDatabaseUtility.log`	`\`*`User_Profile_Temp_Folder`*`\NetIQ`
Control Center services	`nqCC_Install.log`	`\NetIQ\Temp\NetIQ_Debug`
	`nqXmlUtil.log`	`\NetIQ\Temp\NetIQ_Debug\`*`Computer_Name`*
	`DeploymentService.log`	`\NetIQ\Temp\NetIQ_Debug\CC_ADSTrace`
	`CQSLog.txt`	`\NetIQ\Temp\NetIQ_Debug\CC_CQSTrace`
Control Center console	`ccLog.Console_Instance.`*`User_Name`*`.txt`	`\`*`User_Profile_Folder`*`\`*`User_Name`*`\AppData\Local\NetIQ\AppManager Control Center\Logs`
Module	◆ *`Module_Name`*`_Install.log` `AM`*`nn`*`-`*`Module_Name`*`-`*`n.n.nn.n`*`.msi.log`	`\NetIQ\Temp\NetIQ_Debug\`*`Computer_Name`*

6 Installing a Management Site

This chapter describes the steps for interactively installing management site components. A management site comprises one QDB, one or more management servers, and the Task Scheduler service.

You can install management site components silently from a command prompt. For more information about silently installing components, see Appendix B, “Performing a Silent Installation,” on page 99.

Understanding QDB Installation

Typically, you install the QDB and the management server on different computers. NetIQ Corporation recommends that you also install an agent on the same computer as the QDB to facilitate database management. For more information about installing agents, see Chapter 8, “Installing Agent Components,” on page 85.

You can install the QDB to remote SQL Servers. You do not have to run the setup program on the SQL Server.

If you plan to install a report agent and you want the agent to generate Active Directory reports, install the QDB on a member server of the same domain or a trusted domain. For more information about report agents, see “Understanding Agent Reporting Capabilities” on page 86.

You can install the QDB and the CCDB on computers that belong to different domains. For more information about installing the CCDB, see Chapter 7, “Installing Control Center Components,” on page 71.

For information about installing the QDB on MSCS, see the Appendix D, “Installing on Microsoft Cluster Service,” on page 125.

Understanding Accounts Required for the Installation

During QDB installation, the setup program prompts you for an account that can log in to the SQL Server to create the QDB, and for an account to serve as database owner of the QDB. The database owner account will be used to create tables, users, and stored procedures and manage data in the QDB. For more information about the account requirements, see “Reviewing Required Accounts and Permissions” on page 24.

Understanding QDB Security Options

During QDB installation, you specify security options for the agent computers that report to the QDB. Depending on your environment, you can configure security for Windows agents only, for UNIX agents only, or for both Windows and UNIX agents.

The security level you select during QDB installation affects all communications between the management servers and agent computers within the management site. AppManager offers the following options for securing communication between agents and management servers:

- **Encrypted communications only**

If you select this option, AppManager encrypts data transmissions between agents and management servers, but does not require agents to authenticate the management servers with which they communicate.

- **Authentication and encrypted communications**

  If you select this option, AppManager encrypts data transmissions between agents and management servers and requires agents to authenticate management servers before they transmit data.

If you select either of the secure communication options when you install the QDB, the setup program creates a password-protected encryption key in the QDB and prompts you for a password for agents to use to access their portion of the key. When you install agents, ensure the agents use the same security level and password as the QDB to which they will report.

For more information about security and managing site communications between management servers and agent computers, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

Restricting Knowledge Script Check-in

The setup program checks a copy of every current Knowledge Script into the QDB during installation. To exclude Knowledge Scripts associated with applications you do not use, remove the applicable files from the `\Setup\Setup Files` folder. For example, to exclude all Knowledge Scripts in the AppManager for BlackBerry and AppManager for BES Knowledge Script categories, delete the following files from the `Setup Files` folder, where *xx* is the AppManager or module version:

- `AM`*`xx`*`-BlackBerry-`*`xx`*`.ini`
- `AM`*`xx`*`-BlackBerry-`*`xx`*`.msi`
- `AM`*`xx`*`-BES-`*`xx`*`.ini`
- `AM`*`xx`*`-BES-`*`xx`*`.msi`

WARNING: Do not remove any executable files, such as `ckBES.exe`.

Installing the QDB

This section describes the steps required to install the QDB on local and remote SQL Servers.

To install the QDB:

1. Ensure the accounts required to install the QDB are properly configured.

   For more information about the required accounts, see "Reviewing Required Accounts and Permissions" on page 24.

2. Complete the steps in "Understanding the AppManager Pre-Installation Check" on page 61.

   After you view the pre-installation check report, click **Next** to start the installation.

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

   - Name of the SQL Server and, if applicable, instance that will host the QDB. To specify a SQL Server instance, use the format *Server_Name\instance*.

- Name of the QDB. Do not include spaces at the beginning or end of the QDB name, and do not specify a name that is longer than 128 characters. If the name is too long, the installation will fail.
- Account that can log in to the SQL Server to create the QDB.

4 Specify initial sizes and locations for the QDB data and log files and click **Next**.

The minimum required value for the data file size is 100 MB. The default value is 2048 MB.

The minimum required value for the log file size is 50 MB. The default value is 512 MB.

You can adjust the default values based on the amount of data you intend to collect and the number of computers you plan to monitor. For more information about sizing the QDB, see "Sizing the QDB" on page 30.

Because the autogrowth rate is set to 256 MB for each file, allow at least enough free space to accommodate the autogrowth rate.

If you specify a non-default location for the files, ensure the folders exist before you install the QDB.

5 Provide information about the account that will own the QDB and click **Next**.

If you specify a Windows user account, the QDB becomes the default database for the Windows user in Microsoft SQL Server. Before you uninstall the QDB, use Microsoft SQL Server Management Studio to change the default database for the Windows user account to an appropriate system database. Otherwise, the Windows user will not be able to connect to the SQL Server after you uninstall the QDB. For more information about uninstalling the QDB, see Appendix C, "Uninstalling AppManager," on page 121.

6 Select to configure security for Windows agents only, UNIX agents only, or both and click **Next**.

NetIQ Corporation recommends initially configuring all Windows agents and all UNIX agents to use the same security level. After installation, you can use the `NQKeyGenWindows.exe` and `NQKeyGenUnix.exe` utilities to manage security separately for Windows and UNIX agents. For more information about using these utilities, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

7 Select the security level for communication between management servers and agents and click **Next**.

For more information about the security levels, see "Understanding QDB Security Options" on page 63.

8 (Conditional) If you selected **Encrypted communications only** or **Authentication and encrypted communications**, specify a password for retrieving encryption keys from the QDB and click **Next**.

For information about QDB encryption keys, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

9 (Conditional) If you selected **Encrypted communications only** or **Authentication and encrypted communications**, provide the following information and click **Next**:

- A password for the agent to access its portion of the QDB encryption key
- The location for the agent encryption key file

For Windows agents, you can also select to export the key information to a text file. For UNIX agents using authentication and encrypted communications, the key file information is automatically exported to a text file. For UNIX agents using encrypted communications only, the key file information is not exported to a text file.

The name of the Windows agent key file is `nqWindowsPublic0.key` and the name of the UNIX agent key file is `nqUNIXPublic0.key`. You can rename and move the key files after installation. You will need the agent key file location when you install Windows agents. For more information about the key files, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

10 Review the installation settings. When you are ready to install the QDB, click **Install**.

For the QDB to function correctly, you must also install the Task Scheduler service and add the QDB to the service. For more information about installing and configuring the service, see "Understanding Task Scheduler Service Installation" on page 66, "Installing the Task Scheduler Service" on page 66, and "Adding Repositories to the Task Scheduler Service" on page 67.

After you install Control Center components, you can use Control Center to manage QDBs. To manage a QDB through Control Center, use the Control Center console to add the QDB to Control Center. You cannot add a QDB to more than one CCDB. The primary QDB must be the same version as the CCDB. Before you add an existing non-primary QDB to Control Center, ensure that it is version 8.2 or later.

For more information about installing Control Center components, see Chapter 7, "Installing Control Center Components," on page 71. For more information about adding a QDB to Control Center, see the *Control Center User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

Understanding Task Scheduler Service Installation

To install the Task Scheduler service, you must run the setup program on the computer where you want to install the service. Installing the service also installs the Task Scheduler Configuration Utility. You can either choose to run the utility immediately after you complete the service setup program, or you can choose to run it later. You must use the utility to add QDBs and the Control Center repository (CCDB) to the service. The repositories will not function correctly until you add them to the service.

During installation, the setup program prompts you for an account under which the service will run. You can select to use the local system account or provide a Windows user account. If you select to use the local system account, the repositories that the service will manage must use SQL Server authentication.

Regardless of the option you select for the service account, the account must have access to the SQL Server instances that host the databases the service will manage. If the account does not already have the Log on as a service privilege, the setup program grants it.

Installing the Task Scheduler Service

This section describes the steps required to install the Task Scheduler service.

To install the service:

1 Decide whether you want the service to use the local system account or a Windows user account and properly configure the account.

For more information about the required accounts, see "Reviewing Required Accounts and Permissions" on page 24.

2 Complete the steps in "Understanding the AppManager Pre-Installation Check" on page 61.

After you view the pre-installation check report, click **Next** to start the installation.

3 When you reach the Destination Folder window, select the folder where you want to install the service and click **Next**.

4 Select the type of account the service will use to connect to the repositories.

5 (Conditional) If the service will use a Windows user account to connect to the repositories, provide the account credentials.

6 Review the installation settings. When you are ready to install the service, click **Install**.

When the installation is complete, you can select to start the Task Scheduler Configuration Utility so that you can add repositories to the service. If you choose not to start the utility immediately after the installation completes, you must use the **Start** menu to start the utility. The repositories will not function correctly until you add them to the service. For information about adding repositories to the service, see "Adding Repositories to the Task Scheduler Service" on page 67.

Adding Repositories to the Task Scheduler Service

For QDBs and the CCDB to function correctly, you must use the Task Scheduler Configuration Utility to add them to the Task Scheduler service so that it can schedule required SQL Server jobs.

To add a repository to the service:

1 In the **Tasks** pane, click **Add**.

2 Provide the required information about the repository.

If you are using a port other than the default port (1433) for communications with SQL Server, you must use the format *SQL_Server_Name*/*Instance*,*Port_Number* when you specify the SQL Server name.

3 Click **OK**.

After you add the repository, the Task Scheduler service creates and runs the SQL Server jobs that are associated with the repository. If the SQL Server that hosts the repository is not available for any reason (for example, the Task Scheduler service starts before the SQL Server service), the Task Scheduler service is not able to add the SQL Server jobs and AppManager generates a repository synchronization error in the Control Center console. Because the Task Scheduler service checks all of the SQL Servers that you add to the Task Scheduler Configuration Utility every 30 minutes to discover new or removed SQL Server jobs and update the Configuration Utility, AppManager should automatically resolve the error within 30 minutes. If you need to resolve the error more quickly, you can modify the `SQLJobDiscoveryInterval` parameter in the `NetIQTaskScheduler.exe.config` file (located by default in `C:\Program Files\NetIQ\AppManager\TaskScheduler\bin`). For more information about the configuration file, see Configuring the Task Scheduler Service and SQL Server Jobs in the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

To remove a repository from the service:

1 In the **Tasks** pane, click **Remove**.

2 Select the repository you want to remove, and then click **OK**.

You can also use the utility to perform the following tasks:

- Change the authentication method for a repository
- Disable a SQL Server job
- Change the schedule for a SQL Server job

For information about performing these tasks, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

Understanding Management Server Installation

Because the management server requires an agent on the same computer, the setup program automatically selects the agent components (the agent services and the AppManager for Microsoft Windows module) for installation when you select the management server. For more information about installing agents, see Chapter 8, “Installing Agent Components,” on page 85.

NetIQ Corporation recommends installing at least two management servers per QDB and designating one as primary and one as secondary to provide failover support for agent computers. For more information about determining the number of management servers for your environment, see “Implementation Guidelines” on page 35. For more information about designating primary and secondary management servers, see Chapter 8, “Installing Agent Components,” on page 85 and the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

Reviewing Management Server Port Information

For information about the default ports that enable communication between the management server and the Windows and UNIX agents, see “Reviewing AppManager Port Usage” on page 28. If you need to change the ports the management server and agents use, update both the management server and each agent computer with which the management server communicates. For example, if you change the port to which the management server binds for receiving information from the agent computer but do not set corresponding port information when you install the agent, the management server and the agent computer cannot communicate. For more information about changing the default listening ports, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager). Consult a network security administrator before you change ports.

Understanding Accounts Required for the Installation

During management server installation, the setup program prompts you for an account for the NetIQ AppManager Management Service (`NetIQms`) to use, and for an account for the management server to use to connect to the QDB. For more information about the account requirements, see “Reviewing Required Accounts and Permissions” on page 24.

Discovering Management Site Components for Health Monitoring

Once the setup program successfully installs the management server, if an 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 agent installation. For information about using Control Center to monitor the health of your AppManager components, see the *Control Center User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

Installing the Management Server

This section describes the steps required to install the management server.

To install the management server:

1. Ensure the accounts required to install the management server are properly configured.

   For more information about the required accounts, see "Reviewing Required Accounts and Permissions" on page 24.
2. Ensure the QDB to which the management server will connect is running.
3. Complete the steps in "Understanding the AppManager Pre-Installation Check" on page 61.

   After you view the pre-installation check report, click **Next** to start the installation.
4. When you reach the Destination Folder window, select the folder where you want to install the management server and click **Next**.
5. Accept the default ports for communication between the management server and agents or change the port information and click **Next**.
6. Provide information about the account the NetIQ AppManager Management Service (`NetIQms`) will use and click **Next**.
7. Provide the following information and click **Next**:
   - Name of the SQL Server and, if applicable, instance that hosts the QDB to which the management server will connect. To specify a SQL Server instance, use the format *Server_Name\instance*.
   - Name of the QDB to which the management server will connect.
   - Account the management server will use to connect to the QDB.
8. Review the installation settings. When you are ready to install the management server, click **Install**.

If you want to use Performance Monitor to monitor the operational health and performance of the management server and the agent on the management server, you must manually install performance counters. For more information about installing the counters, see "Installing Management Server Performance Counters" on page 69 and "Installing Agent Performance Counters" on page 89.

Installing Management Server Performance Counters

You can use Microsoft Performance Monitor to monitor the operational health and performance of the management server. Installing the management server does not automatically install the performance counters. You must manually install the counters.

To install 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"`

If you installed the performance counters on a 64-bit operating system, to view the counters, you must open Performance Monitor from the `Windows\SysWOW64` folder.

7 Installing Control Center Components

This chapter describes the procedures for interactively installing Control Center components.

You can install Control Center components silently from a command prompt. For more information about silently installing components, see Appendix B, “Performing a Silent Installation,” on page 99.

Understanding Control Center Installation

Control Center consists of the following components:

- CCDB, a SQL Server database that stores information Control Center collects from the QDBs it manages, user preferences, and security settings

  You can install the CCDB to remote SQL Servers. You do not have to run the setup program on the SQL Server.

  You can install the CCDB and the QDB on computers that belong to different domains. For more information about installing the QDB, see Chapter 6, “Installing a Management Site,” on page 63.

  For information about installing the CCDB on MSCS, see Appendix D, “Installing on Microsoft Cluster Service,” on page 125.
- Command queue service, a Windows service that performs the following functions:
  - Retrieves commands from the CCDB, sends them to the appropriate QDBs, and maintains the command status
  - Supports multiple Control Center consoles if they are connected to the same CCDB
  - Handles error recovery
- Deployment Service, which communicates with the CCDB to process deployment rules and tasks

  The computer where you install the Deployment Service is the deployment server. You can have multiple deployment servers for deploying agents remotely. If a firewall is active on your network between the deployment server and the CCDB, the Deployment Service can run in proxy mode, which allows it to use the Deployment Web Service to communicate with the CCDB. For more information about how the Deployment Service communicates with the CCDB and enabling the service to run in proxy mode, see “Understanding Deployment Server Configuration” on page 73.
- Deployment Web Service, which consists of two Web services on a Microsoft Internet Information Services (IIS) server called the Web Depot

  The Deployment Web Service performs the following functions:
  - Checks deployment packages into the Web Depot

    For more information about checking in deployment packages, see “Understanding Package and Deployment Rule Check-in” on page 74.
  - Distributes deployment packages to the Deployment Service using Microsoft Background Intelligent Transfer Service (BITS) server extensions

    You must enable BITS and BITS Server Extensions on the deployment web server before you install the Deployment Web Service. For more information about enabling BITS and BITS Server Extensions, see “Enabling BITS and BITS Server Extensions on the Deployment Web Server” on page 74.

  - Provides a communication proxy for Deployment Services across a firewall

  Installing the Deployment Web Service creates three virtual directories under the default website in IIS:

  - `DeploymentWebService`
  - `ProxyDeploymentWebService`
  - `WebDepot`

  The `ProxyDeploymentWebService` directory is only used if you run the Deployment Service in proxy mode for cross-firewall deployments.

- Control Center console, which connects to the CCDB and allows you to run jobs on the systems and applications you manage across multiple QDBs

Because you need a Windows agent to enable Control Center to monitor the health of AppManager components, the setup program automatically selects the agent components (the agent services and the AppManager for Microsoft Windows module) for installation when you select the Control Center and Deployment services for installation. For more information about installing agents, see Chapter 8, “Installing Agent Components,” on page 85.

NetIQ Corporation recommends distributing Control Center components across computers to improve performance. Because the command queue service runs under a Windows user account and connects to the CCDB using that account, ensure no firewall is present between the command queue service and CCDB computers. On computers running Microsoft Windows Server 2008 or later, a firewall is enabled by default. At a minimum, you will need to open ports 1433 and 135.

If you distribute Control Center components across computers and use Windows authentication between Control Center and the QDBs it manages, configure Kerberos delegation to ensure successful communication between components. For more information about configuring Kerberos delegation, see “Configuring Kerberos Delegation for a Distributed Control Center Environment” on page 75.

While larger networks require multiple QDBs and management servers, a single CCDB can manage your entire organization. Similarly, a single Deployment Web Service is sufficient for your entire organization. NetIQ Corporation recommends installing multiple deployment servers reporting to a single Deployment Web Service and CCDB. Install a deployment server for every firewall-separated segment of your network.

If you install Control Center components on computers in separate network domains, the CCDB uses both the domain name and user name for authentication purposes. Making connections between Control Center components across *untrusted* domains is not possible unless you install the CCDB on a SQL Server instance using SQL authentication. To allow a Control Center console to connect to a CCDB in a different *trusted* network domain using Windows authentication, you must add an Administrator account (in *domain*\Administrator format) as a Control Center user. For more information about adding users to Control Center, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

Understanding Microsoft DTC Connectivity

The CCDB uses Microsoft Distributed Transaction Coordinator (DTC) to connect to the QDBs it manages. DTC must run as a service on the CCDB computer. During Control Center and Deployment services installation, you can run a utility to verify that DTC connectivity exists between the CCDB and each QDB it will manage. For more information about checking DTC connectivity and troubleshooting connectivity issues, see "Verifying Microsoft DTC Connectivity" on page 80 and "Troubleshooting DTC Connectivity" on page 81.

Understanding the CCDB Accounts

During CCDB installation, the setup program prompts you for an account that can log in to the SQL Server to create the CCDB, and for an account to serve as database owner of the CCDB. For more information about the account requirements, see "Reviewing Required Accounts and Permissions" on page 24.

Understanding the Command Queue Service, Deployment Service, and Deployment Web Service Accounts

Each service runs under a Windows user account that allows the service to connect to the CCDB. The command queue service can also use the account to access each managed QDB. For more information about the account requirements, see "Reviewing Required Accounts and Permissions" on page 24.

If a firewall is present between the Deployment Service and the CCDB, the Deployment Service uses the Deployment Web Service account to connect to the CCDB. For more information about how the Deployment Service connects to the CCDB, see "Understanding Deployment Server Configuration" on page 73.

You can change the Deployment Service and Deployment Web Service accounts after installation. For more information about changing the Deployment Service account, see "Changing the Deployment Service User Account" on page 82. For more information about changing the Deployment Web Service account, see "Changing the Deployment Web Service User Account" on page 83.

Understanding Deployment Server Configuration

The Deployment Service must be able to retrieve task information from the CCDB, either directly or in proxy mode. If you plan to install multiple Deployment Services, NetIQ Corporation recommends co-locating them in your remote sites.

If no firewall is active between the deployment server and the CCDB, the Deployment Service requires a Windows user account to access the CCDB. If the Deployment Service cannot directly access the CCDB because of firewalls, the deployment server must use the Deployment Web Service as a proxy to access the CCDB. In this case, the deployment server uses the Deployment Web Service account to connect to the CCDB.

To allow the Deployment Service to run in proxy mode, enable SSL security on the IIS Web Server for the default website and install an SSL certificate signed by a certification authority on the proxy deployment server and the Deployment Web Service computer. Do not use a self-signed certificate. When you enable SSL security, do **not** enable the option to **require SSL** for the certificate. For more information about installing an SSL certificate, see the Microsoft documentation for your operating system.

Understanding Package and Deployment Rule Check-in

If you are installing the Deployment Web Service, you can select to check in packages and rules for use in remote deployment. If you choose not to check in packages and rules as part of installation, you can check them in later using the Control Center console. For more information about checking in packages and rules after installation, see the *Control Center User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

Packages are installation files for deploying agents and modules. The package check-in procedure makes the installation and configuration files associated with all modules and the Windows agent available to the Control Center console and CCDB.

The default deployment rules are samples that can help you perform basic deployments of agents and modules, with modifications. The rules are disabled by default. Deployment does not occur until you edit and configure the rules for your environment and enable them.

Discovering Control Center Components for Health Monitoring

Once the setup program successfully installs the command queue service, if an 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 agent installation. For information about using Control Center to monitor the health of your AppManager components, see the *Control Center User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

Enabling BITS and BITS Server Extensions on the Deployment Web Server

The Deployment Web Service requires the Microsoft BITS operating system feature and the BITS Server Extensions component of Microsoft IIS. You must enable BITS and BITS Server Extensions on the Deployment Web Server before you install the Deployment Web Service. This section describes the steps required to enable BITS and BITS Server Extensions based on the operating system you are running.

To enable BITS and BITS Server Extensions:

1. In Server Manager, select **Features** and click **Add Features**.
2. (Conditional) If you are running Microsoft Windows Server 2008, select **BITS Server Extensions**.

   BITS Server Extensions includes the BITS operating system feature and the BITS Server Extensions component of Microsoft IIS.
3. (Conditional) If you are running Microsoft Windows Server 2008 R2 or later, select **Background Intelligent Transfer Service (BITS)**.

Background Intelligent Transfer Service (BITS) includes the BITS operating system feature and the BITS Server Extensions component of Microsoft IIS.

4 Click **Install**.

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 service for each QDB computer and the CCDB computer to be trusted for delegation.
- Configure the CCDB computer to impersonate the SQL Server service 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 TCP/IP is listed first.

2 Determine the TCP dynamic port number the SQL Server service is using and verify it is not blocked by a firewall.

3 Ensure the SQL Server service is running under a domain account.

4 (Conditional) If you are running Windows Server 2008 or later, 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_Service_Runs}
```

```
setspn -A MSSQLSvc/{Fully-
qualified_Domain_Name_of_the_QDB_or_CCDB_Computer}:{Port_on_Which_the_SQL_
Server_Service_Runs}{Domain_Account_Name_Under_Which_the_SQL_Server_Service_Ru
ns}
```

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

```
setspn -A MSSQLSvc/
{NETBIOS_Name_of_the_QDB_or_CCDB_Computer}:{Port_on_Which_the_SQL_
Server_Service_Runs}
{Domain_Account_Name_Under_Which_the_SQL_Server_Service_Runs}
```

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 service runs 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 service runs and click **OK**.
6. Select the `MSSQLSvc` entries associated with the QDB or CCDB computer and click **OK**.
7. (Conditional) If the SQL Server service 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_Service_Runs}
   ```

   ```
   setspn -A MSSQLSvc/{NETBIOS_Name_of_the_QDB_or_CCDB_Computer}:{DNS_Service_Port} {Domain_Account_Name_Under_Which_the_SQL_Server_Service_Runs}
   ```

   ```
   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_Service_Runs}
   ```

   ```
   setspn -A MSSQLSvc/{NETBIOS_Name_of_the_QDB_or_CCDB_Computer}:{Kerberos_Ticket_Granting_Service_Port} {Domain_Account_Name_Under_Which_the_SQL_Server_Service_Runs}
   ```

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

   ```
   setspn -A MSSQLSvc/{NETBIOS_Name_of_the_QDB_or_CCDB_Computer}:{Time_Service_Port} {Domain_Account_Name_Under_Which_the_SQL_Server_Service_Runs}
   ```
8. Restart the SQL Server service on the QDB or CCDB computer.

To configure the CCDB computer to impersonate the SQL Server service 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 service runs 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 76.

Installing the CCDB

This section describes the steps required to install the CCDB on local and remote SQL Servers.

To install the CCDB:

1. Ensure the accounts required to install the CCDB are properly configured.

   For more information about the required accounts, see “Reviewing Required Accounts and Permissions” on page 24.

2. Complete the steps in “Understanding the AppManager Pre-Installation Check” on page 61.

   After you view the pre-installation check report, click **Next** to start the installation.

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

   - Name of the SQL Server and, if applicable, instance that will host the CCDB. To specify a SQL Server instance, use the format *Server_Name\instance*.
   - Name of the CCDB. Do not include spaces at the beginning or end of the CCDB name, and do not specify a name that is longer than 128 characters. If the name is too long, the installation will fail.
   - Account that can log in to the SQL Server to create the CCDB.

4. Provide information about the account the command queue service will use and click **Next**.

5. Provide information about the account that will own the CCDB and click **Next**.

   If you specify a Windows user account, the CCDB becomes the default database for the Windows user in Microsoft SQL Server. Before you uninstall the CCDB, use Microsoft SQL Server Management Studio to change the default database for the Windows user account to an appropriate system database. Otherwise, the Windows user will not be able to connect to the SQL Server after you uninstall the CCDB. For more information about uninstalling the CCDB, see Appendix C, “Uninstalling AppManager,” on page 121.

6. Specify initial sizes and locations for the CCDB’s data and log files and click **Next**.

   The minimum required value for the data file size is 512 MB. The default value is 1024 MB.

   The minimum required value for the log file size is 50 MB. The default value is 512 MB.

   Adjust the data and log file sizes based on the number of CCDBs you plan to manage and the number of management groups and views you plan to create.

   Because the autogrowth rate is set to 256 MB for each file, allow at least enough free space to accommodate the autogrowth rate.

   If you specify a non-default location for the files, ensure the folders exist before you install the CCDB.

7. Provide information about the account the Deployment Service will use and click **Next**.

   You can specify the same account you specified for the command queue service, or you can specify a different account.

8. Review the installation settings. When you are ready to install the CCDB, click **Install**.

For the CCDB to function correctly, you must also install the Task Scheduler service and add the CCDB to the service. For more information about installing and configuring the service, see "Understanding Task Scheduler Service Installation" on page 66, "Installing the Task Scheduler Service" on page 66, and "Adding Repositories to the Task Scheduler Service" on page 67.

Installing Control Center and Deployment Services

This section describes the steps required to install the Control Center and Deployment services.

To install the Control Center and Deployment services:

1. Ensure the accounts required to install the services are properly configured.

   For more information about the account requirements, see "Reviewing Required Accounts and Permissions" on page 24.
2. Complete the steps in "Understanding the AppManager Pre-Installation Check" on page 61.

   After you view the pre-installation check report, click **Next** to start the installation.
3. (Optional) If you want to check DTC connectivity between the CCDB and the QDBs it will manage, run the Configuration Checker utility and click **Next**.

   For more information about running the utility and troubleshooting connectivity issues, see "Verifying Microsoft DTC Connectivity" on page 80 and "Troubleshooting DTC Connectivity" on page 81.
4. When you reach the Select Features window, select the services to install and click **Next**.
5. (Conditional) If you are installing the command queue service, provide information about the Windows user account the command queue service will use to connect to the CCDB and click **Next**.
6. (Conditional) If you are installing the command queue service, provide information about the CCDB to which the service will connect and click **Next**.

   Connect only one command queue service to a CCDB.
7. (Conditional) If you are installing the Deployment Web Service, select whether to check in agent and module packages and a set of default rules for use in remote deployment and click **Next**.

   For more information about package and deployment rule check-in, see "Understanding Package and Deployment Rule Check-in" on page 74.
8. (Conditional) If you are installing the Deployment Web Service, provide information about the account the service will use to connect to the CCDB and click **Next**.
9. (Conditional) If you are installing the Deployment Web Service, provide information about the CCDB to which the service will connect and click **Next**.

   Connect only one Deployment Web Service to a CCDB.
10. (Conditional) If you are installing the Deployment Service, provide information about how the deployment server will connect to the CCDB and click **Next**.

    For more information about how the deployment server accesses the CCDB, see "Understanding Deployment Server Configuration" on page 73.
11. (Conditional) If you are installing the Deployment Service and no firewall exists between the service and the CCDB, provide information about the account the service will use to connect to the CCDB and click **Next**.
12. Review the installation settings. When you are ready to install the services, click **Install**.

After installation, you can use the Control Center Configuration Checker utility to verify the services are properly configured to remotely deploy agents and updates. For more information about verifying proper configuration, see "Validating Deployment Services Installation" on page 81.

Installing the Control Center Console

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

To install the Control Center console:

1. Complete the steps in "Understanding the AppManager Pre-Installation Check" on page 61.

   After you view the pre-installation check report, click **Next** to start the installation.
2. When you reach the User Information window, provide your identification information and click **Next**.
3. (Conditional) If you do not have other AppManager components installed on the computer or you only have the QDB or CCDB installed, select the folder where you want to install the Control Center console and click **Next**.
4. Review the installation settings. When you are ready to install the Control Center console, click **Install**.

After you install the CCDB, Control Center and Deployment Services, and Control Center console, you can use Control Center to manage QDBs. To manage QDBs through Control Center, use the Control Center console to add the QDBs to Control Center. You cannot add a QDB to more than one CCDB. The primary QDB must be the same version as the CCDB. Before you add an existing non-primary QDB to Control Center, ensure that it is version 8.2 or later.

For more information about adding a QDB to Control Center, see the *Control Center User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

Using the Control Center Configuration Checker Utility

The AppManager setup program installs the Control Center Configuration Checker utility with the Control Center and Deployment services. Use the utility to test Microsoft DTC connectivity between the CCDB and the QDBs it manages and to validate your Deployment services installation. For more information about testing Microsoft DTC connectivity, see "Verifying Microsoft DTC Connectivity" on page 80. For more information about validating your Deployment services installation, see "Validating Deployment Services Installation" on page 81.

Verifying Microsoft DTC Connectivity

Control Center uses the Microsoft DTC to connect to every QDB it manages. The DTC must run as a service on the computer where you install the CCDB. To ensure Control Center can find and use DTC, you might need to change some security settings. For more information about the recommended security settings, see "Configuring DTC Security Settings" on page 80.

Run the Control Center Configuration Checker utility for each QDB you plan to manage with Control Center.

To check DTC connectivity between the CCDB and QDBs:

1. Ensure Microsoft .NET Framework 3.5 Service Pack 1 is installed.
2. On the Configuration Check window of the Control Center setup program, click **Run**.
3. Provide the following information and click **OK**:
   - Name of the SQL Server and, if applicable, instance that hosts the CCDB for which you are testing connectivity
   - Name of the CCDB for which you are testing connectivity
   - Name of the SQL Server and, if applicable, instance that hosts the QDB for which you are testing connectivity

   The Configuration Checker utility runs tests in the Preinstall category to verify that DTC connectivity exists between the CCDB and QDB and check the validity of the SQL Server names.
4. To run the utility for another QDB, select **Tasks** > **Server Setup**.

For more information about resolving issues with DTC connectivity, see "Troubleshooting DTC Connectivity" on page 81.

Configuring DTC Security Settings

This section describes how to configure the DTC security settings to ensure Control Center can find and use DTC. If you use clustered servers for the CCDB, complete the task for each cluster node.

If DTC communications must pass through a firewall, you might need to reconfigure DTC to work through the firewall after changing the security configuration. For information about reconfiguring DTC to work through a firewall, see http://support.microsoft.com/kb/311846.

To configure the DTC security settings:

1. In the Component Services application in Administrative Tools, expand **Component Services** and then expand **Computers**.
2. Expand **My Computer** and then expand **Distributed Transaction Coordinator**.
3. Right-click **Local DTC** and select **Properties**.
4. On the **Security** tab, select the following items:
   - **Network DTC Access**
   - **Allow Remote Clients**
   - **Allow Remote Administration**
   - **Allow Inbound**
   - **Allow Outbound**

- **No Authentication Required**
- **Enable XA Transactions**

5 Restart the computer.

Troubleshooting DTC Connectivity

This section describes common DTC issues. If this section does not describe the DTC issue you experience, you might want to review the following Microsoft Knowledge Base articles:

- http://support.microsoft.com/kb/306843

  This article describes how to troubleshoot DTC firewall issues.
- http://support.microsoft.com/kb/306212

  This article describes how to troubleshoot an error that occurs when you use a linked server in Microsoft SQL Server.
- http://support.microsoft.com/kb/293799

  This article describes how to use the DTCTester tool to test a distributed transaction against a SQL Server.

The following table lists common DTC issues and provides references to information about resolving the issues.

Issue	**Reference**
Server name resolution failure	http://support.microsoft.com/kb/169790
SID of one of the DTCs is not unique	http://support.microsoft.com/kb/294209
SQL Server system variable `@@servername` is incorrect or null	http://support.microsoft.com/kb/818334
Double hop error with Kerberos credentials	"Configuring Kerberos Delegation for a Distributed Control Center Environment" on page 75

Validating Deployment Services Installation

The Control Center Configuration Checker utility includes tests that validate proper configuration of the Deployment services. The following table describes the test categories.

Tests in this category...	**Check...**
Remote Deployment Service	Whether the Deployment Service is available and properly configured to remotely deploy agents and monitoring modules
Remote Deployment Web Service	Whether the Deployment Web Service is available and properly configured to remotely deploy agents and monitoring modules
Proxy Remote Deployment Web Service	If the Deployment Service must run in proxy mode to access the CCDB because of firewalls, proper SSL configuration

To validate Control Center and Deployment services installation:

1. Start the AppManager Control Center Configuration Checker in the AppManager program folder.
2. On the **Tasks** menu, select **Remote Deployment Setup**.
3. Provide the following information and click **OK**:
   - Name of the computer where you installed the Deployment Service.

     (Conditional) If the Deployment Service is on a remote computer, also provide credentials to connect to the computer.
   - Name of the computer where you installed the Deployment Web Service.

     (Conditional) If the Deployment Web Service is on a remote computer, also provide credentials to connect to the computer.
   - (Conditional) If the Deployment Service must run in proxy mode to access the CCDB, name of the computer where you installed the proxy Deployment Web Service and credentials to connect to the remote computer.
   - Name of the SQL Server and instance that hosts the CCDB.
   - Name of the CCDB.
4. Select the category that includes the tests you want to run and click **Run**.

   The utility runs each test included in the selected category. View the results of each test on the **Test Result** tab.

Changing the Deployment Service User Account

AppManager stores the Deployment Service account information you provided during installation in the Windows registry under `HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\AppManager\Control Center\1.0\CCDB`. The relevant keywords are `WindowsAuthName` and `WindowsAuthPass`. You can change the account information after installation.

To change the Windows user account for the Deployment Service:

1. Use Control Panel to stop the Deployment Service.
2. From a command prompt, change to the location of the `DeploymentService.exe.config` file and enter the following command:

   ```
   DeploymentService -setwindowsauth Domain\User_Name password
   ```

   For more information about the Deployment Service configuration file, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).
3. Start the Deployment Service.
4. Add the new account to Control Center.

   For information about adding accounts to Control Center, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

Changing the Deployment Web Service User Account

To change the Deployment Web Service Windows user account after installation, update the identity of the IIS application pool associated with the service.

To change the Deployment Web Service user account:

1. In the Application Pools settings in IIS Manager, expand **Application Pools**, right-click **AutoDeploymentAppPool**, and select **Properties**.
2. On the **Identity** tab, select **Configurable** and update the user name and password.
3. Add the user name to the `IIS_WPG` (worker process) group.
4. Stop and restart the application pool.
5. Stop and restart the `IISAdmin` service.
6. Add the new account to Control Center.

   For information about adding accounts to Control Center, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

8 Installing Agent Components

This chapter describes the steps for interactively installing agents and modules on computers running a Windows operating system. For information about installing agents and modules on computers running a UNIX or Linux operating system, see the *AppManager for UNIX and Linux Servers Management Guide*, available on the AppManager Modules Documentation page (http://www.netiq.com/documentation/appmanager-modules).

You can install agent components silently from a command prompt. For more information about silently installing components, see Appendix B, "Performing a Silent Installation," on page 99.

Understanding Windows Agent Installation

You can use the following methods to interactively install agents on computers running a Windows operating system:

- Run the setup program to install the agent locally.

  For more information about installing the agent locally, see "Installing the Windows Agent Locally" on page 88.
- Use Control Center to deploy agents to remote computers.

  For more information about installing Control Center components for deploying agents remotely, see Chapter 7, "Installing Control Center Components," on page 71. For more information about using Control Center to deploy agents remotely, see the *Control Center User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

When you run the setup program to install agents on Windows computers, you install a package that consists of the following components:

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

When you use Control Center to deploy agents to remote computers, the AppManager for Microsoft Windows module is not automatically deployed. You must also deploy the module to the computers where you deploy the agent.

After you install the agent, NetIQ Corporation recommends installing at least one module for an application you plan to monitor with the agent. For more information about installing modules, see "Installing Modules" on page 90.

You can install the Windows agent on Microsoft Cluster Service (MSCS). For more information about installing the agent on MSCS, see Appendix D, "Installing on Microsoft Cluster Service," on page 125.

Understanding Space Considerations

The setup program places the files it uses for agent installation in a `Temp` directory. To avoid a failed installation, ensure the `Temp` directory is on a drive with sufficient space for the installation. The `Temp` directory the setup program uses depends on the installation method you choose:

- If you run the setup program to install agents, the system `TEMP` environment variable on the local computer defines the `TEMP` directory.
- If you perform a silent or remote installation to install agents, the user `TEMP` environment variable on the target computer defines the `TEMP` directory.

Understanding Agent Reporting Capabilities

When you enable the agent reporting capability, report Knowledge Scripts collect monitoring data and generate reports. AppManager typically stores reports in the `\Program Files\NetIQ\Common\Report` folder.

Report agents can query QDBs, NetIQ Analysis Center repositories, or Microsoft Active Directory. For QDBs, the AppManager Layout engine, which the setup program installs when you select the reporting option, uses Microsoft ActiveX Data Objects (ADO) to connect to QDBs and execute SQL stored procedures to gather report data.

If you choose to enable the agent reporting capability, the agent services must run under a Windows user account. Enabling the reporting capability might result in a restart of Internet Information Services (IIS).

To optimize system resources for generating large reports, NetIQ Corporation recommends the following practices:

- Do not enable the reporting capability for the agent on the management server computer.
- Only enable one or two agents to generate reports. Install each report-enabled agent on a dedicated report server without any other core AppManager components installed.

If you want a report agent to query Active Directory, meet the following requirements:

- Install the report agent on a computer that is a member server of the domain.
- Install the QDB for the report agent on a member server of the same domain or a trusted domain.
- Ensure the NetIQ AppManager Client Resource Monitor (`NetIQmc`) service runs under a Windows user account that meets the following requirements:
    - The account must have at least the Read Only User role for the QDB.

      For more information about assigning AppManager roles, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).
    - The account must be an administrator on the report agent computer and a domain user.

For more information about using the standard reports AppManager provides, see the *Control Center User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

For more complex reporting, NetIQ Analysis Center extends AppManager reporting capabilities to provide more sophisticated data access and specialized report scripts. For more information about Analysis Center, see the *User Guide for NetIQ Analysis Center*, available on the Analysis Center Documentation page (https://www.netiq.com/documentation/analysis-center/).

Using Secure Communication

When you select a security level during agent installation, select the same security level you selected when you installed the QDB to which the agent will report. For more information about the options for securing communications between agents and management servers, see "Understanding QDB Security Options" on page 63.

Understanding MAPI Mail Settings

During agent installation, you have the option to enable MAPI mail to allow the agent to automatically send email messages in response to certain events as part of a Knowledge Script job. For example, you can configure the Action_MapiMail Knowledge Script to send email notifications to specific users when AppManager raises an event with a minimum severity level. For more information about the Action_MapiMail Knowledge Script, see the *AppManager Knowledge Script Reference Guide*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

If you want to enable MAPI mail during agent installation, complete the following actions before the installation:

- Install an Exchange client (for example, Microsoft Outlook) on the agent computer.
- Set up a Windows user account for the AppManager agent services to use.

  For more information about Windows user account requirements for the agent services, see "Reviewing Required Accounts and Permissions" on page 24.

  Both agent services must use the same account. When you specify the account information during agent installation, it automatically applies to both services. If you change the account after installation, change it for both services.
- Set up an Exchange mailbox for the agent services account.

During agent installation, provide the Exchange Server, Exchange client profile, and mailbox alias names.

Because Microsoft has tightened security in the most recent versions of Outlook, the NetiqMAPImail helper script only works with Outlook 2000 or Outlook 2003 with Service Pack 1. That service pack is required. This action does not run on Outlook 2003 without service packs or on Outlook 2003 with Service Pack 2.

Understanding the Agent Services Account

The agent services can run using either the Windows local system account or a Windows user account. In some situations, you must specify a Windows user account with the right to log on as a service. For more information about the account requirements, see "Reviewing Required Accounts and Permissions" on page 24.

Understanding Agent Automatic Discovery and Management Server Designation

To enable AppManager to automatically discover the computer on which you install the agent and run the Discovery_AMHealth Knowledge Script to prepare AppManager components for health monitoring in Control Center, install a management server and designate a primary management server for the agent. If you install the agent and the management server on the same computer, the primary management server must be the local management server.

To provide failover support, NetIQ Corporation recommends designating a primary and secondary management server for each agent. For more information about designating primary and secondary management servers, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

Once you successfully install the agent services and the AppManager for Microsoft Windows module, the setup program automatically discovers the agent and runs the Discovery_AMHealth Knowledge Script.

If the agent cannot communicate with either the primary or secondary management server, the setup program cannot automatically discover the agent and run the Discovery_AMHealth Knowledge Script. If automatic discovery fails, use the Control Center console to run the appropriate operating system Discovery Knowledge Script to discover the agent. After successful discovery, run the AMAdmin_SetPrimaryMS Knowledge Script to designate the primary and secondary management servers. After you designate the management servers, run the Discovery_AMHealth Knowledge Script. For more information about running the Knowledge Scripts, see the *AppManager Knowledge Script Reference Guide*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

For more information about using Control Center to monitor the health of your AppManager components, see the *Control Center User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

Installing the Windows Agent Locally

This section describes the steps required to install the Windows agent on the local computer.

To install the agent:

1 Ensure the agent services account is properly configured.

For more information about the account, see “Reviewing Required Accounts and Permissions” on page 24.

2 Complete the steps in “Understanding the AppManager Pre-Installation Check” on page 61.

After you view the pre-installation check report, click **Next** to start the installation.

3 When you reach the Report Generation Option window, select whether to enable the agent to generate reports and click **Next**.

4 Select the security level for communications between the agent and management servers and click **Next**.

Select the same security level you selected when you installed the QDB to which the agent reports.

5 (Conditional) If you selected **Encrypted communications only** or **Authentication and encrypted communications**, specify the password for the agent to access its portion of the QDB encryption key and click **Next**.

Specify the same password you specified when you installed the QDB to which the agent reports.

6. Specify the name that will identify the agent computer in the Control Center console and Operator Console.

   You can specify the agent computer name, IP address, DNS name, or hostname.

7. Specify the ports for communication between the agent and management servers.

   Specify the same ports you specified when you installed the management servers with which the agent will communicate.

8. Select whether to enable MAPI mail for the agent and click **Next**.

   If you select to enable MAPI mail, also provide the Exchange Server, mailbox alias, and Exchange client profile names you established before installation.

9. Provide information about the account the agent services will use and click **Next**.

10. (Optional) If you are installing the agent on the same computer as the management server and want to designate a secondary management server, specify the name of the secondary management server and click **Next**.

    The primary management server is the local management server.

11. (Conditional) If you are not installing the agent on the same computer as the management server, select whether to designate management servers during agent installation and click **Next**.

    If you select to designate management servers during agent installation, also specify the names of the primary and secondary management servers.

    To designate management servers after installation, run the AMAdmin_SetPrimaryMS Knowledge Script. For more information about running the Knowledge Script, see the *AppManager Knowledge Script Reference Guide*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

12. Review the installation settings. When you are ready to install the agent, click **Install**.

If you want to use Performance Monitor to monitor the operational health and performance of the agent, you must manually install performance counters. For more information about installing the counters, see “Installing Agent Performance Counters” on page 89.

Installing Agent Performance Counters

You can use Microsoft Performance Monitor to monitor the operational health and performance of the agent. Installing the agent does not automatically install the performance counters. You must manually install the counters.

To install 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"
   ```

If you installed the performance counters on a 64-bit operating system, to view the counters, you must open Performance Monitor from the `Windows\SysWOW64` folder.

Installing Agents Remotely

After you install a QDB, management server, agent, and Control Center components, you can use Control Center to install more agents to remote Windows computers. If you use Control Center to deploy agents to remote computers, you must also deploy the AppManager for Microsoft Windows module to those computers. The module is not automatically deployed with the agent.

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 installing agents to remote Windows computers, see the *Control Center User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager). For information about installing agents remotely on UNIX or Linux computers, see the *AppManager for UNIX and Linux Servers Management Guide*, available on the AppManager Modules Documentation page (http://www.netiq.com/documentation/appmanager-modules).

If you want to use Performance Monitor to monitor the operational health and performance of the agent, you must install the counters on the agent computer. For more information about installing the counters, see “Installing Agent Performance Counters” on page 89.

Installing Modules

A module is AppManager software that enables management of a particular third-party product. You install modules on the primary QDB computer, on Windows agent computers you want to monitor, computers monitoring by proxy, and on all Operator Console and Control Center console computers. Once you install the module on the primary QDB computer, Control Center automatically replicates the module to non-primary QDBs. For more information about installing modules on computers running Microsoft Windows operating systems, see the management guide for the applicable module.

For information about installing modules on computers running UNIX or Linux operating systems, see the *AppManager for UNIX and Linux Servers Management Guide*, available on the AppManager Modules Documentation page (http://www.netiq.com/documentation/appmanager-modules).

9 Installing the Operator Console and Console Programs

This chapter describes the steps for interactively installing the Operator Console and console programs.

You can install the Operator Console and console programs silently from a command prompt. For more information about silently installing components, see Appendix B, “Performing a Silent Installation,” on page 99.

If you monitor UNIX computers, you can also use the UNIX Agent Manager. For more information about the UNIX Agent Manager, see the *AppManager for UNIX and Linux Servers Management Guide*, available on the AppManager Modules Documentation page (http://www.netiq.com/documentation/appmanager-modules).

Understanding Operator Console Installation

When you install the Operator Console, you also have the option to install the following console programs:

- **Security Manager, a** utility that lets you identify users who are allowed to access AppManager, define user roles and rights, and maintain passwords and other secure information.

  For more information about Security Manager, see “Starting Security Manager the First Time” on page 92.
- **Developer’s Console**, a tool for editing Knowledge Scripts and developing custom Knowledge Scripts.

  For more information about the Developer’s Console, see the *Developing Custom Knowledge Scripts Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

Installing the Operator Console automatically installs the Chart Console.

Installing the Operator Console

This section describes the steps required to install the Operator Console.

To install the Operator Console:

1. Complete the steps in “Understanding the AppManager Pre-Installation Check” on page 61.

   After you view the pre-installation check report, click **Next** to start the installation.
2. When you reach the User Information window, provide your identification information and click **Next**.
3. Select the console programs to install and click **Next**.
4. Review the installation settings. When you are ready to install the Operator Console, click **Install**.

After you install the Operator Console, if you want to enable logging for the Chart Console, you must edit the Chart Console registry keys. For more information about editing the registry keys, see "Enabling Logging for the Chart Console" on page 92.

Enabling Logging for the Chart Console

The Chart Console log files, `AMChartCon.log` and `IExplorer.log`, are located in the `Users\`*`User_Name`*`\AppData\NetIQ\NetIQ_Debug\`*`Computer_Name`* folder. Chart Console logging is not automatically enabled. To enable logging, you must edit registry keys.

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 enable logging for the Chart Console:

1. Click **Start** > **Run**.
2. In the **Open** field, type `regedit` and then click **OK**.
3. In the left pane of the Registry Editor, navigate to `HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\Common\AMREPORTS`.
4. In the right pane, double-click **DebugHost**.
5. In the **Value data** field, set the value to **1** and then click **OK**.
6. In the left pane of the Registry Editor, navigate to `HKEY_LOCAL_MACHINE\SOFTWARE\NetIQ\Generic\Tracing`.
7. In the right pane, double-click **TraceConsole**.
8. In the **Value data** field, set the value to **1** and then click **OK**.

Starting Security Manager the First Time

Security Manager allows AppManager administrators to control access to views and tasks in the Operator Console and manages application or computer-specific security information, such as SNMP community strings and passwords. Depending on your access rights and your SQL Server security setting, you can use Security Manager to identify SQL Server users who have permission to use AppManager, add new SQL Server users, assign roles to AppManager users, and manage user rights. For more information about SQL Server security settings and accounts, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

To use Security Manager for the first time, you must have at least one SQL Server login or Windows user account with permission to access SQL Server and the QDB. You can use Microsoft SQL Server Management Studio to verify the authentication type and whether the account has permission to access SQL Server databases. Once you configure at least one Windows or SQL Server account for logging in to the QDB the first time, you can use Security Manager to grant other SQL Server login accounts access to AppManager.

To start Security Manager the first time:

1. Click **Start** > **Programs** > **NetIQ** > **AppManager** > **Tools & Utilities** > **Security Manager**.

2. Provide the following information and click **Logon**:
   - Name of the SQL Server that hosts the QDB for which you want to manage security
   - Name of the QDB for which you want to manage security
   - Whether to use a Windows account or a SQL Server account to log in to the QDB

For more information about using Security Manager to configure security for the QDB, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

A Updating License Information

This appendix describes how to view and update AppManager license information.

Understanding AppManager License Keys

NetIQ Corporation offers evaluation and production licenses for AppManager. An evaluation license allows you to install all AppManager components with full functionality and is time limited. For more information about installing AppManager for evaluation purposes, see the *Trial Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

When you purchase AppManager, you receive a permanent license key that replaces the evaluation key and identifies the components you purchased.

Because AppManager modules also require license keys, you will receive multiple keys to enter during AppManager installation. For example, if you purchase the base AppManager product and a 50-server license for monitoring Microsoft Windows, you receive two license keys: one for the base product and one for the AppManager for Microsoft Windows module. For more information about importing multiple license keys from a file, see “Importing License Keys from a File” on page 96.

If you purchase additional AppManager components after installation, use the AppManager License Manager to update the license information. For more information about updating license information after installation, see “Adding and Deleting a License Key” on page 96.

Starting License Manager

You can use the License Manager to view, add, import, delete, and request AppManager licenses. You can use the following methods to open the License Manager:

- From the **Start** menu, select **Programs** > **NetIQ Corporation** > **AppManager** **> Tools & Utilities > License Manager.**
- From the Operator Console or Security Manager, select **Help > License Manager**.

From Control Center, you can only view license information. On the **Global Tasks** tab, click **View Licenses**.

If you installed components with an evaluation license, the License Manager displays an expiration date. If you do not update the License Manager with permanent license keys, the components will not be accessible after the evaluation period.

Importing License Keys from a File

If you have multiple license keys to add or update, you can import the information from a text file you receive when you purchase AppManager or request new licenses. After you import the information, it automatically appears in the License Manager each time you install an AppManager component.

To import license keys from a text file:

1. Start the License Manager.
2. Click **Import**.
3. Locate the `license.txt` file you received and click **Open**.

   The License Manager imports the license key information.

Requesting License Keys

To request license keys, you can contact Sales or use the License Manager to send an email request.

To request licenses through email:

1. Start the License Manager.
2. Click **Request**.
3. To include information about your current licenses with your request, select **Send NetIQ licensing information from your repository.**
4. Provide your contact information.
5. Click **Request Licenses**.

   The License Manager sends your request to Sales Support. You will receive an email response to your request.

Adding and Deleting a License Key

When you convert from an evaluation copy or purchase additional AppManager components, use the License Manager to update the license information. If you purchase AppManager after the evaluation period, you do not need to re-install AppManager. You only need to add the permanent license key. Use the License Manager to add and delete license keys.

If you are updating license information for an AppManager Connector, stop and restart the AppManager Management Service (`NetIQms`) to enable the Connector.

Running a License Report

If you installed and discovered at least one report-enabled agent, you can run reports that provide information about the number of licensed and installed AppManager components associated with a specific QDB. The following license report Knowledge Scripts are available:

- CompLic

  This Knowledge Script generates a report that lists the number of AppManager licenses you have for a particular application and the number of computers on which you discovered that application.

- CompDeploy

  This Knowledge Script generates a report that details the total number of instances of each AppManager component installed on computers in an AppManager site (for example, the number of IIS and Exchange managed objects).

For more information about the Knowledge Scripts, see the *AppManager Knowledge Script Reference Guide*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

B Performing a Silent Installation

This appendix explains the steps for installing AppManager Windows components silently over a network from a command prompt. For information about silently installing the UNIX agent, see the *AppManager for UNIX and Linux Servers Management Guide*, available on the AppManager Modules Documentation page (http://www.netiq.com/documentation/appmanager-modules).

Understanding Silent Installation

Silent installation allows you to install AppManager components without having to interact with the setup program. From a command prompt, you instruct the Windows Installer package associated with a component to perform the installation. You can specify only the required parameters and default settings, or specify additional parameters and alternate settings.

Before you silently run the Windows Installer package for a component, you must complete the following tasks:

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

  For more information about installing the runtime libraries, see "Installing Runtime Libraries" on page 59.
- If the computer from which you will run the Windows Installer package has the User Access Control (UAC) feature enabled, ensure the computer is authorized to run the package.

  For more information about ensuring the computer is authorized, see "Running Windows Installer Packages When UAC is Enabled" on page 60.

Silently Installing the QDB

The procedure for silently installing the QDB differs from the procedure for silently installing other components. To silently install the QDB, you create an initialization file that records the variables to use during the installation, and then execute a command that reads the variables in the file to install the QDB.

You can silently install QDBs to remote SQL Servers. You do not have to run the installation on the SQL Server.

To silently install the QDB:

1 On the computer where you saved the AppManager installation files, copy the `qdbinstall.iss` file from the `Setup Files` folder to the folder in which you will create the initialization file.

2 In the folder where you copied `qdbinstall.iss`, create a response file named `silent.ini` with the appropriate parameters.

The following table describes the parameters available for silently installing the QDB. When you add the parameters to the `silent.ini` file, follow the order in the table.

Parameter	Description and Values
`RP_UPGRADE`	If the QDB specified for the `RP_NAME` parameter already exists, whether to upgrade the existing QDB Specify `FALSE` if you do not want to upgrade the existing QDB. Specify `TRUE` if you want to upgrade the existing QDB. If you do not specify the `RP_UPGRADE` parameter and the QDB already exists, the QDB will not be upgraded.
`RP_CLEARTEXTPASSWORD`	Indicates the `silent.ini` file will submit passwords to the Windows Installer package in clear text format Specify `1` to use clear text format for passwords. This parameter is required for successful installation. If you do not include this parameter, the Windows Installer package will attempt to decrypt passwords and you will receive the following error message: `Failed to decrypt the input data with FIPS compliant algorithms.`
`RP_SQLSERVER`	Name of the SQL Server and, if applicable, instance that will host the QDB To specify a SQL Server instance, use the format *Server_Name\instance*.
`RP_NAME`	Name of the QDB

Parameter	**Description and Values**
RP_WINDOWSAUTH	Whether to use the current Windows user account or a SQL Server user account to log in to the SQL Server to create the QDB Specify 1 to use the current Windows user account to log in to the SQL Server to create the QDB. Specify 0 to use a SQL Server user account to log in to the SQL Server to create the QDB. If you specify 0, specify values for the RP_SQLUSER and RP_SQLPWD parameters. For more information about the account, see "Reviewing Required Accounts and Permissions" on page 24.
RP_SQLUSER	User name for the SQL Server user account that will log in to the SQL Server to create the QDB
RP_SQLPWD	Password for the SQL Server user account that will log in to the SQL Server to create the QDB
RP_DATAPATH	Location of the data file associated with the QDB If you do not specify this parameter, the default value is the root folder for Microsoft SQL Server. For example, C:\Program Files\Microsoft SQL Server\MSSQL10\MSSQL\Data. Ensure the folder exists before you start the installation.
RP_DATASIZE	Initial size of the data file associated with the QDB The minimum required value is 100 MB. The default value is 2048 MB. If you do not specify this parameter, the default value is used. The autogrowth rate for the file is set to 256 MB. Allow at least enough free space to accommodate the autogrowth rate.
RP_DATANAME	Name of the data file associated with the QDB, without the file type extension. For example, QDBData.
RP_LOGPATH	Location of the log file associated with the QDB If you do not specify this parameter, the default value is the root folder for Microsoft SQL Server. For example, C:\Program Files\Microsoft SQL Server\MSSQL10\MSSQL\Data. Ensure the folder exists before you start the installation.
RP_LOGSIZE	Initial size of the log file associated with the QDB The minimum required value is 50 MB. The default value is 512 MB. If you do not specify this parameter, the default value is used. The autogrowth rate for the file is set to 256 MB. Allow at least enough free space to accommodate the autogrowth rate.
RP_LOGNAME	Name of the log file associated with the QDB, without the file type extension. For example, QDBLog.

Parameter	Description and Values
RP_WINREPOWNER	Whether the account that will own the QDB will be a Windows user account or a SQL Server user account Specify 1 to use a Windows user account. If you specify 1, specify a value for the RP_OWNER parameter. Specify 0 to use a SQL Server user account. If you specify 0, specify values for the RP_OWNER and RP_OWNERPWD parameters. For more information about the account, see “Reviewing Required Accounts and Permissions” on page 24.
RP_OWNER	User name for the account that will own the QDB If the account is a Windows user account, specify the name in *domain\User_Name* format.
RP_OWNERPWD	If the QDB owner is a SQL Server user account, password for the account
RP_ENC_CONFIG	Whether to configure security options for Windows agents, UNIX agents, or both Windows and UNIX agents Specify 1 to configure security for Windows agents only. Specify 2 to configure security for UNIX agents only. Specify 3 to configure security for both Windows and UNIX agents. For more information about configuring security, see “Understanding QDB Security Options” on page 63.
RP_ENC_LEVEL	Security level for communications between management servers and agents Specify 1 if you do not want to secure communications between management servers and agents. Specify 2 to encrypt communications between management servers and agents. Specify 3 to encrypt and authenticate communications between management servers and agents. If you specify security level 2 or 3, specify a value for the RP_SSLWINUNIX_PWD_QDB parameter. For more information about the security levels, see “Understanding QDB Security Options” on page 63.
RP_SSLWINUNIX_PWD_QDB	If you specified 2 or 3 for the RP_ENC_LEVEL parameter, password for the encryption key stored in the repository
RP_SSLWINUNIX_PWD_AGENT	If you specified 2 or 3 for the RP_ENC_LEVEL parameter, password for the agent to access its portion of the repository encryption key

Parameter	Description and Values
RP_SSLWINUNIX_IMPORTKEY	For the Windows agent, if you specified 2 or 3 for the RP_ENC_LEVEL parameter, whether to export the agent key information to a file For the UNIX agent, if you specified 3 for the RP_ENC_LEVEL parameter, whether to export the agent key information to a file Specify 1 if you do not want to export the agent key information to a file. Specify 0 to export the agent key information to a file.
RP_SSLWINUNIX_KEYPATH	If you specified 1 for the RP_SSLWINUNIX_IMPORTKEY parameter, location and file to which you want to export the agent key information The default path is C:\Program Files\NetIQ\AppManager \nqWindowsPublic0.key.

3 Run the following command from the folder in which you saved the QDB setup program:

```
"NetIQ AppManager Repository Installation.exe" /s
/f1"Full_Path_to_qdbinstall.iss_File"
-silentinstall="Full_Path_to_silent.ini_File"
```

Ensure that you do not include a space between /f1 and "*Full_Path_to_qdbinstall.iss_File*". The installation will fail if you include a space.

For more information about installing the QDB, see Chapter 6, "Installing a Management Site," on page 63.

Silently Installing the Task Scheduler Service

To silently install the Task Scheduler service using the required parameters and default settings, run the following command from the folder in which you saved the Task Scheduler service setup program:

```
msiexec.exe /i "NetIQ AppManager Task Scheduler Service.msi"
INSTALLDIR="Path_to_the_Installation_Folder" NQAMTS_B_WINUSER=1
NQAMTS_WINDOMAINUSER=
"domain\User_Name_for_the_Windows_User_Account_under_which_the_Service_
Will_Run"
NQAMTS_WINPWD="Password_for_the_Windows_User_Account_under_which_the_Service_
Will_Run" /l*v Path_to_the_Installation_Log /qn
```

To specify additional parameters or alternate settings, add the appropriate parameters to the command. The following table describes all parameters available for silently installing the Task Scheduler service.

Parameter	Description and Values
INSTALLDIR	Target installation folder The default value is C:\Program Files\NetIQ\.

Parameter	Description and Values
NQAMTS_B_WINUSER	Whether the service will use a Windows user account or the local system account Specify 1 if the service will use a Windows user account. If you specify 1, specify a value for the NQAMTS_WINDOMAINUSER and NQAMTS_WINPWD parameters. Specify 0 if the service will use the local system account. For more information about the account, see "Reviewing Required Accounts and Permissions" on page 24.
NQAMTS_WINDOMAINUSER	If the service will use a Windows user account, domain and user name for the account
NQAMTS_WINPWD	If the service will use a Windows user account, password for the account

For more information about installing the Task Scheduler service, see Chapter 6, "Installing a Management Site," on page 63.

Silently Installing the Management Server

To silently install the management server using the required parameters and default settings, run the following command from the folder in which you saved the management server setup program:

```
msiexec.exe /i "NetIQ AppManager management server.msi" /qn
INSTALLDIR="C:\Program Files\NetIQ\" MS_B_WINUSER=1 MS_WINDOMAINUSER=
"domain\User_Name_for_the_Windows_User_Account_under_which_the_NetIQms_Service_
Will_Run"
MS_WINPWD="Password_for_the_Windows_User_Account_under_which_the_NetIQms_Service_
Will_Run"
MS_RPSERVERNAME="Name_and_Instance_of_the_SQL_Server_that_Hosts_the_QDB_to_which_
the_Management_Server_Will_Connect"
MS_RPNAME="Name_of_the_QDB_to_which_the_Management_Server_Will_Connect"
MS_B_SQLAUTHMODE=0
```

To specify additional parameters or alternate settings, add the appropriate parameters to the command. The following table describes all parameters available for silently installing the management server.

Parameter	Description and Values
INSTALLDIR	Target installation folder The default value is C:\Program Files\NetIQ\.
MS_PORT	RPC port where the management server listens for communications from Windows agents If you do not specify this parameter, the default value 9999 is used. For more information about the ports the management server and agents use, see "Reviewing Management Server Port Information" on page 68.

Parameter	Description and Values
MC_PORT	RPC port where Windows agents listen for communications from the management server If you do not specify this parameter, the default value 9998 is used. For more information about the ports the management server and agents use, see "Reviewing Management Server Port Information" on page 68.
MS_PORTUNIX	RPC port where the management server listens for communications from UNIX agents If you do not specify this parameter, the default value 9001 is used. For more information about the ports the management server and agents use, see "Reviewing Management Server Port Information" on page 68.
MS_B_WINUSER	Whether the NetIQms service will use a Windows user account or the local system account Specify 1 if the service will use a Windows user account. If you specify 1, specify a value for the MS_WINDOMAINUSER and MS_WINPWD or MS_WINPWDE parameters. Specify 0 if the service will use the local system account. For more information about the account, see "Reviewing Required Accounts and Permissions" on page 24.
MS_WINDOMAINUSER	If the NetIQms service will use a Windows user account, domain and user name for the account
MS_WINPWD	If the NetIQms service will use a Windows user account and your site does not use Federal Information Processing Standards (FIPS), password for the account
MS_WINPWDE	If the NetIQms service will use a Windows user account and your site uses FIPS, the encrypted password for the account
MS_RPSERVERNAME	Name of the SQL Server and, if applicable, instance that hosts the QDB to which the management server will connect To specify a SQL Server instance, use the format *Server_Name\instance*.
MS_RPNAME	Name of the QDB to which the management server will connect
MS_B_SQLAUTHMODE	Whether the management server will use a SQL Server user account or the same Windows user account as the NetIQms service to connect to the QDB Specify 1 if the management server will use a SQL Server user account to connect to the QDB. If you specify 1, also specify values for the MS_RP_OWNER and MS_RP_PASSWD parameters. Specify 0 if the management server will use the same Windows user account as the NetIQms service to connect to the QDB. For more information about the account, see "Reviewing Required Accounts and Permissions" on page 24.
MS_RP_OWNER	If the management server will use a SQL Server user account to connect to the QDB, user name for the account

Parameter	Description and Values
`MS_RP_PASSWD`	If the management server will use a SQL Server user account to connect to the QDB, password for the account

For more information about installing the management server, see Chapter 6, “Installing a Management Site,” on page 63.

Silently Installing the Windows Agent

For information about silently installing the UNIX agent, see the *AppManager for UNIX and Linux Servers Management Guide*, available on the AppManager Modules Documentation page (http://www.netiq.com/documentation/appmanager-modules).

To silently install the Windows agent using the required parameters and default settings, run the following command from the folder where you saved the agent setup program:

```
msiexec.exe /i "NetIQ AppManager agent.msi" /qn
INSTALLDIR="C:\Program Files\NetIQ\" MC_B_REPORTAGENT=0 MC_SECLEVEL=0 MC_B_PORT=0
MC_B_MAPI=0 MC_B_WINUSER=1
MC_WINDOMAINUSER="Domain\User_Name_for_the_Windows_User_Account_under_which_the_
Agent_Services_Will_Run"
MC_WINPWD="Password_for_the_Windows_User_Account_under_which_the_Agent_Services_
Will_Run" MC_B_ONMS=0|1
```

To specify additional parameters or alternate settings, add the appropriate parameters to the command. The following table describes all parameters available for silently installing the Windows agent.

Option	Description
`MC_B_UPGRADE`	Whether to install a new agent or upgrade an existing agent Specify `1` to upgrade an existing agent. Specify `0` to install a new agent. If you specify 1, also specify the `REINSTALLMODE` and `REINSTALL` options.
`REINSTALLMODE`	If you specified to upgrade an existing agent, the type of reinstall to perform Specify `vomus`.
`REINSTALL`	If you specified to upgrade an existing agent, the features to be reinstalled Specify `ALL`.
`INSTALLDIR`	Target installation folder The default value is `C:\Program Files\NetIQ\`.

Option	Description
MC_B_REPORTAGENT	Whether to enable the agent to generate reports. Specify 1 to enable the agent to generate reports. Specify 0 if you do not want to enable the agent to generate reports. For more information about the agent reporting capability, see "Understanding Agent Reporting Capabilities" on page 86.
MC_SECLEVEL	Security level to use for communication between the agent and management servers Specify 0 if you do not want to secure communication between the agent and management servers. Specify 1 to encrypt communication between the agent and management servers. Specify 2 to encrypt and authenticate communication between the agent and management servers. For more information about securing communication between the agent and management servers, see "Understanding QDB Security Options" on page 63.
MC_SECPWD	If you specified to encrypt communication or to encrypt and authenticate communication between the agent and management servers, the password for the agent to access its portion of the QDB encryption key Specify the same password you specified when you installed the QDB to which the agent reports.
MC_B_DISPLAYNAME	Whether to change the display name for the agent computer Specify 1 to change the display name for the agent computer. If you specify 1, also specify a value for the MC_DISPLAYNAME parameter. Specify 0 to use the agent computer name as the display name. Use the MC_DISPLAYNAME parameter to specify the display name.
MC_DISPLAYNAME	If you specified to change the display name for the agent computer, name to display for the agent computer in the Control Center console and Operator Console
MC_B_PORT	Whether to change the default RPC ports where the management server and the agent listen for communications from each other Specify 1 to change the ports. Specify 0 to use the default ports. If you specify 1, also specify values for the MS_PORT and MC_PORT parameters. For more information about the ports the management server and agents use, see "Reviewing Management Server Port Information" on page 68.

Option	Description
MS_PORT	If you specified to change the default RPC ports where the management server and the agent listen for communications from each other, port where the management server will listen for communications from the agent
MC_PORT	If you specified to change the default RPC ports where the management server and the agent listen for communications from each other, port where the agent will listen for communications from the management server
MC_B_MAPI	Whether to enable MAPI mail to allow the agent to automatically send email messages in response to certain events Specify 1 to enable the MAPI mail option. If you specify 1, also specify values for the MC_EXCHSVR, MC_MAILBOX, and MC_PROFILE parameters. Specify 0 if you do not want to enable the MAPI mail option. For more information about the MAPI mail option, see “Understanding MAPI Mail Settings” on page 87.
MC_EXCHSVR	If you specified to enable MAPI mail, name of the Microsoft Exchange Server the MAPI mail client will use
MC_MAILBOX	If you specified to enable MAPI mail, Exchange mailbox alias name for the agent service account to use
MC_PROFILE	If you specified to enable MAPI mail, Exchange client profile name
MC_B_WINUSER	Whether the agent services will use a Windows user account or the local system account Specify 1 if the services will use a Windows user account. If you specify 1, also specify values for the MC_WINDOMAINUSER and MC_WINPWD or MC_WINPWDE parameters. Specify 0 if the services will use the local system account. In some cases, the services must use a Windows user account. For more information about account requirements for the agent services, see “Reviewing Required Accounts and Permissions” on page 24.
MC_WINDOMAINUSER	If the agent services will use a Windows user account, the domain and user name for the account
MC_WINPWD	If the agent services will use a Windows user account and your site does not use FIPS, the password for the account
MC_WINPWDE	If the agent services will use a Windows user account and your site uses FIPS, the encrypted password for the account

Option	Description
MC_B_ONMS	Whether the management server is present on the agent computer Specify 1 if the management server is present on the agent computer. Specify 0 if the management server is not present on the agent computer. If the management server is present on the agent computer, it must also be the primary management server for the agent. AppManager automatically selects the local management server as the primary management server.
MC_B_MSPRISEC	If you are not installing the agent on the same computer as the management server, whether to designate the primary and, optionally, secondary management server during agent installation Specify 1 to designate the primary and, optionally, secondary management server during agent installation. If you specify 1, use the MC_MSPRIMARY and MC_MSSECONDARY parameters to specify the management server names. Specify 0 to designate the primary and secondary management servers later using the AMAdmin_SetPrimaryMS Knowledge Script. NetIQ Corporation recommends designating at least the primary management server during agent installation. For more information about designating management servers, see “Understanding Agent Automatic Discovery and Management Server Designation” on page 88.
MC_MSPRIMARY	If you are not installing the agent on the same computer as the management server, name of the primary management server
MC_MSSECONDARY	Name of the secondary management server
MC_ALLOWMS	Whether to allow an anonymous management server to communicate with the agent Specify * to allow an anonymous management server to communicate with the agent. If you are installing the agent on the local computer, specify the names of the primary and secondary management servers to only allow those management servers to communicate with the agent. If you are installing the agent on a remote computer, specify ; if you do not want to allow an anonymous management server to communicate with the agent. If you are installing the agent on a remote computer, specify *; to allow an anonymous management server to communicate with the agent until you designate primary and secondary management servers.

Option	Description
MC_B_MSPRISEC_ REMOVEALLOWMSSTAR	Removes authorization for all management servers to communicate with the agent computer during agent installation. The AllowMS registry key stores the list of management servers that are authorized to communicate with an agent computer. The key uses an asterisk (*) to allow all management severs to communicate with an agent computer if you do not designate a primary management server. If you designate the primary management server during agent installation, the MC_B_MSPRISEC_REMOVEALLOWMSSTAR parameter removes the asterisk from the AllowMS registry key. If you do not designate the primary management server during agent installation, the parameter indicates that the AllowMS registry key should not change until you run the SetPrimaryMS Knowledge Script.
MC_B_MSPRISEC_REMOVESTAR	Whether to allow an anonymous management server to exchange reports with the agent Specify 1 if you do not want to allow an anonymous management server to exchange reports with the agent. Specify 0 to allow an anonymous management server to exchange reports with the agent.
MC_B_MSPRIMARY_EXIST	Whether the primary management server is available Specify 1 if the primary management server is available. Specify 0 if the primary management server is not available.
MC_B_MSSECONDARY_EXIST	Whether the secondary management server is available Specify 1 if the secondary management server is available. Specify 0 if the secondary management server is not available.
MC_B_PROXY	Installs the agent as a proxy Specify 1 to install the agent as a proxy.
MC_MDBPATH	Target installation folder for the local repository

For more information about installing the agent, see Chapter 8, “Installing Agent Components,” on page 85.

Silently Installing Modules

For information about silently installing a module, see the management guide for the module.

Silently Installing the CCDB

The procedure for silently installing the CCDB differs from the procedure for silently installing other components. To silently install the CCDB, you create an initialization file that records the variables to use during the installation, and then execute a command that reads the variables in the file to install the CCDB.

You can silently install CCDBs to remote SQL Servers. You do not have to run the installation on the SQL Server.

To silently install the CCDB:

1. On the computer where you saved the AppManager installation files, copy the `ccdbinstall.iss` file from the `Setup Files` folder to the folder in which you will create the initialization file.
2. In the folder where you copied `ccdbinstall.iss`, create a response file named `silent.ini` with the appropriate parameters.

   The following table describes the parameters available for silently installing the CCDB. When you add the parameters to the `silent.ini` file, follow the order in the table.

Parameter	Description and Values
`RP_UPGRADE`	If the CCDB specified for the `RP_NAME` parameter already exists, whether to upgrade the existing CCDB Specify `FALSE` if you do not want to upgrade the existing CCDB. Specify `TRUE` if you want to upgrade the existing CCDB. If you do not specify the `RP_UPGRADE` parameter and the CCDB already exists, the CCDB will not be upgraded.
`RP_CLEARTEXTPASSWORD`	Indicates the `silent.ini` file will submit passwords to the Windows Installer package in clear text format Specify `1` to use clear text format for passwords. This parameter is required for successful installation. If you do not include this parameter, the Windows Installer package will attempt to decrypt passwords and you will receive the following error message: `Failed to decrypt the input data with FIPS compliant algorithms.`
`RP_SQLSERVER`	Name of the SQL Server and, if applicable, instance that will host the CCDB To specify a SQL Server instance, use the format *Server_Name\instance*.
`RP_NAME`	Name of the CCDB

Parameter	Description and Values
RP_WINDOWSAUTH	Whether to use the current Windows user account or a SQL Server user account to log in to the SQL Server to create the CCDB Specify 1 to use the current Windows user account to log in to the SQL Server to create the CCDB. Specify 0 to use a SQL Server user account to log in to the SQL Server to create the CCDB. If you specify 0, specify values for the RP_SQLUSER and RP_SQLPWD parameters. For more information about the account, see "Reviewing Required Accounts and Permissions" on page 24.
RP_SQLUSER	User name for the SQL Server user account that will log in to the SQL Server to create the CCDB
RP_SQLPWD	Password for the SQL Server user account that will log in to the SQL Server to create the CCDB
RP_DATAPATH	Location of the data file associated with the CCDB If you do not specify this parameter, the default value is the root folder for Microsoft SQL Server. For example, C:\Program Files\Microsoft SQL Server\MSSQL10\MSSQL\Data. Ensure the folder exists before you start the installation.
RP_DATASIZE	Initial size of the data file associated with the CCDB The minimum required value is 512 MB. The default value is 1024 MB. If you do not specify this parameter, the default value is used. The autogrowth rate for the file is set to 256 MB. Allow at least enough free space to accommodate the autogrowth rate.
RP_DATANAME	Name of the data file associated with the CCDB, without the file type extension. For example, NQCCDBData.
RP_LOGPATH	Location of the log file associated with the CCDB If you do not specify this parameter, the default value is the root folder for Microsoft SQL Server. For example, C:\Program Files\Microsoft SQL Server\MSSQL10\MSSQL\Data. Ensure the folder exists before you start the installation.
RP_LOGNAME	Name of the log file associated with the CCDB, without the file type extension. For example, NQCCDBLog.
RP_LOGSIZE	Initial size of the log file associated with the CCDB The minimum required value is 50 MB. The default value is 512 MB. If you do not specify this parameter, the default value is used. The autogrowth rate for the file is set to 256 MB. Allow at least enough free space to accommodate the autogrowth rate.

Parameter	Description and Values
RP_WINREPOWNER	Whether the account that will own the CCDB will be a Windows user account or a SQL Server user account Specify 1 to use a Windows user account. If you specify 1, specify a value for the RP_OWNER parameter. Specify 0 to use a SQL Server user account. If you specify 0, specify values for the RP_OWNER and RP_OWNERPWD parameters. For more information about the account, see "Reviewing Required Accounts and Permissions" on page 24.
RP_OWNER	User name for the account that will own the CCDB. If the account is a Windows user account, specify the name in *domain\User_Name* format.
RP_OWNERPWD	If the CCDB owner is a SQL Server user account, password for the account.
RP_CQSUSERDOMAIN	Domain and user name for the Windows user account the command queue service will use to connect to the CCDB For more information about the account, see "Reviewing Required Accounts and Permissions" on page 24.
RP_ADSUSERDOMAIN	Domain and user name for the Windows user account the Deployment Service will use to connect to the CCDB For more information about the account, see "Reviewing Required Accounts and Permissions" on page 24.

3 Run the following command from the folder where you saved the CCDB setup program:

```
"Full_Path_to_Setup_Files_Folder\
NetIQ Control Center Repository Installation.exe" /s
/f1"Full_Path_to_ccdbinstall.iss_File"
-silentinstall="Full_Path_to_silent.ini_File"
```

Ensure that you do not include a space between /f1 and "*Full_Path_to_ccdbinstall.iss_File*". The installation will fail if you include a space.

For more information about installing the CCDB, see Chapter 7, "Installing Control Center Components," on page 71.

Silently Installing the Command Queue Service

To silently install the command queue service using the required parameters and default settings, run the following command from the folder where you saved the Control Center setup program:

```
"NetIQCCSetup.exe" /s /v"INSTALLDIR=\"C:\Program Files\NetIQ\" CC_CQS=1
CQS_DOMAIN="Domain_for_the_Windows_User_Account_the_Command_Queue_Service_
Will_Use" RP_CLEARTEXTPASSWORD=1
CQS_USER="User_Name_for_the_Windows_User_Account_the_Command_Queue_Service_
Will_Use"
CQS_PWD="Password_for_the_Windows_User_Account_the_Command_Queue_Service_Will_Use"
CCDB_SQLSERVERINSTANCE="Name_of_the_SQL_Server_and_Instance_that_Will_Host_the_
CCDB_to_which_the_Command_Queue_Service_Will_Connect"
CCDB_REPOSITORYNAME="Name_of_the_CCDB_to_which_the_Command_Queue_Service_Will_
Connect""
```

To specify additional parameters or alternate settings, add the appropriate parameters to the command. The following table describes all parameters available for silently installing the command queue service.

Parameter	Description
`INSTALLDIR`	Target installation folder The default value is `C:\Program Files\NetIQ\`.
`CC_CQS`	Installs the command queue service Specify `1` to install the command queue service.
`RP_CLEARTEXTPASSWORD`	Indicates the silent installation command will submit passwords to the Windows Installer package in clear text format Specify `1` to use clear text format for passwords. This parameter is required for successful installation. If you do not include this parameter, the Windows Installer package will attempt to decrypt passwords and you will receive the following error message: `Failed to decrypt the input data with FIPS compliant algorithms.`
`CQS_DOMAIN`	Domain for the Windows user account the command queue service will use to connect to the CCDB For more information about the account, see "Reviewing Required Accounts and Permissions" on page 24.
`CQS_USER`	User name for the Windows user account the command queue service will use to connect to the CCDB
`CQS_PWD`	Password for the Windows user account the command queue service will use to connect to the CCDB
`CCDB_SQLSERVERINSTANCE`	Name of the SQL Server and, if applicable, instance that will host the CCDB to which the command queue service will connect To specify a SQL Server instance, use the format *Server_Name\instance*. Only connect one command queue service to a CCDB.

Parameter	Description
CCDB_REPOSITORYNAME	Name of the CCDB to which the command queue service will connect
/L* "*Full_Path_to_Installation_Log_File*"	Creates a log file that describes the operations of the setup program For more information about parameters that specify the information included in the installation log file, see "Specifying Control Center Installation Log File Options" on page 118.

For more information about installing the command queue service, see Chapter 7, "Installing Control Center Components," on page 71.

Silently Installing the Deployment Web Service

To silently install the Deployment Web Service using the required parameters and default settings, run the following command from the folder where you saved the Control Center setup program:

```
"NetIQCCSetup.exe" /s /v"INSTALLDIR=\"C:\Program Files\NetIQ\" CC_AD_WEBSERVICE=1
RP_CLEARTEXTPASSWORD=1 CHECKIN_AD_PACKAGES_NOW=1
ADW_WINDOMAIN="Domain_for_the_Windows_User_Account_the_Deployment_Web_Service_
Will_Use"
ADW_WINUSER="User_Name_for_the_Windows_User_Account_the_Deployment_Web_Service_
Will_Use"
ADW_WINPWD="Password_for_the_Windows_User_Account_the_Deployment_Web_Service_
Will_Use"
ADW_DBSERVER="Name_of_the_SQL_Server_and_Instance_that_Will_Host_the_CCDB_to_which
_the_Deployment_Web_Service_Will_Connect"
CCDB_REPOSITORYNAME="Name_of_the_CCDB_to_which_the_Deployment_Web_Service_
Will_Connect""
```

To specify additional parameters or alternate settings, add the appropriate parameters to the command. The following table describes all parameters available for silently installing the Deployment Web Service.

Parameter	Description
INSTALLDIR	Target installation folder The default value is `C:\Program Files\NetIQ\`.
CC_AD_WEBSERVICE	Installs the Deployment Web Service Specify 1 to install the Deployment Web Service.
RP_CLEARTEXTPASSWORD	Indicates the silent installation command will submit passwords to the Windows Installer package in clear text format Specify 1 to use clear text format for passwords. This parameter is required for successful installation. If you do not include this parameter, the Windows Installer package will attempt to decrypt passwords and you will receive the following error message: `Failed to decrypt the input data with FIPS compliant algorithms.`

Parameter	Description
`CHECKIN_AD_PACKAGES_NOW`	Whether to check in agent and module packages and a set of default rules for use in remote deployment Specify `1` to check in packages and rules during Deployment Web Service installation. Specify `0` to check in packages and rules later. For more information about package and deployment rule check-in, see "Installing Control Center and Deployment Services" on page 78.
`ADW_WINDOMAIN`	Domain for the Windows user account the Deployment Web Service will use to connect to the CCDB For more information about the account, see "Reviewing Required Accounts and Permissions" on page 24.
`ADW_WINUSER`	User name for the Windows user account the Deployment Web Service will use to connect to the CCDB
`ADW_WINPWD`	Password for the Windows user account the Deployment Web Service will use to connect to the CCDB
`ADW_DBSERVER`	Name of the SQL Server and, if applicable, instance that will host the CCDB to which the Deployment Web Service will connect To specify a SQL Server instance, use the format *Server_Name\instance*. Only connect one Deployment Web Service to a CCDB.
`CCDB_REPOSITORYNAME`	Name of the CCDB to which the Deployment Web Service will connect
`/L* "`*`Full_Path_to_Installation_Log_File`*`"`	Creates a log file that describes the operations of the setup program For more information about parameters that specify the information included in the installation log file, see "Specifying Control Center Installation Log File Options" on page 118.

For more information about installing the Deployment Web Service, see Chapter 7, "Installing Control Center Components," on page 71.

Silently Installing the Deployment Service

To silently install the Deployment Service using the required parameters and default settings, run the following command from the folder where you saved the Control Center setup program:

```
"NetIQCCSetup.exe" /s /v"INSTALLDIR=\"C:\Program Files\NetIQ\" CC_AD_SERVICE=1
RP_CLEARTEXTPASSWORD=1
ADS_SQLSERVERINSTANCE="Name_of_the_SQL_Server_and_Instance_that_Will_Host_the_CCDB
_to_which_the_Deployment_Service_Will_Connect"
CCDB_REPOSITORYNAME="Name_of_the_CCDB_to_which_the_Deployment_Service_
Will_Connect"
ADS_WINDOMAIN="Domain_for_the_Windows_User_Account_the_Deployment_Service_
Will_Use"
ADS_WINUSER="User_Name_Associated_with_the_Windows_User_Account_the_Deployment_
Service_Will_Use"
ADS_WINPWD="Password_Associated_with_the_Windows_User_Account_the_Deployment_
Service_Will_Use""
```

To specify additional parameters or alternate settings, add the appropriate parameters to the command. The following table describes all parameters available for silently installing the Deployment Service.

Parameter	Description
`INSTALLDIR`	Target installation folder The default value is `C:\Program Files\NetIQ\`.
`CC_AD_SERVICE`	Installs the Deployment Service Specify `1` to install the Deployment Service.
`RP_CLEARTEXTPASSWORD`	Indicates the silent installation command will submit passwords to the Windows Installer package in clear text format Specify `1` to use clear text format for passwords. This parameter is required for successful installation. If you do not include this parameter, the Windows Installer package will attempt to decrypt passwords and you will receive the following error message: `Failed to decrypt the input data with FIPS compliant algorithms.`
`ADS_SQLSERVERINSTANCE`	Name of the SQL Server and, if applicable, instance that will host the CCDB to which the Deployment Service will connect To specify a SQL Server instance, use the format *Server_Name\instance*.
`CCDB_REPOSITORYNAME`	Name of the CCDB to which the Deployment Service will connect
`ADS_WINDOMAIN`	Domain for the Windows user account the Deployment Service will use to connect to the CCDB For more information about the account, see "Reviewing Required Accounts and Permissions" on page 24.
`ADS_WINUSER`	User name for the Windows user account the Deployment Service will use to connect to the CCDB
`ADS_WINPWD`	Password for the Windows user account the Deployment Service will use to connect to the CCDB
`ADS_B_FIREWALL`	Whether a firewall is active between the deployment server and the CCDB Specify `1` if a firewall is active. If you specify `1`, also specify a value for the `ADS_WEBSERVICESERVER` parameter.
`ADS_WEBSERVICESERVER`	If a firewall is active between the deployment server and the CCDB, name of the web server where the Deployment Web Service is installed When a firewall is active between the deployment server and the CCDB, the Deployment Web Service can provide a communication proxy to the CCDB so that you do not have to open additional ports to allow direct communication between the Deployment Service and the CCDB.

Parameter	Description
/L* "*Full_Path_to_Installation_Log_File*"	Creates a log file that describes the operations of the setup program For more information about parameters that specify the information included in the installation log file, see "Specifying Control Center Installation Log File Options" on page 118.

For more information about installing the Deployment Service, see Chapter 7, "Installing Control Center Components," on page 71.

Specifying Control Center Installation Log File Options

When you install Control Center and Deployment services, you can add parameters to specify a path to the installation log file and define the information to include in the log file. Add the parameters to the silent installation commands for the command queue service, Deployment Service, and Deployment Web Service.

The following table describes the parameters available for defining the information to include in the installation log file

Parameter	Description
/L	Indicates that you will specify a path to the installation log file
i	Logs status messages
w	Logs warning messages
e	Logs error messages
a	Logs the commencement of action sequences
r	Logs action-specific records
u	Logs user requests
c	Logs initial user interface parameters
m	Logs out-of-memory messages
p	Logs terminal settings
v	Indicates to use the verbose output format
+	Appends data to an existing file
*	Logs all information, excluding verbose output

Silently Installing the Control Center Console

You can silently install the Control Center console using only the required parameters and default settings, or you can specify additional parameters and alternate settings.

To silently install the Control Center console using the required parameters and default settings, run the following command from the folder where you saved the Control Center console setup program:

```
msiexec.exe /i "NetIQ AppManager Control Center Console.msi" /qn
INSTALLDIR="C:\Program Files\NetIQ\"
```

To specify additional parameters and alternate settings, add the appropriate parameters to the command. The following table describes all parameters available for silently installing the Control Center console.

Parameter	Description
`INSTALLDIR`	Target installation folder The default value is `C:\Program Files\NetIQ\`.
`USERNAME`	User name under which you want to register AppManager
`COMPANYNAME`	Company name under which you want to register AppManager

For more information about installing the Control Center console, see Chapter 7, "Installing Control Center Components," on page 71.

Silently Installing the Operator Console and Console Programs

You can silently install the Operator Console and console programs using only the required parameters and default settings, or you can specify additional parameters and alternate settings.

To silently install the Operator Console and console programs using the required parameters and default settings, run the following command from the folder where you saved the Operator Console setup program:

```
msiexec.exe /i "NetIQ AppManager Console Installation.msi" /qn
INSTALLDIR="C:\Program Files\NetIQ\"
```

To specify additional parameters and alternate settings, add the appropriate parameters to the command. The following table describes all parameters available for silently installing the Operator Console and console programs.

Parameter	Description
`INSTALLDIR`	Target installation folder The default value is `C:\Program Files\NetIQ\`.
`USERNAME`	User name under which you want to register AppManager
`COMPANYNAME`	Company name under which you want to register AppManager
`UI_SDK`	Installs SDK files Specify `1` to install SDK files.
`UI_WIN32`	Installs Win32 files Specify `1` to install Win32 files.
`UI_SECMGR`	Installs Security Manager Specify `1` to install Security Manager.

For more information about installing the Operator Console and console programs, see Chapter 9, “Installing the Operator Console and Console Programs,” on page 91.

C Uninstalling AppManager

This appendix explains how to uninstall AppManager Windows components. For information about how to uninstall the UNIX agent, see the *AppManager for UNIX and Linux Servers Management Guide*, available on the AppManager Modules Documentation page (http://www.netiq.com/documentation/appmanager-modules).

Understanding AppManager Uninstallation

You must uninstall components before you can reinstall or repair them. You do not have to uninstall components before you upgrade them. For more information about upgrading components, see the *Upgrade and Migration Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

To uninstall any Windows component except the QDB or CCDB, use the Add or Remove Programs application in Control Panel. You must manually uninstall all components except Windows agents and modules from the local computer. You can remotely uninstall Windows agents and modules across a network.

You must also manually uninstall the runtime support packages that were installed when you installed AppManager. Use the Add or Remove Programs application in Control Panel to uninstall the runtime support packages.

To uninstall the QDB or CCDB, use Microsoft SQL Server Management Studio to delete the database. When you delete the QDB or CCDB, you permanently remove the component and all stored data and jobs. When you upgrade a QDB or CCDB instead of uninstalling it, you can retain management data. For more information about upgrading a QDB or CCDB, see the *Upgrade and Migration Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

If AppManager components reside on multiple computers, before you uninstall any component, stop the NetIQ AppManager Management Service (`NetIQms`) and disconnect any users and applications that are accessing the QDB.

NetIQ Corporation recommends that you uninstall AppManager components in the following order:

1. Modules

   The order in which you uninstall individual modules is not significant. NetIQ Corporation recommends that you use Control Center to remotely uninstall Windows modules and UNIX Agent Manager to uninstall UNIX modules. For more information about remotely uninstalling Windows modules, see "Uninstalling Windows Agents and Modules Remotely" on page 122. For more information about uninstalling UNIX modules, see the *AppManager for UNIX and Linux Servers Management Guide*, available on the AppManager Modules Documentation page (http://www.netiq.com/documentation/appmanager-modules).

   The AppManager ResponseTime for Networks module installs the NetIQ Performance Endpoint software as a separate component. Uninstalling the module does not remove the endpoint software. To remove the endpoint software, uninstall it separately.

2. Operator Console
3. Management server

4. Windows or UNIX agent

   NetIQ Corporation recommends that you use Control Center to remotely uninstall Windows agents. For more information about remotely uninstalling Windows agents, see "Uninstalling Windows Agents and Modules Remotely" on page 122.

   To uninstall the UNIX agent, use the Patch Manager in the UNIX Agent Manager console. Do not uninstall the UNIX agent from computers using the Agent for NetIQ Security Manager or NetIQ Secure Configuration Manager. For more information about uninstalling the UNIX agent, see the *AppManager for UNIX and Linux Servers Management Guide*, available on the AppManager Modules Documentation page (http://www.netiq.com/documentation/appmanager-modules).

   Uninstalling the agent does not remove the agent's software inventory information from the CCDB.

5. UNIX Agent Manager

   For more information about uninstalling the UNIX Agent Manager, see the *AppManager for UNIX and Linux Servers Management Guide*, available on the AppManager Modules Documentation page (http://www.netiq.com/documentation/appmanager-modules).

6. Control Center and Deployment services

   For more information about uninstalling Control Center and Deployment services, see "Uninstalling Control Center and Deployment Services" on page 123.

7. Control Center console
8. Task Scheduler service
9. QDB

   For more information about uninstalling the QDB, see "Uninstalling the QDB or CCDB" on page 123.

10. CCDB

    For more information about uninstalling the CCDB, see "Uninstalling the QDB or CCDB" on page 123.

Uninstalling Windows Agents and Modules Remotely

Use Control Center to uninstall Windows agents and modules from remote computers.

To remotely uninstall Windows agents and modules:

1. Use the Deployment Rule Wizard to create a deployment rule to uninstall the selected components.

   For more information about using the Deployment Rule Wizard, see the *Control Center User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

2. On the **Packages** pane, select **Uninstall the selected packages from the target computers**.
3. On the **Deployment Schedule** pane, set up a schedule for executing the deployment rule.
4. To allow the Deployment Service to generate a deployment task, enable the deployment rule.

   For more information about enabling deployment rules, see the *Control Center User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

5. To begin the uninstallation, approve the deployment task.

   For more information about approving deployment tasks, see the *Control Center User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

   If an outstanding deployment task to uninstall the agent is waiting for approval, you cannot create additional deployment tasks to uninstall modules.

Uninstalling Control Center and Deployment Services

You can uninstall all services, or select individual services to uninstall.

To uninstall Control Center and Deployment services:

1. Open the Add or Remove Programs application in Control Panel.
2. Select **NetIQ AppManager Control Center** from the list of installed programs.
3. Click **Remove**.
4. (Conditional) If you want to uninstall all services, on the Welcome window select **Remove** and click **Next**.
5. (Conditional) If you want to select services to uninstall, on the Welcome window select **Modify** and click **Next**.
6. (Conditional) If you want to select services to uninstall, deselect the services you want to uninstall and click **Next**.
7. On the Confirmation window, click **Uninstall** to begin uninstalling the services.

Uninstalling the QDB or CCDB

To uninstall the QDB or CCDB, use Microsoft SQL Server Management Studio to delete the repository, the SQL Server jobs that were created when you installed the repository, and the account that owns the repository.

If a Windows user account owns the QDB or CCDB, the repository is the default database for the Windows user in Microsoft SQL Server. Before you uninstall the QDB or CCDB, use Microsoft SQL Server Management Studio to change the default database for the Windows user account to an appropriate system database. Otherwise, the Windows user will not be able to connect to the SQL Server after you uninstall the QDB or CCDB.

To uninstall the QDB or CCDB:

1. Stop any services that connect to the QDB or CCDB.
2. On the computer where you want to uninstall the QDB or CCDB, start SQL Server Management Studio.
3. (Conditional) If the QDB or CCDB owner is a Windows user account, change the default database for the Windows user to an appropriate system database.
4. Select the QDB or CCDB you want to uninstall.
5. Delete the QDB or CCDB from the SQL Server.
6. (Conditional) If you are deleting the QDB, use the Task Scheduler Configuration Utility to delete the following SQL Server jobs:
   - NetIQ Archive Event

- NetIQ Dynamic View
- NetIQ MS HealthCheck
- NetIQ Monitoring Policy
- NetIQ Purge Archive Event
- NetIQ Rule Based Dynamic View
- NetIQ Update MG Server Membership
- NetIQ Uphold Parameter Overrides
- NetIQ Daily
- NetIQ Hourly
- NetIQ Minutely
- NetIQ Weekly
- NetIQ License Audit
- NetIQ Remove Old Data
- NetIQ Rebuild Data Views
- NetIQ Clear Data Details
- NetIQ VSG Modtime Update

7 (Conditional) If you are deleting the CCDB, use the Task Scheduler Configuration Utility to delete the following SQL Server jobs:

- NetIQ Daily
- NetIQ Half-Hourly Task
- NetIQ Hourly
- NetIQ Manage SQL Jobs
- NetIQ SMV Hourly Task

8 Delete the account that owns the QDB or CCDB.

D Installing on Microsoft Cluster Service

This appendix describes procedures for installing AppManager components on Microsoft Cluster Service (MSCS). NetIQ Corporation does not support installing AppManager components on other clustering products.

Installing the QDB and CCDB on MSCS

You can install the QDB and CCDB on a local or remote virtual SQL Server running in either **active/passive** or **active/active** mode. You do not have to run the setup program on the SQL Server. If you do run the setup program on the SQL Server, you can run it on an active or passive cluster node.

In active/passive mode, if the QDB names are unique, you can install multiple QDBs on the same instance of the virtual SQL Server. In active/active mode, each active instance of the virtual SQL Server can have a QDB with the same name.

During installation, when you specify the SQL Server instance that will host the QDB or CCDB, specify the network name of the cluster to which the instance belongs.

The default location for the QDB and CCDB data and log files is the SQL Server's data folder on the shared cluster disk. You must use the default locations when installing the QDB and CCDB on a cluster.

For more information about installing the QDB, see the Chapter 6, "Installing a Management Site," on page 63. For more information about installing the CCDB, see Chapter 7, "Installing Control Center Components," on page 71.

Installing the Management Server on MSCS

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 designating primary and secondary management servers, see the Chapter 8, "Installing Agent Components," on page 85 and the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager).

If you have a requirement to install the management server on MSCS, contact Technical Support.

Installing Agents on MSCS

Because the agent does not run as a virtual application, you must install it on the local disk of each cluster node, and not the shared cluster disk. Also install the same set of modules on each cluster node.

Before installing the AppManager agent and modules, determine whether the virtual server is active on the cluster node where you are performing the installation. If the virtual server is not active on the cluster node, some resource objects will not be discovered. To avoid this issue, move the virtual server to each node before you install the agent. If it is not possible to move the virtual server before you install the agent on the cluster node, you can move it later and use the Control Center console to run the appropriate operating system Discovery Knowledge Script to discover the agent.

Before you install the agent, ensure you have a Windows account for the agent services to use. For more information about the account, see "Reviewing Required Accounts and Permissions" on page 24. For more information about installing the agent locally on each cluster node, see "Installing the Windows Agent Locally" on page 88. For more information about installing modules, see "Installing Modules" on page 90.

You can remotely deploy agents to cluster nodes as long as you deploy to physical nodes and not the virtual server. For more information about remotely deploying agents, see "Installing Agents Remotely" on page 90.

After you install the agent and modules, you can run a combination of application-specific, MSCS, and NT Knowledge Scripts to monitor a virtual server.

Installing the Command Queue Service on MSCS

If you have a requirement to install the command queue service on MSCS, contact Technical Support.