NetIQ® AppManager®
Control Center User Guide

December 2018

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/.

About this Book and the Library

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

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

Intended Audience

This guide provides information for system and network administrators working with AppManager console programs, including the NetIQ AppManager Control Center console and Chart Console.

For information about configuring and maintaining an AppManager site, see the *Administrator Guide for AppManager*.

Other Information in the Library

The library provides the following information resources:

Installation Guide

Provides detailed planning and installation information.

Administrator Guide

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

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.

NetIQ UNIX Agent documentation

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

About NetIQ Corporation

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

Our Viewpoint

Adapting to change and managing complexity and risk are nothing new

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

Enabling critical business services, better and faster

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

Our Philosophy

Selling intelligent solutions, not just software

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

Driving your success is our passion

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

Our Solutions

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

Contacting Sales Support

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

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

Contacting Technical Support

For specific product issues, contact our Technical Support team.

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

Contacting Documentation Support

Our goal is to provide documentation that meets your needs. 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.

Contents

1 Introduction to AppManager

AppManager is an integrated enterprise systems management solution that lets you effectively manage your applications and infrastructure within today's tight resource constraints and growing workloads.

AppManager provides enterprises with the ability to:

- **Gain Greater Control over the IT Environment:** AppManager establishes control through features such as automated detection and deployment, policy exception management, secure delegation, and self-maintaining service maps. These features help establish a solid systems management foundation so that enterprises safely adopt and exploit next-generation technologies.
- **Improve IT Management Productivity and Visibility:** AppManager gives IT automation that adapts to dynamic business environments. End-to-end service visibility vastly reduces and preempts business service downtime and improves event impact assessment through visually represented service maps.
- **Maximize Return on IT Investment:** AppManager's extensive out-of-the-box functionality, flexible integration with existing IT infrastructure, extensible platform, and easy customization ensure that enterprises benefit from maximum functionality with the shortest time to value.

1.1 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	Required
	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 the UNIX agent documentation, which is included in the AppManager UNIX download package.	
Report-enabled agent	Optional supplement to the agent that allows you to create and configure reports on selected computers in your environment	Optional
	You discover report-related elements on agent computers to enable different types of reporting. For more information about enabling the agent reporting capability, see the *Installation Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).	
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 CCDB), 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

1.2 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.
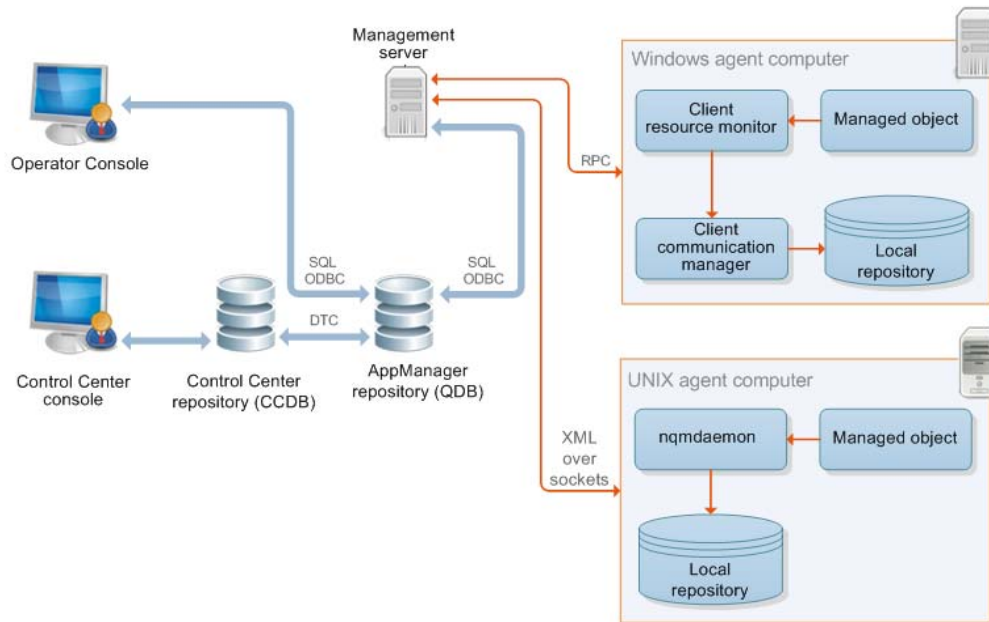
The following graphic illustrates the architecture of the Control Center console.
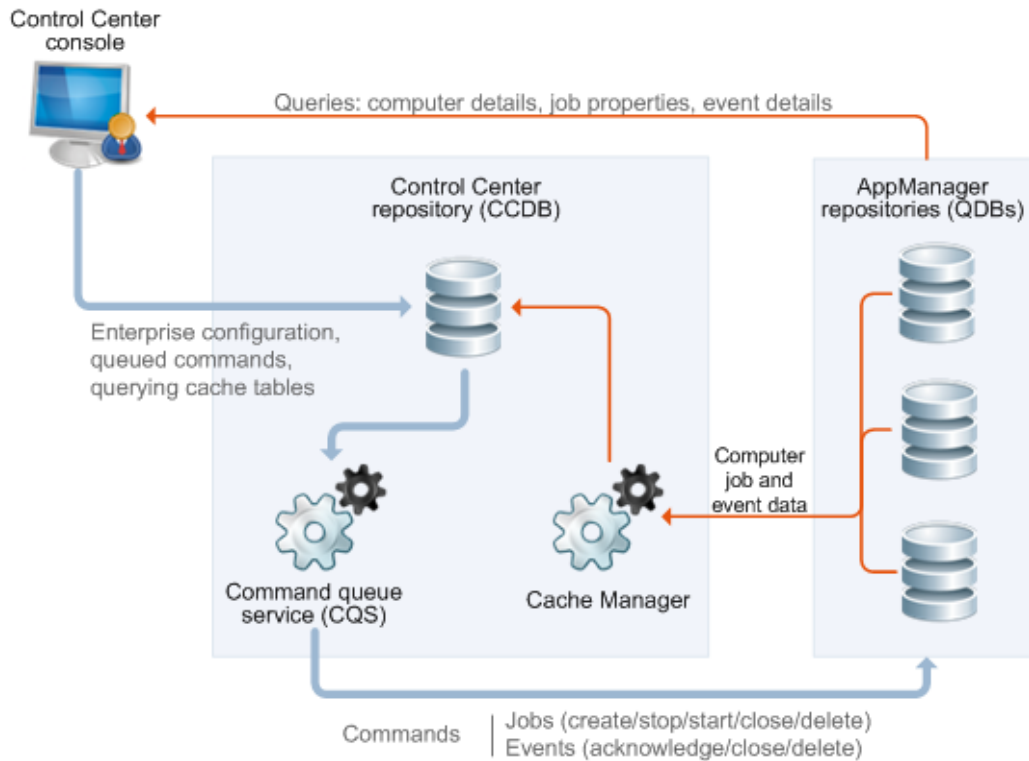
For information about the options for distributing AppManager components across multiple computers, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

1.3 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.

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 occurs 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.

1.3.1 Monitoring in a Windows Environment

When you start a job on a Windows computer, the Control Center console notifies the CCDB that you requested a Knowledge Script to run (the Operator Console contacts the QDB directly). The command queue service updates the appropriate QDB with information about the job properties and the QDB, with updated 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.
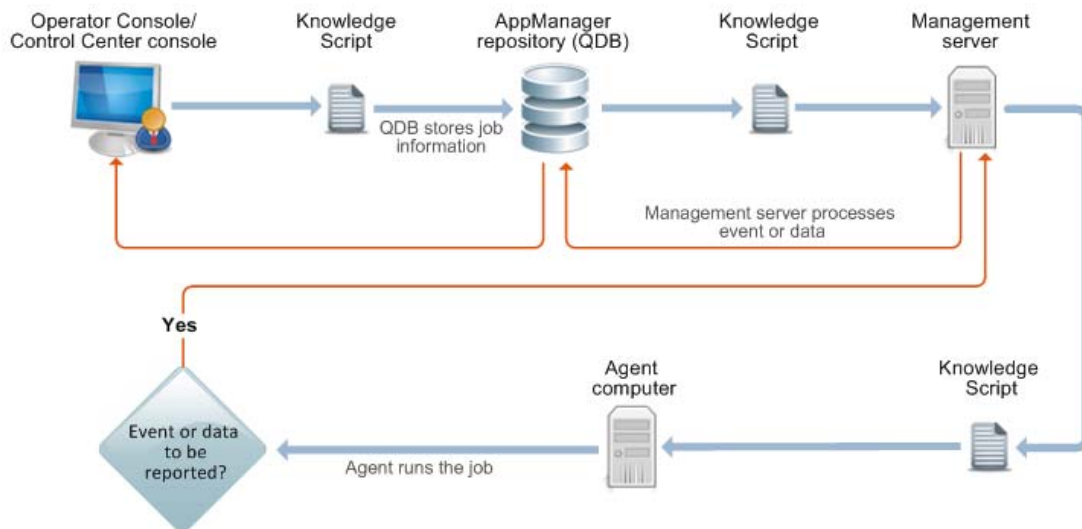
As the agent runs a job, it uses the associated Knowledge Script to gather information. Knowledge Scripts gather information a variety of ways. For example, a Knowledge Script might check the value of performance counters, read log files, execute queries, or access system tables.

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 the *Installation Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

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.

1.3.2 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.
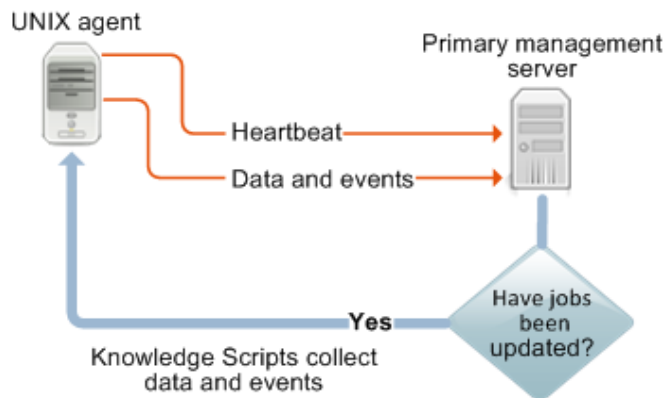
After it receives a job from the management server, the UNIX agent runs the job to access log files, system tables, or other data providers and retrieves the information requested.

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.
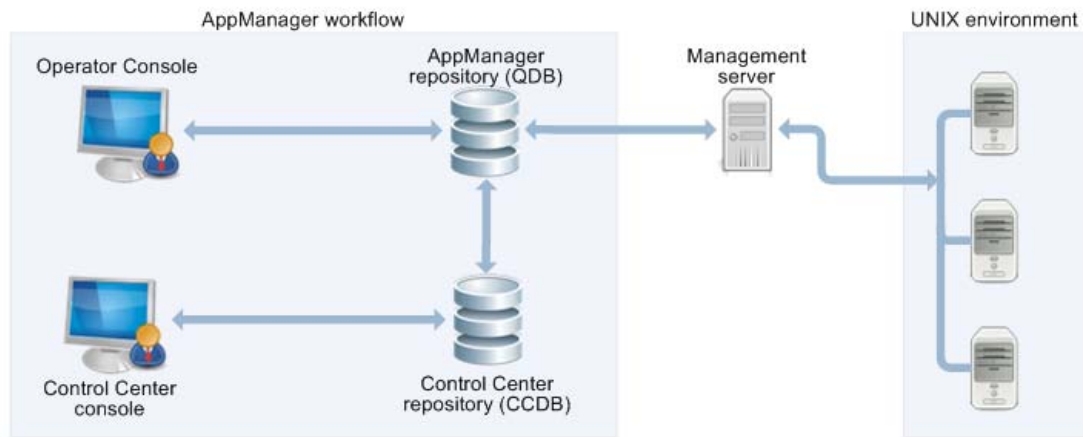
1.3.3 Working with Both Windows and UNIX Computers

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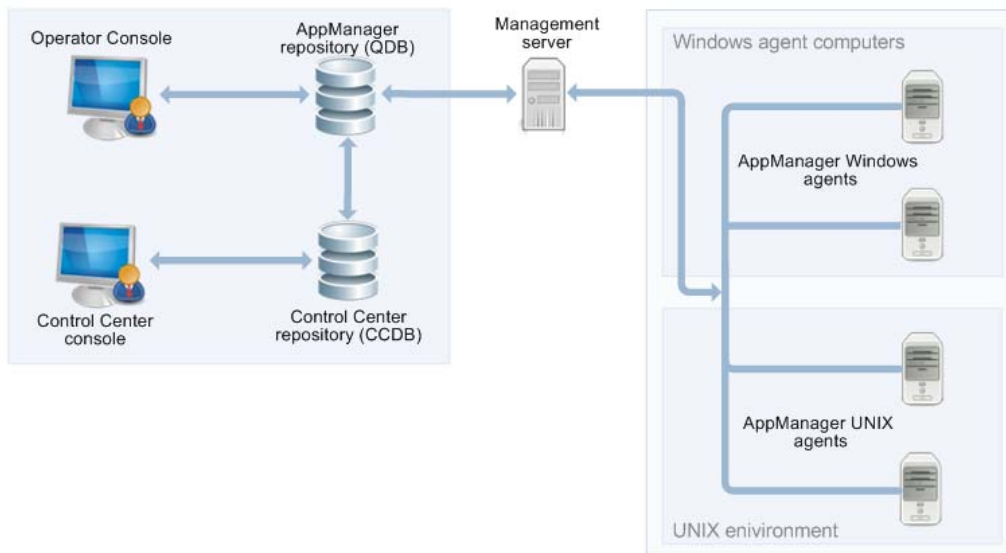
In an environment with both Windows computers and UNIX or Linux computers, a single management server can communicate with:

- 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-9-2).
- Installing and configuring a UNIX agent and monitoring in a UNIX or Linux environment, see the *AppManager for UNIX Management Guide*, available on the AppManager Modules Documentation page (http://www.netiq.com/documentation/appmanager-modules).

1.4   How are Control Center Components Used?

The management server is a Windows computer running the `NetIQms` service. The `NetIQms` service manages the event-driven communication between the QDB and the agent computers.

The QDB is a central Microsoft SQL Server database that stores management data for a site.

The management server and QDB can reside on the same physical computer or on separate computers to distribute the workload. NetIQ Corporation recommends installing them on separate computers.

The command queue service polls the Command Queue at regular intervals, looking for commands to be sent to the individual QDBs.

The command queue service retrieves blocks of commands from the Command Queue. The command queue service sends the commands to the QDBs that serve the managed computers.

The CCDB is a central Microsoft SQL Server database that stores view data (**Events**, **Jobs**, **Servers**, **Knowledge Scripts**, and **Service Map** views) brought from one or more QDBs for individual management groups that appear in the Control Center console. The CCDB contains:

- A **Cache** table to hold view data.
- A **Command Queue** table that contains commands to be executed on any of the individual QDBs (for example, to acknowledge an event or create a job).
- Various configuration tables that hold other information, such as definitions of management groups.

The Cache Manager is a set of SQL tasks that poll for information from each QDB based on the management group definition and includes lists of servers, running jobs, and events.

When you use the Control Center console to start a job, a record is inserted into the CCDB and the command queue service communicates the Knowledge Script and its properties to the appropriate QDBs. AppManager then creates the job in the QDB and the management server sends the job to the agent computer.

The Cache Manager retrieves the job status information from the individual QDBs and puts the information into the CCDB.

Communication between the Control Center console and the CCDB is through ODBC.
Communication between the CCDB and the individual QDBs is through DTC.

1.5 How Does Deployment Work?

The Control Center console and UNIX Agent Manager allow you to quickly and easily install deployment packages for the agent, module updates, and hotfixes to the remote computers in your environment. This section describes deploying Windows components. For more information about using UNIX Agent Manager to deploy UNIX and Linux components, see the AppManager UNIX documentation.

Deployment provides:

- Automated detection of new computers and applications.
- Automated remote installation of agents and modules.
- Rapid response to changing environments.
- Multiple, firewall-friendly deployment servers.
- Customized deployment schedules to meet corporate maintenance windows.

1.5.1 Rule-based Deployment

The Control Center console provides an easy-to-use wizard to help you create deployment rules. A **deployment rule** allows you to select computers and install the agent and appropriate modules based upon a variety of criteria, including:

- Organization units in Active Directory.
- Remote registry check.
- Management groups in the Control Center console.
- IP address ranges.
- Operating system or applications installed.

1.5.2 Approval-based Deployment Tasks

When a target computer meets the conditions for a deployment rule, Control Center generates a deployment task. A **deployment task** delivers a specific deployment package to a particular computer at a scheduled time.

In the Control Center console, you approve a deployment task *before* the Deployment Service can install the agent or module on the target computer.

Each deployment task includes the required credentials to run on the target computer.

To avoid monitoring interruptions during business hours, you can schedule deployment tasks to run outside business hours.

1.5.3 Windows Agent Deployment Architecture and Components

The Deployment Service communicates with the CCDB to process deployment rules and deployment tasks. Deployment tasks run on the target computer using credentials provided in the deployment rule.

The computer where the Deployment Service is installed is the **deployment server**. If you configure more than one Deployment Service, when you configure a deployment rule, you can choose the Deployment Service you want.

To deploy the Windows agent to computers behind a firewall, configure a Deployment Service on a computer that is also behind the firewall.

The Deployment Web Service consists of two Web services that are installed on a Microsoft Internet Information Services (IIS) server called the **Web Depot**. The Deployment Web Service:

- Checks deployment packages into the Web Depot.
- Distributes deployment packages to the Deployment Services. The Deployment Web Service uses Microsoft Background Intelligent Transfer Service (BITS) server extensions to distribute deployment packages to the Deployment Service.
- Provides a communication proxy for Deployment Services that are across a firewall.

After you install the Control Center deployment components, use the Control Center console to configure a deployment rule.

For more information about how to deploy the UNIX agent to remote UNIX and Linux computers, see the AppManager UNIX documentation.

1.6 What are AppManager Reports?

Whether you need to report on service-level conformance, historical trends, or ad hoc performance results, AppManager delivers a powerful and flexible reporting infrastructure, including:

- Performance-related reports to help you understand the performance of your applications and systems.
- Trend analysis reports to help you plan for system or application upgrades.
- Service level agreement reports to quantify the availability of services and service response time.
- Event summary reports to provide an overview of the events AppManager creates. Use this information to develop a plan of action.
- Configuration reports to provide an overview of the system details being monitored and configuration details, such as event threshold settings.

AppManager reports use the same Knowledge Script-based mechanism that you use to run jobs that collect data and raise events.

An AppManager report queries data from a single QDB. At this time you cannot use AppManager to report on data from more than one QDB in the same report. To report on AppManager data from more than one QDB in the same report, use the NetIQ Analysis Center product (Analysis Center).

Analysis Center enhances the reporting capabilities of the Service and Security Management solutions from NetIQ Corporation, helping IT organizations communicate service levels and security compliance. By offering quick time-to-value without sacrificing functionality or flexibility, Analysis

Center rises above the level of individual metrics. The product provides vital information, such as overall system utilization, security incident interpretation, root-cause analysis and historical trending, and critical trends and conditions.

For more information about AppManager reports, see Chapter 10, "Running Reports," on page 151. For more information about Analysis Center, see the NetIQ Analysis Center Product Upgrades Web site.

Please discuss your AppManager reporting needs with your NetIQ Sales Representative or NetIQ Partner.

2 Getting Started

This chapter provides an introduction to the Control Center console and suggests how to get started using AppManager in your organization.

After you install AppManager components on one or more computers, use the Control Center console to monitor your environment. For information about installing AppManager, and system requirements, see the *Installation Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

2.1 Starting the Control Center Console

As a Control Center administrator, you can start the Control Center console. To enable other users to access Control Center, configure Control Center security. For information about configuring user access to Control Center, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

To start the Control Center console:

1 On a computer where you installed the Control Center console, click the **Start** button, then **Programs** > **NetIQ** > **AppManager** > **AppManager Control Center**.

2 Log on to the CCDB. (When you start other AppManager programs, such as the Chart Console, you will log on to a particular QDB.) If you are not sure what to enter in any field or have problems logging on, see your AppManager administrator.

For...	Do this...
Server Name	Type the name of the SQL server that manages the CCDB. When specifying a computer name, you can enter the Windows computer name or the IP address. At this time, AppManager only supports v4 IP addresses. For example, to specify a named SQL Server instance, you can enter `10.1.10.43\INST1`.
Repository	Type the name of the CCDB. If the repository uses the default name, `NQCCDB`, you do not have to complete this field.
Use Windows Authentication	Select this option to specify that SQL Server uses the Windows user information to validate the user before granting access to the CCDB. The Control Center console computer must be part of a Windows domain or workgroup. If you do not select the **Use Windows Authentication** option, SQL Server uses the standard SQL Server security validation. This is the default when SQL Server is configured to use both SQL Server and Windows authentication. You need to add the login name to SQL Server before a user can log in. If you enable FIPS-compliant security for the Control Center console, you cannot log in using SQL credentials. For more information about FIPS-compliant security, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).
User name	To use the standard SQL Server security validation to grant access to the CCDB, type the user name of the SQL Server login account used to access the CCDB.

For...	Do this...
Password	To use the standard SQL Server security validation to grant access to the CCDB, type the password for the SQL Server login account.
Sign me in automatically	Select this option to save the Windows or SQL Server credentials used to log on to the console. The Control Center console no longer prompts for credentials if you log on using the same credentials.

3 Click **OK.**

Now that you are logged in to the CCDB, the Control Center console displays information dynamically.

NOTE

 By running separate instances of the Control Center console, different users can access the same CCDB from the same console computer.

 If you are running the Control Center console on Windows Vista or Windows Server 2008, you cannot start the console with different logon credentials using the **Run As** command.

Your AppManager environment should have the same version of the core components (QDB and management server) and Control Center components (CCDB, command queue service, and console). If you are upgrading Control Center from an earlier version, upgrade your core components to the current version as soon as possible. You cannot perform some AppManager tasks while your core components are still on an earlier version. For more information, see the *Upgrade and Migration Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

2.2 About the Control Center Console

The Control Center console is where you do most of the work to manage the systems on your network. To give you the tools for viewing and managing your computer resources, the Control Center console consists of the **Navigation** pane and the current view pane, which changes based on your selection in the **Navigation** pane. If you select an object in the **Enterprise Layout** view of the **Navigation** pane, the view pane displays the contents for that object. Tabs on the right side of the view pane display information about system status, tasks you can perform, Knowledge Scripts, and Help.

2.2.1 Working with Panes

Information about the systems you are managing is grouped in panes.

This pane...	Does this...
Navigation	◆ **Enterprise Layout** Shows a hierarchical list of management groups, views, and folders. Click to expand the list and select a management group, view, or folder. By default, the **Enterprise Layout** view shows the cumulative worst open event status icon for any physical computer or logical server in the view. This view also shows default management groups for monitoring the health of AppManager components and services. ◆ **Deployment** Shows deployment packages, rules, tasks, and the software inventory. Use the Deployment Rule Wizard to install the AppManager Windows agent and AppManager modules on remote computers. For more information, see Chapter 9, "Deploying AppManager to Agent Computers," on page 131.
View pane	Displays information based on your selection in the **Navigation** pane. If you select: ◆ The top-most AppManager icon or a management group, the view pane displays either AppManager configuration information or a summary about the management group. ◆ A view, the view pane displays the objects in the view. Select an object to displays its details. For example, in a **Servers** view, select a server to show the status of events, jobs, custom properties, discovery details, and chart data on that server. For information about Control Center views, see Section 2.2.2, "Working with Management Groups," on page 30. Displays information based on your selection in the **Deployment** pane. If you select: ◆ **Packages**, the view pane displays all packages checked into the QDB. ◆ **Rules**, the view pane displays all defined deployment rules and their status. ◆ **Tasks**, the view pane displays all deployment tasks the Deployment Service generated. ◆ **Software Inventory**, the view pane displays all agents and modules installed in the managed environment.
Tasks	Displays related tasks for the selected object in the view pane. These tasks are similar to right-click menu items.
System Status	Displays the status of Control Center, including: ◆ **Component Status** Displays the connection or availability status of key AppManager components, such as the command queue service and QDBs. ◆ **Command Queue** Displays the status of commands in the Queue Manager. ◆ **Deployment** Displays the status of deployment tasks.
Knowledge Scripts	Displays all Knowledge Scripts checked into the primary QDB, regardless of whether you have discovered objects pertaining to a particular Knowledge Script. This is different than a **Knowledge Scripts** view, which only displays Knowledge Scripts relating to discovered objects in the managed environment.
Help	Displays a context-sensitive Help topic for the active pane or selected item.

For more information about customizing the Control Center console, see Section 3.1, "Customizing the Control Center Console Layout," on page 37.

2.2.2 Working with Management Groups

The Control Center console uses management groups to manage the physical computers and logical servers in your AppManager environment. You can configure a management group to include agent computers or logical servers from one or more QDBs based on:

- ◆ QDB views.
- ◆ Groups of servers in a QDB view.
- ◆ Rules that can select physical computers or logical servers based on their properties, such as the name of the computer or discovered resources.
- ◆ An ad hoc list of computers.

To manage the objects in a management group, the Control Center console includes views. The standard views in most management groups are:

Servers

Displays a list of physical computers and logical servers in the management group. Select a server from the list to display server information and perform common tasks like responding to events, managing jobs and custom properties, and viewing discovery details.

Jobs

Displays a list of jobs on the physical computers and logical servers in the management group.

Events

Displays a list of events in the management group. Use the view to group events, for example, by computer, and respond to events.

Knowledge Scripts

Displays a list of Knowledge Scripts. Use the list to browse Knowledge Scripts and create groups of Knowledge Scripts. A **Knowledge Scripts** view shows only those Knowledge Scripts pertaining to those objects discovered in your managed environment. This is different than the **Knowledge Scripts** pane, which shows all Knowledge Scripts checked into the primary QDB.

Summary

Displays a summary of the members of the management group, events, jobs, and servers information.

The following standard management groups do not include a **Jobs**, **Events**, or **Knowledge Scripts** view:

- **All Logical Servers**
- **Databases**
- **Network Devices**
- **Virtual Infrastructure**

You can also create **Service Map** views to manage events and view the overall health of a business process. For more information, see Chapter 8, "Using Service Map Views," on page 117.

NOTE: In the Control Center console, administrators assign groups of users permission sets, and assign both user groups and permission sets to management groups to control access to tasks and managed objects in the Control Center console. If you don't have access to a particular management group, you might not be a member of a user group assigned to the management group. For more information about Control Center security, see the *Administrator Guide for AppManager* (available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2)) or contact your AppManager system administrator.

2.2.3 Deploying AppManager Agents

The Control Center console includes a rule-based deployment tool, the Deployment Rule Wizard, to install and update the AppManager Windows agent. The Control Center console provides an easy-to-use interface for configuring deployment rules, managing installation packages, and validating deployments before you send them. For more information, see Chapter 9, "Deploying AppManager to Agent Computers," on page 131.

The UNIX Agent Manager provides similar capabilities for the UNIX agent. For more information about the UNIX Agent Manager, see the *AppManager for UNIX Management Guide*, available on the AppManager Modules Documentation page (http://www.netiq.com/documentation/appmanager-modules).

2.3 Managing QDBs

To manage a QDB with Control Center, use the Control Center console to add the QDB. Note that you cannot add a QDB to more than one CCDB.

After you add a QDB, its agent computers are automatically displayed in Control Center management groups that are configured to include **All Repositories**, such as the **Master** management group.

The **primary** QDB is the QDB that contains the Knowledge Scripts that you want to use when running AppManager jobs from Control Center. If you have more than one QDB, you can change the primary QDB. For more information, see Section 2.3.2, "Performing QDB Management Tasks," on page 33.

2.3.1 Before You Begin

Before you add a QDB:

- If the QDB is in a different domain than the Control Center console user, ensure that **both** domains are trusted.

- Properly configure Kerberos delegation and SQL Server DTC settings. If you do not properly configure the QDB computer, the Control Center console adds the QDB but does not allow you to manage the computers. For more information, see the *Installation Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

- To manage more than one QDB with Control Center, configure Control Center to use the same authentication method to communicate with all QDBs. The authentication method you choose depends on how the SQL Server Agent service is configured. For example, if the service on the CCDB computer runs as LocalSystem, and you add the QDB using Windows authentication, the Cache Manager cannot communicate with the QDB.

- When you add a QDB earlier than the current version, the command queue service runs a patch script to enable the QDB to synchronize with the CCDB. The command queue service displays the status of the patch in the Queue Manager. Before you add an earlier version of a QDB, be sure to apply the latest QDB hotfixes.

- Control Center does not upgrade the QDB to the latest AppManager version. For information about upgrading the QDB, see the *Upgrade and Migration Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

- Configure SQL Server and AppManager security to enable Control Center to communicate with the QDB you want to add.

To configure SQL Server and AppManager security:

1 In Microsoft SQL Server Management Studio, add the Log On As account for the NetIQ AppManager Control Center Command Queue Service to the list of database users on the QDB and make sure the account has the `db_owner` database role.

2 Add the Windows user account for each Control Center user to the list of database users on the QDB. Give each user the same permissions they have on the primary QDB.

To enable the user to...	Do this...
- Create, copy, or delete Knowledge Scripts or Knowledge Script Groups	- Give the user `db_owner` permission on the QDB.
- Perform all other tasks	- Give the user `public` permission on the QDB.

3 In AppManager Security Manager, give the QDB user the same AppManager role they have on the primary QDB.

To enable the user to...	Do this...
- Create, copy, or delete Knowledge Scripts or Knowledge Script Groups	- No AppManager role is required when the user has `db_owner` permission.
- Perform all other tasks	- Add the QDB user to the Read-only, Standard, or Administrator AppManager role.

You are now ready to add the QDB to Control Center. For more information, see .

2.3.2 Performing QDB Management Tasks

You can use the Control Center console to perform the following QDB management tasks:

- Add QDBs.
- Configure the cache settings.

  Cache settings determine how the Control Center Cache Manager synchronizes event and job information in each QDB in the Control Center console. You can improve the overall performance of the Control Center console by reducing the amount of information the Cache Manager must synchronize.

  By default, the Cache Manager synchronizes:

  - Open and acknowledged events.
  - All jobs except closed jobs.

- Change the authentication method.

  You must be a member of the Control Center Administrator group to change the authentication method that Control Center uses to access the QDB.

  You can change the authentication method for Control Center, for example, if your corporate security policy requires you to use Windows authentication for all applications or if you choose to implement FIPS-compliant security.

  Before changing the authentication method:

  - If the SQL Server Agent service on the CCDB computer is running as a domain user account, make sure this account has privileges on each QDB.
  - Make sure all QDBs that Control Center manages use the same authentication method.
  - Correctly configure the Log On As account for the SQL Server Agent service:

    - To use Windows authentication, configure the Log On As account for the SQL Server Agent service on the CCDB computer as a Windows user account with permissions on the QDB. You can configure the SQL Server Agent service with the same Log On As account as the command queue service.
    - To use SQL authentication, the Log On As account for the SQL Server Agent service on the CCDB computer can be LocalSystem or a SQL user account with permissions on the QDB.

  - If you are migrating a QDB, Control Center must use Windows authentication to connect to the QDB for the migration.

    For more information about migrating a QDB, see the *Upgrade and Migration Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

  - Consider whether you are using or plan to use only FIPS-compliant algorithms for AppManager security. If you enforce FIPS-only compliance, the console continues to accept SQL authentication, with certain limitations:

    - The console cannot add or modify a QDB.
    - The console cannot create SQL accounts.

    For more informatin about AppManager FIPS compliance, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

◆ Change the primary QDB.

You must be a member of the Control Center Administrator group to change the primary QDB.

The **primary** QDB is the QDB that contains the Knowledge Scripts that you use when you run AppManager jobs.

When you change the primary QDB, it can take some time for Control Center to update the list of Knowledge Scripts in the CCDB to match the Knowledge Scripts in the primary QDB.

You cannot start a new job, view the properties of a running job, or update a job when Control Center synchronizes the Knowledge Scripts in the primary QDB.

◆ Remove QDBs from Control Center.

You must be a member of the Control Center Administrator group to remove a QDB.

Before you remove the QDB from Control Center, remove it from any management groups that still include it as a member. For more information, see Section 4.3, "Working with Management Groups," on page 56.

If the QDB you want to remove is the primary QDB, designate a new primary QDB before you remove the current primary QDB.

If you move a QDB from one SQL Server to another, you do not need to remove the QDB from the CCDB. Use the Control Center console to update the new SQL Server details for the QDB. For more information, see the *Upgrade and Migration Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2)

After you remove a QDB from Control Center, the CCDB no longer contains the event and job information and the information is no longer available in the Control Center console. The existing jobs, events, and data remain in the QDB.

To perform QDB management tasks:

1 Log on to the Control Center console with a login account that is a member of the Administrator group. This user must also have the `db_owner` database role for the QDB you are managing.

2 On the **Global Tasks** tab of the ribbon, in the **Administration** group, click **Manage Repositories**.

3 Complete the appropriate action:

To...	Do this...
Add a QDB	1. Click **Add New**. 2. Provide information that identifies the QDB and select the authentication method the command queue service wil use to access the QDB. 3. Indicate whether this will be the primary QDB.
Configure the cache settings	1. Click **Modify**, and then click the **Cache** tab. 2. Select options for caching event and job information. 3. Specify a threshold date for viewing event and job information.

To...	Do this...
Change the authentication method	1. Select a QDB, and then click **Modify**.
	2. Configure the connection details between Control Center and the QDB.
	3. Stop the command queue service.
	4. In SQL Server Management Studio, update the linked server properties for the QDB:
	◆ (Conditional) If you changed from Windows to SQL authentication, for the **Security** options, select **Be made using this security context** and provide the correct credentials.
	◆ (Conditional) If you changed from SQL to Windows authentication, for the **Security** options, select **Be made using the login's current security context**.
	5. Start the command queue service.
Change the primary QDB	1. Select the QDB you want to be the primary, and then click **Modify**.
	2. Select **Primary Repository**.
Remove a QDB	Select the QDB you want to remove, and then click **Remove**.

3 Configuring the Control Center Console

This chapter describes how you can configure the Control Center console to fit your needs.

3.1 Customizing the Control Center Console Layout

The Control Center console provides a number of ways for you to customize the appearance and performance of the console as well as the data it displays. You can specify what panes to display and how to arrange them, you can group, sort, and rearrange the data the console displays in the view pane, you can filter data, and you can use security settings to limit the data available to different console users.

To improve the performance of the Control Center console, change the console options to optimize the number of rows you want to view at one time. As a rule, you can improve performance by reducing the number of rows the console displays. For more information, see Section 3.2, "Setting Preferences," on page 41.

3.1.1 Selecting a Management Group

The Control Center console organizes the computers and other resources you are managing into management groups, each of which contain multiple views that determine the details you see in the view pane. For example, the **Servers** view displays all of the managed servers in the management group. By default, the Control Center console displays the **AM Logical Servers**, **Master**, **Agent Managed Computers**, and **AM Platform Components** management groups. The Control Center console displays management groups in the **Enterprise Layout** view of the **Navigation** pane. For more information, see Section 2.2, "About the Control Center Console," on page 28.

In a management group, switch views by clicking the view you want to use. When you switch from one view to another, the view pane reflects information for the selected view. The **AM Logical Servers** management group (and each management group contained within it) includes only the **Servers** view. For more information about the standard views included with management groups, see Section 2.2.2, "Working with Management Groups," on page 30.

Control Center permissions allow you to display only the management groups in which you are interested. For example, if you are the Exchange administrator, you might want to focus strictly on Exchange servers using the Exchange management group.

You can select only one view within a management group at a time.

3.1.2 Saving View Settings as the Default

You can save the settings in the current view, such as sorting, grouping, and local filters, as the default for that view. You must be a Control Center administrator or have permissions to modify a view to save view settings as the default. Changes you make to the **Server Information** pane view are not saved as part of the default settings for a view. This affects **Servers** views and **Service Map** views only.

To save the current view as the default view:

1 On the **Main** tab of the ribbon, click **Action**.

2 Click **Save View Settings As Default**.

3.1.3 Working with the View Pane

In the standard Control Center views (**Knowledge Scripts**, **Jobs**, **Events**, and **Servers**) the view pane consists of a grid that has several powerful features for manipulating its data. These features enable you to perform your job functions more efficiently.

Working with Columns

In the view pane, you can perform the following actions on the grid columns to manipulate the data that Control Center displays:

◆ Resize the column width to the best fit.

To resize a single column width to best fit the view, right-click the column heading and click **Best Fit**. To resize all columns in the view, right-click a column heading and click **Best Fit (all columns)**. Both commands calculate the optimum column width and automatically adjust the width to those values.

◆ Rearrange columns.

You can drag a column by its heading to the right or left to change the order of the columns.

You can also configure the default columns that appear in a view, as well as the column order, by right-clicking a column head and clicking **Customize Columns**.

◆ Lock columns.

You can lock columns on either the left-hand or right-hand side of the grid. When a column is locked, you cannot drag it to a new position and it does not scroll horizontally. To lock a column, right-click the column heading and click either **Fixed Right** or **Fixed Left**.

If you have so many columns in your grid that you must scroll horizontally to see them, you can lock columns so that they do not scroll. For example, you might lock the computer name column so it is always visible no matter how much you scroll.

Working with Lists

In the view pane, you can perform the following actions on lists to manipulate the data that Control Center displays:

◆ Collapse the list.

To collapse the contents of the view pane, on the **Main** tab click **Action** and then click **Collapse All Events** or **Collapse All Jobs**.

◆ Sort the list.

Clicking a column heading will grid-sort the column (and therefore the entire grid list) in ascending (descending) order. Clicking again will grid-sort the column by descending (ascending) order. The sort arrow in the column heading shows the order of the grid-sort.

If you grid-sort several columns in order, the last column sorted will control the overall grid-sort order, but the previous sort orders are preserved.

◆ Group the list.

A grouping region at the top of the view pane grid allows you to group the dataset by the values in the column by dragging a column heading to the region. For example, if you group by Knowledge Script category in a **Knowledge Scripts** view, all Knowledge Scripts listed in the view are grouped according to category. In an **Events** view, if you drag the **Severity** column heading into the grouping region, all events in the **Events** view are grouped by severity. Click the column heading in the grouping region to sort the grouped items in ascending or descending order.

You can group items in the view pane by multiple columns. Simply drag the columns you want to group by into the grouping region. Drag and drop column headings either above or below one another to establish the grouping order. For example, to group events by severity and then by status, drag the **Severity** column heading into the grouping region, and then drag the **Status** column heading into the grouping region below the **Severity** column heading.

Grouping data allows you to execute commands on a number of items simultaneously. For example:

- ◆ In a **Servers** view, you can create a new job simultaneously on all the computers in a group. When you do this, the **EnterprisejobID** will be the same for all the jobs. This means you can later group these jobs in a **Jobs** view by grouping on the **EnterprisejobID**.

- ◆ In a **Jobs** view, you can execute the **Start**, **Stop**, **Close**, and **Delete** commands simultaneously on all the jobs in a group.

- ◆ In an **Events** view, you can execute the **Acknowledge**, **Close**, and **Delete** commands simultaneously on all the events in a group.

◆ Filter the list.

The view pane offers several different ways to filter the data that displays in the list:

- ◆ Server filters

  Server filters apply to data that is returned from the CCDB before it is displayed and available in the Control Center console. Applying server filters can significantly improve the performance of the console since the filtering is done by the CCDB and the console only contains data matching the filter criteria. Only Control Center administrators and users that have permissions to modify a view can create, change, or delete server filters.

  Server filters are available in **Events**, **Jobs**, **Servers**, and **Knowledge Scripts** views.

  To apply a server filter, click **Server Filters** to display the **Server Filters** tab of the View Properties window. For more information, see the Help.

- ◆ Local filters

  Local filters apply to any data that is already displayed and available in the Control Center console.

  You can apply either pre-defined or custom local filters. To apply a pre-defined local filter, click the filter icon on the right-hand side of a column heading to display a drop-down list of all the data values that appear in the column.

  To apply a custom local filter, click **Local Filters** to display the **Local Filters** tab of the View Properties window. Use this tab to create custom queries. For more information, see the Help.

- ◆ Column filters

Column filters apply to data contained within a single column of the list.

To apply a column filter, place the cursor over the column heading and click the filter icon that appears in the upper right corner of the heading.

Viewing Events and Jobs

You can change how the console displays events and jobs in the view pane. In either an **Events** or **Jobs** view, on the **Main** tab of the ribbon click **Action**, and then click the view type you want to use.

The following view types are available:

Hierarchical

Displays child events or jobs grouped under parent events or jobs in an expandable/collapsible list. This is the default view type.

Flat view (all events/jobs)

Displays both parent and child events or jobs in a flat list. By default this view includes the **Event ID** and **Parent Event ID** columns so you can identify both parent and child events in the list.

Flat view (child events/jobs)

Excludes any parent events or jobs from the view.

In an **Events** view, you can turn the display of event messages on or off. By default event messages are displayed. Turning event messages off enables you to display more events in the view pane. To turn event messages off, in an **Events** view, on the **Main** tab of the ribbon, click **Action**, and then click **Show Event Message**.

3.1.4 Setting General View Properties

You can set general properties for **Events**, **Jobs**, **Knowledge Scripts**, and **Servers** views. These properties are in addition to view settings such as local and server filters, column settings, grouping, and sorting. Every view type allows you to change the name and the description of the view. Other general properties vary depending on the view. To change general view properties, right-click the view you want to change in the **Enterprise Layout** view of the **Navigation** pane, and then click **View Properties > General**. For more information about general view properties, see the Help.

3.1.5 Saving Your View Settings as the Default

If you are a Control Center administrator or have permissions to modify a view, you can save your view settings as default settings for yourself and other Control Center console users. To save the current view settings as the default, select the view in the **Enterprise Layout** view of the **Navigation** pane, and then on the **Main** tab click **Action > Save View Settings as Default**.

To revert to the default view settings, select the view in the **Enterprise Layout** view of the **Navigation** pane, and then on the **Main** tab click **Action > Revert to Default View Settings**.

3.1.6 Exporting or Printing the Contents of a View

You can export, print, or email the contents of a view, click **Action > Print/Export/Email**and then choose the appropriate option.

3.2 Setting Preferences

You must be a member of the Control Center Administrator group to set Control Center preferences. Preferences control certain aspects of how the Control Center console operates.

To set QDB preferences, use the AppManager Operator Console. For more information, see the *Operator Console User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

3.2.1 Console Options

To configure the Control Center console options, on the **Main** tab, in the **Tools** group, click **Options**. In the Options window, click the **Console** link to configure any of the following options:

- ◆ General

  These options specify the refresh rate for the current view, how to display Severe events in the console, and the behavior of the **Navigation** pane when you start the console.

- ◆ Events

  These options specify how you work with events in the console.

- ◆ Jobs

  These options allow you to configure the console to prompt the user before starting, stopping, or closing a job.

- ◆ Charts

  These options allow you to configure chart settings that affect how much data to display in charts and how often to update the data.

- ◆ Servers

  These options allow you to configure the types of data and how much data Control Center loads in the **Server Information** pane.

- ◆ Knowledge Base

  These options allow you to enable the console to connect to a Knowledge Base server.

- ◆ Skin Selection

  These options allow you to change the look and feel of the interface.

For more information about the Console options, see the Help.

3.2.2 Health Check Options

To configure the Control Center console health check options, on the **Main** tab, in the **Tools** group, click **Options**. In the Options window, click the **Health Check** link. The options allow you to configure settings for agent and management server health check events. For more information about the options, see the Help.

3.2.3   Deployment Options

To configure the Deployment Server options, on the **Main** tab, in the **Tools** group, click **Options**. In the Options window, click the **Deployment** link to configure the following options:

- General
- Credentials

  These are the default credentials to use when deploying an installation package on a target computer (for example, installing the AppManager agent). You must provide a domain or local user account that is a member of the Administrator group on the target computer. When configuring a rule, you can use the default credentials you specify here or provide different logon credentials.

  If you change the default deployment credentials, the change only applies when you create a new rule. Existing deployment tasks maintain their original deployment credentials.

- Schedule

  These options specify how frequently the Deployment Server should re-run rules and search for new computers, and when deployment tasks should run. You can also set a default schedule for running deployment tasks.

- Confirmations

  These options specify whether to display a confirmation window before deleting a package, rule, or deployment task.

- Task Management

  These options specify how to manage deployment tasks, including:

  - How long to keep a deployment task in the Control Center database.
  - How long to display a deployment task in the Control Center console.

For more information about the options, see the Help.

3.2.4   Knowledge Scripts Options

To configure the Control Center console Knowledge Scripts options, on the **Main** tab of the ribbon, in the **Tools** group, click **Options**. In the Options window, click the **Knowledge Scripts** link.

The option **Automatically update Derived KS properties for checked in Base KSs** determines whether AppManager automatically updates the logic of derived Knowledge Scripts when you install a new module.

Knowledge Scripts are classified as **base** Knowledge Scripts or **derived** Knowledge Scripts. A base Knowledge Script is checked in to the primary QDB and holds all of the Knowledge Script settings (the schedule, parameters, and Advanced tab) and the script logic. A derived Knowledge Script is a copy of a base Knowledge Script (a direct copy or a member of a Knowledge Script Group) and shares the logic with the base Knowledge Script but has its own settings. For example, a base Knowledge Script might have a default threshold of 10 for a certain parameter while the derived Knowledge Script has a default threshold of 20. The two Knowledge Scripts share the script logic, but the settings are different.

The default is to automatically update derived Knowledge Scripts. The update retains the settings (such as thresholds and schedule) of the derived Knowledge Scripts and includes monitoring policy jobs but not ad hoc jobs. If a base Knowledge Script includes new values or removed values, AppManager merges those changes to the derived Knowledge Scripts.

If you do not enable the automatic update option, when you log in to the AppManager consoles and base Knowledge Scripts have been updated, AppManager informs you that base Knowledge Scripts are pending propagation to derived Knowledge Scripts and/or running jobs and provides the option to start the Knowledge Script Propagation Wizard to select which Knowledge Scripts and/or jobs to update.

For more information about how Knowledge Scripts and the automatic update option work, see Chapter 5, "Running Monitoring Jobs," on page 67. For more information about monitoring policies, see Chapter 7, "Monitoring by Policy," on page 105.

3.2.5 Security Options

To configure the Control Center console security options, on the **Main** tab, in the **Tools** group, click **Options**. In the Options window, click the **Security** link.

Use the **Security** options to:

- Enable FIPS-compliant security algorithms for AppManager. For more information about FIPS-compliant security, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

- Disable vulnerable Transport Layer Security (TLS) versions. This option allows you to disable any TLS versions (for example, TLSv1.0) that are considered vulnerable.

  This option is selected by default. If you have a management server that communicates with UNIX agents using the **encrypted communications only** or **authentication and encrypted communications** security level and this option is selected, the management server will not be able to communicate with the agents because they require TLSv1.0. You must deselect the option to restore communications.

Any time you change the **Security** options, you must restart any management servers so they will recognize the new security settings. To allow sufficient time for Control Center to synchronize the changes to the connected QDBs, wait at least one minute before restarting management servers.

3.2.6 Event Severity Status Options

To configure the Control Center console event severity status options, in the **Tools** group, click **Options**. In the Options window, click **Event Severity Status**.

Enable the option if you want to display the highest event severity for management groups at the top-most level. Otherwise, you must expand management groups to view the event severity icons for servers in the groups.

If you disable the option, you must refresh the console in order for Control Center to apply the change. In addition, after you disable the option, the **Repository Synchronization** status might be in an **Error** or **Warning** state. This is a temporary condition and the status should return to **OK** after several minutes.

3.2.7 Command Queue Service (CQS) Options

To view the command queue service options, on the **Main** tab, in the **Tools** group, click **Options**. In the Options window, click the **Command Queue Service** link.

The **Command Queue Service** link displays four read-only parameters for the command queue service. To change these parameters, you must log on to the computer where the command queue service is running, stop the service, modify the NQCQS.exe.config file, and then restart the service. Before you edit the file, contact Technical Support. For more information about editing the file and

additional parameters that are not visible in the Control Center console, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

3.2.8 Global Preferences Options

To configure the Global Preferences options, on the **Main** tab, in the **Tools** group, click **Options**. In the Options window, click the **Global Preferences** link to configure the following options:

- Audit Settings

    This option specifies how long to store historical information in certain tables in the repositories (for example, the `ArchiveQueue` table in the Control Center repository). For more information, see the Help.

- Data

    This option specifies how long to store the data points that Knowledge Script jobs collect in the AppManager repository (QDB). For more information, see the Help.

3.3 Identifying the Computers to Monitor

To manage the servers and workstations in your environment from the Control Center console, you first need to identify the computers and the applications on them that need to be monitored. The process by which AppManager is made aware of the computers and applications you want to monitor is called discovery.

Discovery typically involves the following tasks:

- Installing the AppManager agent on the server or workstation you want to manage.
- Adding the server or workstation to the master view of a QDB.
- Running one or more specialized Discovery Knowledge Scripts from a management group based on the **Master** view.

When you manage Windows computers, the Deployment Server performs all of these scenarios for you, if the management server and QDB server are running and you have network access to them.

You can also run Discovery Knowledge Scripts at any time after you add servers and workstations to a view. Run Discovery Knowledge Scripts from a management group that is based on the **Master** view.

For more information about running Knowledge Scripts on managed resources, see Chapter 5, "Running Monitoring Jobs," on page 67.

3.4 Adding a Computer to the Master View of a QDB

You can manually add computers from the Control Center console to the master view of any QDB that is connected to the CCDB.

To add computers to a QDB:

1 Log on to the Control Center console with an account that has permissions to modify a management group.

2 On the **Global Tasks** tab of the ribbon, click **Add Computers** to start the Add Computer wizard.

3 In the Add Computer wizard, click **Select Repository Member**.

4 Select the QDB and view where you want to add the computer and click **Next**.

5 Select a tab to choose the type of computer you want to add (Windows or UNIX) and then type the names or IP addresses of the computers you want to add in the **Computers** field. You do not need to include leading backslashes [\\]. To add multiple computer names, separate each name by a comma.

To...	Do this...
Select a computer type	Select a tab to choose the type of computer you want to add: ◆ **Windows computers** This tab allows you to add computers that run on supported Windows operating systems. This is the default. ◆ **UNIX computers** This tab allows you to add computers that run on supported UNIX or Linux operating systems.
Add computers	Type the names or IP addresses of the computers you want. At this time, only IPv4 addresses are supported. To add multiple computer names, separate each name by a comma. To add computers that are not accessible by NETBIOS name from the Control Center console computer, type the name or IP address. If you attempt to add a computer that is not accessible from the Control Center console computer, the Control Center console does not automatically add the computer. If the computer you want to add is not accessible from the Control Center console, the Add Computers Wizard Results window lists the inaccessible computers.
Discover objects automatically	This option is available when you add a Windows, UNIX, or Linux computer. ◆ Click **Discover Windows objects automatically** to discover Windows resources. Selecting this option is the same as running the Discovery_NT Knowledge Script on the computer. ◆ Click **Discover UNIX objects automatically** to discover UNIX and Linux resources. Selecting this option is the same as running the Discovery_UNIX Knowledge Script on the computer. **NOTE:** You might want to run other discovery Knowledge Scripts on the computer after adding it to the management group.
Add computers even if they are down	This option is available when adding a Windows, UNIX, or Linux computer. When you select this option, the Control Center console adds the computer to the console and the management server continues to monitor for unavailable computers and add them to the QDB once they become available. You can also select this option to add computers that are behind a firewall.

6 Click **Finish**.

If you entered a valid server or workstation name that can be reached over the network, it is added to the QDB. If you specified a NETBIOS name, it is displayed in uppercase.

If you entered the name of a computer that is not valid or is not accessible by the CCDB, the Add Computers Wizard Results window displays a list of the computers you were unable to add. Click **Print/Export/Email** to print the list of computers, export the list to a variety of file formats, or email the list as an attachment in a variety of file formats. Once you close the Add Computers Wizard Results window, the list of computers you were unable to add is no longer available.

3.5 Viewing Detailed Information about Discovered Resources

In a **Servers** view for a physical computer, use the **Details** tab of the **Server Information** pane to view discovery details for a discovered resource.

To view discovery details for discovered resources:

1 In the Control Center console, expand the **Enterprise Layout** view of the **Navigation** pane to select a management group that contains the physical computer on which the resource was discovered.

2 In the **Servers** view, click the computer.

3 On the **Details** tab of the **Server Information** pane, expand the list of details to view discovery information about the computer.

3.6 Viewing AppManager System Information

In a **Servers** view, use the grid to view configuration details for a selected server.

The configuration details vary depending on the columns that are displayed in the grid. For information about changing the columns that are displayed, see "Working with Columns" on page 38.

3.7 Monitoring the Status of Control Center

The **System Status** pane displays information about the following items:

 • Health of the command queue service

 • Synchronization with the primary QDB

 • Availability of all managed QDBs

 • Control Center console commands in the Queue Manager

 • Status of deployment tasks

To display the **System Status** pane, on the **Main** tab, in the **View** group, click **System Status Pane**.

3.7.1 Monitoring Component Status

The **Component Status** group displays the status of the command queue service, synchronization between the Control Center console and any QDBs, and the availability of managed QDBs. The status for these components is also visible in the status bar at the bottom of the Control Center console.

The possible status conditions are:

 • OK (green flag)

 • Warning (yellow flag)

 • Error (red flag)

If the command queue service is down for 1 - 2 minutes, it displays a Warning status. If the service is down for three minutes or longer, it displays an Error status. The same intervals and status conditions apply to the availability of QDBs.

If a synchronization between the QDBs and the Control Center console has not taken place within the last two minutes, the Repository Synchronization status goes to Warning. If a synchronization has not taken place in five minutes or longer, the status goes to Error.

3.7.2 Monitoring Commands in the Queue Manager

The **Command Queue** group displays the commands that are waiting to be run by the command queue service (for example, to acknowledge an event). The command status can be one of the following:

Permission Denied

The user lacks proper security access.

New

The command is in queue, but not yet picked up by the command queue service.

Pending

The command queue service picked up the command and the command is waiting for execution.

Success

The command queue service successfully executed the command.

Fail

The command failed for a particular reason.

Corrupt

The command queue service cannot decipher data in the command.

Aborted

The user requested cancellation and action is pending.

Abort Completed

The command was cancelled prior to execution.

To view details on a command in the list, double-click the command you want.

To view details on all commands in the queue, on the **Global Tasks** tab of the ribbon, in the **Administration** group, click **Launch Queue Manager**.

3.7.3 Monitoring Deployment Tasks

The **Deployment** group displays any deployment task with a status of **Waiting for Approval**, **Waiting for Schedule**, **Active,** or **Error**. Double-click an entry in the **Deployment** group to display the Deployment Summary view.

3.7.4 Monitoring Connectivity to QDBs

The **System Status** panel displays the connectivity status from the command queue service and the Cache Manager to the managed QDBs. The status may be **OK**, **Warning**, or **Error**.

3.7.5 Understanding Command Queue Service Status

The command queue service status can be one of the following:

OK

> The command queue service can connect to all QDBs.

Warning

> The command queue service cannot connect to one or more (but not all) QDBs. A **Warning** state also indicates the command queue service has not updated its health tables in the set interval of two minutes. For more information, see Section 3.2.7, "Command Queue Service (CQS) Options," on page 43.

Error

> All the QDBs that the command queue service is attached to are down, or some other significant problem has occurred—such as the command queue service has stopped responding. An **Error** state also indicates that the command queue service has not updated its health tables in the set interval of five minutes. For more information, see Section 3.2.7, "Command Queue Service (CQS) Options," on page 43.

3.8 Getting Information about a Knowledge Script

A short description of each Knowledge Script is displayed in the **Knowledge Scripts** view.

For detailed information about an individual Knowledge Script, double-click the script in the **Knowledge Scripts** view and then click **Help** on the **Values** tab in the Knowledge Script Properties window.

You can view the development history of a Knowledge Script, including the version number and build ID, by right-clicking the Knowledge Script in the **Knowledge Scripts** view and then clicking **Version History**.

3.9 Checking Communication with Agent Computers

From time to time, the **Agent Status** column in the **Servers** view might indicate that an agent computer is disconnected, which means the management server is unable to communicate with the computer. While in this state, you cannot run Knowledge Scripts against the computer. In the background, the management server periodically checks whether it can communicate with the computer. Once the management server reestablishes communication, the status of the agent computer returns to its normal Online state.

Instead of waiting for the management server's next polling interval, you can check the communication with an agent computer yourself. You might want to do this if you shut down a computer to perform some maintenance. When you reboot the computer, you can check the connection immediately rather than wait for the management server.

To check the status of an agent on an agent computer:

1 In the **Enterprise Layout** view of the **Navigation** pane, select the computer with the agent you want to check.

2 In the **Tasks** pane under **Server Tasks**, click **Request Agent Communication Status Verification**.

3 In the **System Status** pane, in the **Command Queue**, check the queue to see if the status request succeeded or failed.

If an agent computer's agent status remains unknown for more than twenty minutes, check for network communication problems and the availability of the management server, QDB, and the agent computer itself. If the availability of the agent computer isn't a network connection problem, you might need to use the `NetIQctrl` utility to track down the source of the problem. For more information about using `NetIQctrl`, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

3.10 Using the Utilities Menu Extensions

The Control Center console provides right-click menu access to a variety of tools, some of which are not part of AppManager, that can be useful to perform system management or diagnostic tasks. For more information about these tools, see the following sections:

- Section 3.10.1, "AppManager Utilities," on page 49
- Section 3.10.2, "Windows Utilities," on page 50

In addition, you can customize the AppManager Utilities menu. For more information, see Chapter 12, "Customizing Menu Extensions," on page 181.

3.10.1 AppManager Utilities

To access the AppManager utilities, in an **Events**, **Jobs**, or **Servers** view, right-click an item in the view and from the menu select **Utilities** > **AppManager Utilities**. The following utilities are available:

Chart Console

Starts the Chart Console, which allows you to generate and view charts of data streams that Knowledge Script jobs generated. For more information, see Chapter 11, "Collecting and Analyzing Data in Control Center," on page 165.

Diagnostic Console

Starts the Diagnostic Console, if it is installed on the Control Center console computer. The Diagnostic Console expands on the problem diagnosis features of AppManager by providing you with a collection of features for remotely investigating and diagnosing problems occurring on Windows servers in your environment. This tool is not applicable when monitoring a UNIX or Linux computer.

You must install the Diagnostic Console separately. The AppManager installation package includes the setup program.

List Jobs on Windows Agent via NetIQCtrl

Starts a command-line interface for checking the status of AppManager components. For more information, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

Operator Console

Starts the Operator Console, if it is installed on the Control Center console computer. Do not use the **Utilities** menu to start more than one instance of the Operator Console. The Operator Console does not work properly when running more than one instance on the same computer using the same login information.

The Operator Console allows you to manage computers on a single QDB. For more information, see the Help.

You can install the Operator Console from the AppManager setup program.

Security Manager

Starts Security Manager, if it is installed on the Control Center console computer. Security Manager enables you to manage user security for the Operator Console.

License Manager

Starts the License Manager, if it is installed on the Control Center console computer.

Repository Browser

Starts the Repository Browser, if it is installed on the Control Center console computer. The Repository Browser allows you to browse the records in the QDB and use standard SQL commands to write queries to retrieve the information from the database. For more information, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

3.10.2   Windows Utilities

To access the Windows utilities, in an **Events**, **Jobs**, or **Servers** view, right-click an item in the view and from the menu select **Utilities** > **Windows Utilities**. The following utilities are available:

Computer Management

Starts the Windows Computer Management console. This tool is not applicable when monitoring a UNIX or Linux computer. For more information, see your Windows documentation.

Event Viewer

Starts the Windows Event viewer. This tool is not applicable when monitoring a UNIX or Linux computer. For more information, see your Windows documentation.

Explore, Explore C$

Opens Windows Explorer and opens Windows Explorer to the `C:` drive, respectively.

Ping Machine

Performs an ICMP ping request on the computer you want.

Service Manager

Opens the Windows Service Manager on the computer you want.

Terminal Services

Starts a Terminal Service session on the remote computer you want.

Trace Route

Runs the Traceroute utility on the computer you want.

3.10.3 Using the Troubleshooter

The Troubleshooter utility provides access to many different types of diagnostic reports about AppManager management servers and agent services through an easy-to-use console interface. Through the Troubleshooter, you can retrieve information about management server and agent communication, detailed and summary job status, detailed operational statistics, and configuration details such as a computer's time zone setting and upload schedule.

The Troubleshooter is not applicable when diagnosing issues on UNIX and Linux computers.

To use the Troubleshooter:

1 In an **Events**, **Jobs**, or **Servers** view, right-click an item in the view pane and from the menu select **Utilities**.

   The context menu displays several troubleshooting options:

   ◆ **Job Troubleshooter**

   ◆ **Client Resource Monitor Troubleshooter**

   ◆ **Client Communication Manager Troubleshooter**

   ◆ **Management Service Info Troubleshooter**

   If a computer is not in the view pane, you can browse for it after you open the Troubleshooter window.

2 Click the troubleshooter option you want.

For more information about using the Troubleshooter, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

3.11 Viewing Audit Information

If you enable auditing for jobs, events, or actions, AppManager records information about every related operation in the QDB. For example, if you enable auditing for job-related operations, the audit trail includes chronological information about who started each job, changes to job properties, and changes to job status.

To access event or job audit information, select a server in a **Servers** view, select the **Events** tab or **Jobs** tab in the **Server Information** pane, and in the **Tasks** pane, click either **Event Audit Trail** or **Job Audit Trail**.

To view the maintenance history for a server, in the **Tasks** pane, click **Maintenance Mode Tasks > Computer Maintenance History.**

For more information about audit information, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

4 Managing a Group of Physical Computers or Logical Servers

Management groups are the means by which you organize and manage the discovered computers and resources in your managed environment. This chapter provides an overview of how to create and use management groups in the Control Center console to manage a group of physical computers or logical servers.

4.1 About Management Groups

Control Center uses management groups to manage a group of physical computers or logical servers. A management group consists of at least one **member** that specifies the managed objects that belong to that management group.

A management group member can be:

- A view in a QDB. For example, the NT view.
- A server group within a view of a QDB. Use the Operator Console to add a server group to a view. You cannot create server groups in the Control Center console.
- A rule that describes the physical computers or logical servers you want. For example, Exchange Servers with more than 2 GB of physical memory. For more information, see Section 4.3, "Working with Management Groups," on page 56.
- One or more agent computers that are part of the ad hoc computer list.

The Control Center console provides the following default management groups:

- **Master** displays all of the physical computers that Control Center manages. A computer can appear in more than one management group.
- **AM Logical Servers** is a container for the following management groups:
  - **All Logical Servers** displays all physical computers and logical servers that Control Center manages.
  - **Databases** displays all SQL Server and Oracle databases that Control Center manages.
  - **Network Devices** displays all network devices that Control Center manages.
  - **Virtual Infrastructure** displays the following resources that Control Center manages:
    - ESX hosts
    - Virtual machines
    - Datacenters
    - Clusters
    - Datastores
    - Datastore clusters

Unlike other management groups, the **AM Logical Servers** container does not display the cumulative worst open event status icon for its members. Instead, the sub-groups in the container display the icon.

A logical server can appear in more than one management group. In addition, if a logical server is a child object of a physical computer or another logical server and a management group also includes the physical computer or parent logical server, the child logical server appears as a top-level object and again as a child object of its parent. For example, in the **All Logical Servers** management group, the master database in the QDB appears as a top-level object and as a child object of the host SQL Server.

After you install and discover an agent, Control Center automatically displays the agent in the **Master** management group. If you configure Control Center to manage more than one QDB, the **Master** management group displays all of the managed computers in all of the QDBs.

Using Control Center's group-based permissions, you can configure permissions on a management group.

Configure additional management groups to select physical computers or logical servers based on the resources you want to monitor. After you identify the computers or logical servers, configure a monitoring policy to automatically monitor those resources. For more information, see Chapter 7, "Monitoring by Policy," on page 105.

4.2 About Rule-based Management Groups

A rule-based management group provides a powerful and dynamic way to manage a group of physical computers or logical servers by allowing you to automatically move objects between management groups as their status and discovered resources change. You can configure rules to compare discovered resources, such as the amount of physical memory, and AppManager-related status, such as the highest event severity level. A management group based on an AppManager view is not dynamic and also not as granular because once AppManager discovers the application resources, the discovered objects always appear in the management group.

Using the Create New Rule window, you can easily configure expressions to select the physical computers or logical servers you want. The following types of expression templates are available:

Application

Selects physical computers with an application or an application component installed on them. For example, computers where IIS is installed.

Computer List Match

Selects a list of physical computers that match a search string that you specify. For example, EXCHNewYork, EXCHLosAngeles, and EXCHSanJose.

Computer Name Wildcard Match

Selects a list of physical computers that match a wildcard search string you specify. For example, EXCH*.

AD

Selects Active Directory domain controller computers. For example:

- ◆ Domain controllers that match a wildcard search string you specify
- ◆ Domain controllers in domains with a specific domain mode
- ◆ Domain controllers in domains that match a wildcard search string you specify
- ◆ Domain controllers that have a specific role

- Domain controllers in domains that have an installed AppManager for Active Directory agent
- Domain controllers with a specific domain functional level
- Domain controllers in a specific forest
- Domain controllers in a specific site

AppManager

Selects AppManager managed physical computers. For example, computers:

- With the AppManager agent or report agent
- With a particular event severity
- Within a specific range of IP addresses
- Which are Proxy Event Servers
- That are included in a particular version of the QDB
- Where the agent has not recently communicated with the management server (grayed out)
- That are in maintenance mode
- With jobs that have an **Error** status
- With a specified threshold for number of running jobs

Custom Properties

Physical Computers

Selects objects based on an existing custom property on a physical computer, including the custom property type (floating point number, integer, string, and time) and value. For more information, see Section 4.5, "About Custom Properties," on page 61.

Hardware

Selects computers with specific hardware. For example, computers with more than two processors.

Logical Servers

Selects logical servers. For example:

- Selects objects based on an existing custom property on a logical server, including the custom property type (floating point number, integer, string, and time) and value. For more information, see Section 4.5, "About Custom Properties," on page 61.
- Selects logical servers according to the following criteria:
  - Matching component type, such as SQL Server
  - Matching component type, object type, and object detail, such as SQL Servers with databases created on a specific date
  - Matching component type and name, such as SQL Server databases with QDB in the name

NT

Selects Windows computers based on the version of Windows installed.

Object - Physical Computers

Selects physical computers with AppManager object information. For example:

- Physical computers with a particular application installed, such as Microsoft SQL Server

- Physical computers with a matching object type and object detail, such as all Windows computers with a logical disk of a specific size

- Physical computers with a matching object type and name, such as physical computers with a SQL Server database with New York in the name

- Physical computers that have similar or matching object details, such as a specific number of cylinders on the physical disk

SQL

Selects SQL server computers. For example, SQL server computers having:

- A specific database

- A specific database size

- A specific detail and value

- No custom databases installed

- More than one custom database installed

- A specific SQL server version

4.3 Working with Management Groups

Use the Control Center console to perform the following management group tasks:

- Create management groups

    You create management groups in the **Enterprise Layout** view of the **Navigation** pane. You can organize management groups into a hierarchy, and permissions you assign to the top-level management group in the hierarchy are inherited by the children of that top-level management group. For more information about assigning permissions to a management group, see Section 4.7, "Configuring Permissions for a Management Group," on page 63.

    When you create a management group, you can create the standard views for the management group, which include **Events**, **Jobs**, **Knowledge Scripts**, and **Servers** views.

    After you create a management group, you can create folders to organize additional **Events**, **Jobs**, **Knowledge Scripts**, and **Servers** views.

    To create a standard view, right-click the management group or folder where you want to create the view, click **New**, and then click the type of view you want to create.

    To change how columns appear in a view, right-click the view you want and click **View Properties > Columns**.

- Add and remove members. A member is an object that you want to include in the management group.

    You can create a management group without specifying any members. However, Control Center creates the management group without any standard views.

    You can configure the following types of members:

    Ad hoc computer list

    Allows you to add agent computers to the management group that are not part of a repository view, server group, or rule.

    Repository View

    Allows you to add one or more views from a QDB. For example, the NT and IIS views.

Server Group

Allows you to add an existing server group from a QDB. If you have not created server groups in the Operator Console, this option is not applicable. You cannot create server groups in the Control Center console.

Rule

Allows you to create and add rules that describe the physical computers or logical servers you want to add based on their discovered attributes. For example, number of processors or a custom property.

You can create a cascade of management groups up to nine levels.

TIP: When adding members to a management group, it is a good idea to limit the number of members in the management group. If you add more than one member to a management group, do not select an object more than once. If you select an object more than once and you run a policy-based job on that object, AppManager creates duplicate policy-based jobs. For example, if you add the NT view and a management group rule that selects physical computers based on discovered NT resources, AppManager creates duplicate policy-based jobs on computers that both members select.

When you remove a member from a management group, the associated objects do not appear in the management group. Existing jobs on those objects continue to run and you can view them from other management groups you configured to select the objects.

You can remove all of the members from a management group even if there are monitoring policy jobs running on the members.

You cannot remove the built-in Ad Hoc Computer List member, but the list can be empty.

◆ Add, modify, and remove rules

The Control Center console provides a set of predefined rules. You can use these rules as they are, modify them, copy them and modify the copies to make your own custom rules, or create new rules from the available set of expression templates. Predefined rules are available in the list of rules on the **Rule** tab of the Add Management Group Members window. This makes predefined rules easily available each time you want to define the membership for a management group.

The Rule Wizard provides an easy-to-use interface for adding and configuring expressions in a rule. You can find information about each expression template by hovering the mouse over an expression. Tooltip information helps you find the expression you need.

◆ Create expression templates

The Control Center console provides a set of predefined expression templates to use when defining rules. You can use these expression templates or copy them and modify the copies to make your own custom templates. You cannot customize the standard expression templates themselves.

◆ Copy management group

When you create a new management group, you need to adjust the following parameters for the newly created group:

1. Views
2. Column settings
3. Customization
4. View filters

By copying a management group, you can reuse previously customized settings of existing management groups by copying them instead of creating new ones.

You can copy an existing management group's UI instance along with its properties (except policy and members) and create a new management group. You can copy all views (**Event, Job**, **Knowledge Scripts**, **Service Map Views**, and **Servers**), column settings, customization, and view filters. The new management group is arranged alphabetically in the tree view pane of the **Enterprise Layout**. It has a default name consisting of the management group from which it is copied, followed by an underscore and time stamp. You may rename the newly created management group based on your requirement.

To create a management group:

1 Click the top-most AppManager icon or an existing management group in the **Enterprise Layout** view of the **Navigation** pane, and then in the **Tasks** pane click **Create New** > **Management Group**.

2 Enter properties for the management group, and then click **OK** to create the group.

For more information about the properties you can set, see the Help.

To add a member to a management group:

1 Right-click the management group, and then click **Management Group Properties** > **Members**.

2 (Conditional) To add computers to the ad hoc computer list, click **Ad Hoc Computer List**, and then click **Modify**.

  2a Click **Add**, and then select the management group that contains the computers you want to add.

  2b Select one or more computers you want to add, and then click **OK**.

3 (Conditional) To add objects to a management group based on a view, server group, or rule, click **Add**, and then select the method for adding members:

Repository View

Enables you to add objects based on a repository view, such as NT, SQL, or IIS.

Server Group

Enables you to add objects based on server groups. If you have not created server groups in the Operator Console, this option is not applicable. You cannot create server groups and views in the Control Center console.

Rule

Allows you to create and add rules that describe the physical computers or logical servers you want to add based on their discovered attributes. For example, number of processors or a custom property. For more information, see Section 4.3, "Working with Management Groups," on page 56.

4 Select the members to add to the management group, and then click **OK**.

To remove a member from a management group:

1 Right-click the management group, and then click **Management Group Properties** > **Members**.

2 Click the member you want to remove, and then click **Remove**.

To add a rule to a management group:

1 Right-click the management group, and then click **Management Group Properties > Members**.

2 Click **Add**, and then select the **Rule** tab.

3 (Conditional) To add an existing rule, select the rule you want, click **OK**, and then click **OK** again.

4 (Conditional) To create a new rule, click **Create New**.

5 Configure the rule. You can add multiple expressions to the rule.

For more information about configuring the rule, see the Help.

You can also create a new predefined rule. On the **Global Tasks** tab of the ribbon, click **Manage Rules**, and then click **Create New**. Configure the rule as described in the Help.

To remove or modify a rule from a management group:

1 Right-click the management group, and then click **Management Group Properties > Members**.

2 (Conditional) To remove a member, select an existing rule in the list of members, and then click **Remove**.

3 (Conditional) To modify a rule, select an existing rule in the list of members, and then click **Modify**.

4 Make your changes, and then click **OK**. Control Center automatically updates the management groups that use the rule.

You can also remove and modify predefined rules. On the **Global Tasks** tab of the ribbon, click **Manage Rules**, and then click **Modify** or **Delete**.

To create an expression template:

1 Right-click the management group, and then click **Management Group Properties > Members**.

2 Click **Add**, and then click **Rule**.

3 Click **Create New**, and then select the expression template that you want to modify.

4 Click **Copy**, and then configure the new expression.

For more information about configuring the expression, see the Help.

To copy a management group:

1 Go to the **Enterprise Layout** in the Control Center Console.

2 Select the management group that you want to copy.

3 Right click the management group and then click **Copy Management Group**.

4.4 Creating Standard Control Center Views

4.4.1 Setting the Default Filtering for a View

You can add local and server filters to a standard view. Adding server filters improves the overall performance of the Control Center console by reducing the amount of data returned from the CCDB as well as the amount of information that appears in the view. Local filters allow you to tailor the information available in the view to your specific needs. If necessary, the user can change the filtering in the actual view to hide additional information.

To add a filter to a view:

1 In the **Enterprise Layout** view of the **Navigation** pane, right-click the view you want and click **View Properties**.

2 On the sub-menu, click either **Local Filters** or **Server Filters**.

3 Apply the filters you want to the view. For more information about filters, see "Working with Lists" on page 38.

4 When you have finished applying filters, click **OK**.

4.4.2 Configuring the Server Information Pane

In a **Server** view, you can configure the **Server Information** pane to show the information you want, including:

- ◆ Events
- ◆ Jobs
- ◆ Details of discovery
- ◆ Custom properties
- ◆ Charts

You can also configure the tabs that display in the **Server Information** pane the same way you configure a standard Control Center view.

To configure the Server Information pane:

1 In the **Enterprise Layout** view of the **Navigation** pane, right-click the **Servers** view you want to modify and then click **View Properties > > General**.

2 On the **General** tab, select a check box to display the corresponding tab, and click **Properties** to configure the tab.

For...	Do this...
Events	Select **Show Events** to show the **Events** tab. In the **Server Information** pane, the **Events** tab functions like an **Events** view. To configure its properties, click **Properties**. In addition, you can configure the **Events** tab to: ◆ Show parent events. ◆ Show child events. This is the default. Note that this option enables the user to group by Parent Job ID. ◆ Show split parent/child events. The **Show event message on separate row** option enables you to view the event message in a separate row that is displayed directly under the event. This option is enabled by default.
Jobs	Select **Show Jobs** to show the **Jobs** tab. In the **Server Information** pane, the **Jobs** tab functions like a **Jobs** view. To configure its properties, click **Properties**.
Details	Select **Show Details** to show the **Details** tab.
Custom Properties	Select Show **Custom Properties** to show the **Custom Properties** tab. For more information, see Section 4.5, "About Custom Properties," on page 61.

3 Click **OK** to apply your changes.

To configure a tab in the Server Information pane:

1 Select the tab you want to configure.

2 Right-click on the tab and then click **View Properties > General**.

3 Configure the display and filtering options you want on the **General** tab and the remaining tabs. For more information about configuring a view, see the Help.

4 When you are finished configuring the view, click **OK**.

4.4.3 Deleting, Renaming, and Moving a Management Group or View

You can delete, rename and move management groups in the **Enterprise Layout** view of the **Navigation** pane. You can also delete and rename views in the **Enterprise Layout** view, and move views between folders within a management group. You cannot move a view from one management group to another.

4.5 About Custom Properties

Using the **Custom Properties** tab in the **Servers** view, you can manage custom property values for one or more selected physical computers or logical servers. You can create a new custom property value or apply an existing custom property value to the selected objects.

Using custom property information, you can:

◆ Override the parameter value for an ad hoc or monitoring policy job that runs on a physical computer. To do this, you must configure the job to use the custom property as a parameter override. AppManager only overrides the original job parameter value when the job runs on physical computers where you defined the custom property. If the job runs on logical servers where you defined the custom property, AppManager ignores the custom property.

After you add a custom property value to a physical computer and a job you configured to use the custom property as a parameter override runs on the computer, AppManager no longer uses the original job parameter value. You can also change the override value by updating the custom property value.

For more information about using job overrides, see Section 5.5, "Setting Override Values," on page 74.

◆ Configure a rule-based management group to select physical computers or logical servers based on custom property information.

You can only include physical computers with version 7.0 (or later) of the AppManager agent installed. If you have AppManager 6.0.2 (or earlier) agents, upgrade them to version 7.0 (or later) to include them in a rule-based management group that selects physical computers based on custom property information.

You can only include logical servers that are associated with version 8.2 (or later) QDBs. If you have logical servers that are associated with QDBs earlier than version 8.2, upgrade the QDBs to version 8.2 (or later) to include the logical servers in a rule-based management group that selects logical servers based on custom property information.

If you use one of the custom property expression templates that are available for logical servers to assign the same custom property to a physical computer and a logical server that is a child of the physical computer, the resulting rule-based management group contains both the physical

computer and the logical server as parent objects within the **Servers** view for the management group. If you expand the physical computer to view its child objects, the logical server also appears as a child object of the physical computer.

Any custom properties you add are available for use in the custom property expression templates. Add one or more custom properties to a physical computer or logical server to provide custom information about that object. For example, you can add a custom property to identify objects by geographic location.

4.6 Working with Custom Properties

To configure a rule-based management group to select physical computers or logical servers based on custom property information, first apply the custom property to the objects you want to include in the management group.

To apply a custom property to an object:

1 Select the object you want to have the custom property.

2 In the **Server Information** pane, click **Custom Properties**, and then, in the **Tasks** pane, click **Create New Custom Property**.

3 Configure the custom property, and then click **OK**.

   For more information about the fields, see the Help.

4.6.1 Applying a Custom Property Value to One or More Physical Computers or Logical Servers

You can apply an existing custom property value to one or more physical computers or logical servers or update selected physical computers or logical servers with a new custom property value.

To apply a custom property value:

1 In the view pane for a **Servers** view, select the physical computers or logical servers you want from the list.

2 Click the **Custom Properties** tab to view a list of custom properties for the selected objects.

   If one of the objects does not have the custom property, the value is **[Property Not Common]**. If the objects have a different value for the custom property, the value is **[Property Value Not Common]**.

3 Select the custom property you want, and in the **Tasks** pane click **Edit Custom Property Value**.

4 In the Edit Custom Property window, configure the custom property and click **OK** to apply the custom property value to the selected physical computers or logical servers.

4.6.2 Removing a Custom Property

You can remove a custom property from one or more selected physical computers or logical servers. For example, you can remove a custom property to prevent an object from appearing in a rule-based management group.

NOTE: When you delete a custom property from a physical computer, AppManager updates any jobs that are running on the computer and using the custom property value as a parameter override to use the default parameter value. For more information, see Section 5.5.5, "Removing a Job Override," on page 77.

To remove a custom property:

1 In the view pane for a **Servers** view, select the physical computers or logical servers you want from the list.

2 Click the **Custom Properties** tab to view the custom property information.

3 (Conditional) If you want to remove a custom property value from the selected objects, select a property from the list and on the **Tasks** pane click **Delete Custom Property Value**. This deletes the value of the custom property from the selected objects. The custom property definition remains in the QDB for use on other objects.

4 (Conditional) If you want to delete a custom property definition from the QDB, select a custom property from the list and on the **Tasks** pane click **Delete Custom Property Definition**. Deleting a custom property from the QDB means objects can no longer use it.

4.7 Configuring Permissions for a Management Group

You must be a member of the Control Center Administrator group to modify the members of a user group or modify permission sets, but you do not have to be an administrator to assign user groups and permission sets to management groups. For more information about Control Center security, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

4.7.1 Understanding Global Permissions

A global permission set is a permission set associated with a specific user group that applies to all management groups in the Control Center console. Since global permissions apply to all management groups, they do not depend on association with a specific management group to take effect.

The Control Center console has a default set of user groups, permission sets, and global permission sets. For more information about these default groups and permission sets and information about which global permission sets are associated with which user groups, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

4.7.2 Understanding Permission Inheritance

Because children of a management group inherit permissions you assign to that management group, any user group you assign to the management group will have the same permissions on any children of the management group. You do not need to assign user groups or permissions sets individually on each child management group. For example, if a user has permission to view and edit jobs on management group 1 and management group 1 has child management groups 1A and 1B, the user also has permission to view and edit jobs on child management groups 1A and 1B.

If you assign a global permission set to a user group and the permission set includes management group permissions, members of the user group will have those permissions on all management groups and child management groups in the Control Center console.

4.7.3 Granting Permission to Access a Management Group

You can grant one or more user groups access to a management group. You can also give the same user group access to a management group using different permission sets. If the same user belongs to more than one user group, Control Center applies the most restrictive set of permissions by combining the permissions with a logical OR. For example, if the same user is a member of two user groups associated with the same management group but with different permission sets, and you grant the user rights in one permission set but deny the same rights in the other permission set, then Control Center denies the rights. If a permission is undefined (you neither grant nor deny the permission) for the same user in two different user groups, then Control Center denies the permission. If you grant a permission for one user group and either do not define it or you grant it in another user group for the same user, then Control Center grants the permission.

To give a user group permission to access a management group:

1 Right-click the management group in the **Enterprise Layout** view of the **Navigation** pane and choose **Management Group Properties > Security**.

2 In the Management Group Properties window, click **Add**.

3 In the Assign Permissions window, select the user group you want from the **User Group** list.

4 Select the permission set you want to associate with the user group for this management group from the **Permission Set** list, and then click **OK**.

For information about the permissions that are defined in the default permission sets, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

5 (Optional) If you want to add more user groups, repeat these steps to add another user group.

NOTE

- To modify a permission set associated with a user group, click **Modify**. To create a new permission set for a user group, click **Create New**.

- To create a new user group, you can only do so in the Manage Security window. To open the Manage Security window, on the **Global Tasks** tab, in the **Administration** group, click **Manage Security**.

4.7.4 Removing Permission to Access a Management Group

If you want to deny all access to a management group for a user group that currently has access, you must remove that user group from the management group.

If a user group currently has access to a management group and you want to deny access for only some of the current permissions, you can assign the user group to the management group again using a permission set with the relevant permissions set to Deny. The resultant set of permissions for the members of the user group will be the result of a logical OR of all the permissions defined across all associated permission sets to produce the most restrictive set of permissions. If the user group is assigned to the management group with a permission set that grants the right to add computers and is also assigned to the management group with a permission set that denies the right to add computers, the members of the user group will not have the right to add computers for that management group.

When removing permission to access a management group, be aware of the following:

- Even though you might not see permissions assigned on a child management group, if the user group has permissions on the parent of that child management group, the user group has the same permissions on the child management group. To determine whether a user group has permissions on a child management group, you must look at the permissions on the parent management group.

- Even though you remove a user group from a management group, if the user group has a global permission set assigned to it and the permission set includes management group permissions, the user group can still access the management group.

To remove a user group from a management group:

1 Right-click the management group in the **Enterprise Layout** view of the **Navigation** pane and choose **Management Group Properties > Security**.

2 Click the user group you want to remove and then click **Remove**.

3 Click **OK**.

4.8 What's Next?

After you configure a management group with the computers you want to manage, you should:

- Configure a monitoring policy for the management group. For more information, see the Chapter 7, "Monitoring by Policy," on page 105.

- Give users access to the management group. For more information, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

- Decide to implement one or more **Service Map** views. For more information, see Chapter 8, "Using Service Map Views," on page 117.

5 Running Monitoring Jobs

This chapter provides an overview of working with jobs.

5.1 About Knowledge Scripts and Jobs

In AppManager, Knowledge Scripts, which are programs that run on agent computers (servers and workstations), facilitate all management and monitoring tasks. Knowledge Scripts can collect data, monitor for specific events, and perform specific actions in response to events. Each script performs a certain task or set of tasks and works on specific types of resources.

Knowledge Scripts are classified as **base** Knowledge Scripts or **derived** Knowledge Scripts. A base Knowledge Script is checked in to the QDB and holds all of the Knowledge Script settings (the schedule, parameters, and Advanced tab) and the script logic. A derived Knowledge Script is a copy of a base Knowledge Script (a direct copy or a member of a Knowledge Script Group) and shares the logic with the base Knowledge Script but has its own settings. For example, a base Knowledge Script might have a default threshold of 10 for a certain parameter while the derived Knowledge Script has a default threshold of 20. The two Knowledge Scripts share the script logic, but the settings are different.

Because a base Knowledge Script and a derived Knowledge Script share logic, when the logic for a base Knowledge Script changes, AppManager can automatically update the logic for derived Knowledge Scripts without updating the settings. This option (**Automatically update Derived KS properties for checked in Base KSs**) is enabled by default for new installations. For more information about enabling or disabling the option, see "Knowledge Scripts Options" on page 42.

The automatic update feature includes monitoring policy jobs but not ad hoc jobs. For ad hoc jobs, when the Knowledge Script that you used to create the job changes (whether it is a base script or a derived script), you must manually propagate the script changes to the job. For example:

1. You copy the base script NT_CpuLoaded to create the derived script NT_MyCpuLoaded.
2. You create an ad hoc job from NT_MyCpuLoaded.
3. NT_CpuLoaded is updated.
4. With the automatic update feature enabled, the derived script NT_MyCpuLoaded is automatically updated. However, the ad hoc job that you created from NT_MyCpuLoaded is not, so you must manually propagate the updated NT_MyCpuLoaded to the ad hoc job.

With both automatic and manual propagation, the target Knowledge Script or job retains its settings (such as thresholds and schedule). If a base Knowledge Script includes new values or removed values, AppManager merges those changes to derived Knowledge Scripts during the propagation process. For more information about manual propagation, see "Manually Propagating a base Knowledge Script to Derived Knowledge Scripts and Ad Hoc Jobs" on page 88.

A job is an instance of a Knowledge Script in the primary QDB that has been replicated to the QDB running the job on agent computers with a matching resource. That is, when you run a Knowledge Script you are creating a job.

Knowledge Scripts are organized into categories that are loosely connected to management groups in the Control Center console. For example, SQL Knowledge Scripts that check database statistics are available in management groups that include the **SQL** and **Master** views. You can apply General Knowledge Scripts to all Windows-based views for generic management of supported Windows platforms, and action Knowledge Scripts let you initiate actions in any view.

To...	Do this...
Select a Knowledge Script category	Click a **Knowledge Scripts** view and in the view pane, expand a Knowledge Script category.
Get detailed information about a Knowledge Script	Double-click a script in the **Knowledge Scripts** view and on the **Values** tab, click **Help**.

The individual Knowledge Scripts displayed in any Knowledge Script category vary depending on the resource objects actually discovered. For example, if printers have not been discovered in the environment, then the PrinterHealth Knowledge Script will not display in the NT Knowledge Script category.

5.1.1 Running Jobs to Pinpoint Problems

Most Knowledge Script jobs are run on a regular basis to keep continuous or periodic watch over system resources. To monitor resources on a continuous basis, configure a monitoring policy. For more information, see Chapter 7, "Monitoring by Policy," on page 105.

In some cases, you need to run a combination of scripts to pinpoint a problem. As an example, assume you run only the Exchange_ServerHealth Knowledge Script on a particular resource. If the Knowledge Script sends you an alert indicating the server is overloaded, you might need to run additional Knowledge Scripts, using the run once schedule to determine the cause of the problem.

By running these additional Knowledge Scripts, you get a better sense of the activity on the resource and the potential sources of the problem you are investigating. To carry this example further, if you discover that a server's CPU is busy, you can run the NT_TopCPUProcs Knowledge Script to pinpoint the processes that are consuming the CPU time.

Once you identify the problem source, there are several things you can do to take appropriate action. For example, in this case you might:

 ◆ Spread out the CPU load so that the top CPU processes and users run at different times to reduce system load.

 ◆ Change the computer's hardware configuration. In some cases, a computer might not have adequate or appropriate hardware for the tasks it runs. For example, there might be insufficient RAM or disk space. You can determine this by examining the details of hardware resource objects.

When jobs that check general system health or a number of different resources report event conditions, investigate the event detail message and consider running additional, more specialized jobs to find the real source of the problem and the best solution.

5.1.2 Parent and Child Jobs

When you create a Knowledge Script job, AppManager creates a parent job and a child job on each agent computer. The number of child jobs AppManager creates corresponds to the number of objects selected on the **Objects** tab of the Job Properties window. For example, if you run the Knowledge Script on one computer, AppManager creates one child job; if you run it on several computers, AppManager creates a child job for each one.

In the Control Center console, you can group child jobs by Parent Job ID. You can also select default **Jobs** view settings to group jobs by Parent Job ID, to show all jobs in a flat view, or show only child jobs in a flat view. Grouping child jobs by parent job enables you to modify all the child jobs under a parent job simply by modifying the parent job. Modifying the properties of a single child job requires that you update each child job individually.

To monitor the overall health of your environment, configure a monitoring policy. One advantage of implementing a monitoring policy is that you can automatically configure changes to policy-based child jobs. For more information, see Chapter 7, "Monitoring by Policy," on page 105.

5.1.3 Adding a Child Job to an Existing Parent

You can add additional child jobs to an existing parent job. This allows you to run an existing job on additional resources in your environment without needing to start a new parent job to include the new resources. You can only add child jobs to ad hoc parent jobs, and you can only include resources that are managed by the same QDB that contains the parent job. Also, you can only add resources from the management group containing the **Jobs** view you are using to add a child job.

To add a child job to a parent job:

1   In the **Enterprise Layout** view of the **Navigation** pane, select the **Jobs** view containing the parent job where you want to add a child job.

2   Right-click the ad hoc parent job you want and then click **Add Child Jobs**.

3   Select the resources where you want to run the Knowledge Script job, and then click **OK**.

5.1.4 Planning Which Knowledge Scripts to Run

Each Knowledge Script runs on certain types of objects that correspond to the types of resource objects you see on the **Details** tab of the **Server Information** pane. AppManager handles all of the appropriate type matching so you never have to worry about running the wrong type of Knowledge Script on a computer or resource.

In deciding which Knowledge Scripts to run, consider the critical resources in your environment, how servers and applications are distributed, potential system bottlenecks, and the breadth and depth of monitoring you want to do.

5.2 Starting a New Job

This section provides detailed instructions for starting new Knowledge Script jobs.

To start a job:

1   In the Control Center console, expand the **Enterprise Layout** view of the **Navigation** pane to select a management group.

2   In the **Servers** view, select a physical computer or logical server.

To select multiple computers or logical servers, press **Ctrl** or **Shift** while making your selections.

3 In the **Tasks** pane, click **Job Tasks > Create New Job**.

> **NOTE:** You can also start a job by dragging a Knowledge Script from the **Knowledge Scripts** view pane onto a physical computer or logical server in the **Enterprise Layout** view of the **Navigation** pane.

4 In the Select Knowledge Script window, select the Knowledge Script you want, and then click **OK**.

If you are not sure which Knowledge Script you want, you can click a Knowledge Script to display a short description.

5 In the Properties window, enter properties for the Knowledge Script. In many cases, you can simply use the default schedule and values. To make changes:

Click this tab...	To...
Schedule	Change the default schedule for running the job. For more information, see Section 5.3, "Setting the Schedule," on page 70.
Values	Adjust the threshold and other parameters for the job. For more information, see Section 5.4, "Setting the Values to Monitor," on page 72 and Section 5.5, "Setting Override Values," on page 74.
Actions	Specify an action you want AppManager to take. For more information, see Section 5.6, "Specifying One or More Corrective Actions," on page 79.
Objects	Modify the resource objects you want to the job to monitor. For more information, see Section 5.7, "Selecting Objects to Monitor," on page 82.
Advanced	Specify whether to filter temporary event conditions and whether to collapse duplicate events for this job into a single event or automatically acknowledge them. Additionally, you can specify data collection options for this job. For more information, see Section 5.8, "Setting Advanced Job Properties," on page 82.
Discovery	Specify the discovery options you want to set for the Knowledge Script. For more information, see Section 5.9, "Setting Discovery Options," on page 85. The tab applies to discovery Knowledge Scripts only.

Any changes you make only affect this instance of the Knowledge Script job. The changes do not affect the default Knowledge Script properties. For information about copying a base Knowledge Script to create a derived script with customized default properties, see "Copying a Knowledge Script" on page 87.

6 After you finish reviewing or changing the parameters, click **OK** to start the jobs.

5.3 Setting the Schedule

Although all Knowledge Scripts have a default schedule, you can modify the schedule for any Knowledge Script job as needed. You can specify when the job should start, the interval (if any) at which the job should run, and when the job should stop. You can also schedule jobs to run on specific days and at set times, or to run only once. To prevent jobs from running on a Windows computer during scheduled maintenance periods, run the AMADMIN_SchedMaint Knowledge Script. On a

UNIX agent, use the AMAdminUNIX_SchedMaint Knowledge Script. For more information, double-click the AMADMIN_SchedMaint Knowledge Script in the **Knowledge Scripts** view and click **Help** on the **Values** tab in the Properties window.

To set the schedule for a job:

1 In the **Enterprise Layout** view of the **Navigation** pane, select a **Jobs** view or on the **Jobs** tab of the **Server Information** pane, select the job you want to modify.

2 In the **Tasks** pane, click **Job Properties**.

3 Click the **Schedule** tab.

NOTE: The options for setting intervals, start and stop times, and daily frequency depend on the selected schedule.

---

4 Select a schedule type.

If you select...	Then...
Run once	Set a start time for the job to run once (see Step 5 on page 72).
Regular intervals	Set the interval (**Every**) — in hours, minutes, or seconds — and start and end times for the job (see Step 6 on page 72).
	The interval controls how many times the job runs between the starting and ending time. For example, if you set a 10-minute interval with a start time of 10:00:00 PM and an end time of 10:59:00 PM, the job runs every 10 minutes — for a total of six times. (If the job takes more than 10 minutes to run, then it will run fewer times. For example, if the job takes 15 minutes to run, then it will run at 10:00:00 PM, 10:20:00 PM (the next scheduled interval), and 10:40:00 — for a total of three times.)
X number of times	Set how often you want the job to run (**Every**) — in hours, minutes, or seconds. Set how many times you want the job to run (**End after ___ Times**).
	Then set a start time (see Step 5 on page 72.).
Daily schedule	Set the daily interval (**Every**) you want the job to run.
	Then set start and end dates (see Step 5 on page 72) and the frequency (see Step 6 on page 72).
Weekly schedule	Set the weekly interval (**Every**) and the days of the week you want the job to run.
	Then set start and end dates (see Step 5 on page 72) and the frequency (see Step 6 on page 72).

If you select...	Then...
Monthly schedule	Set the monthly interval (**Every**) and the days of the month you want the job to run (**On**).

Depending on the option you select in the **On** list, you can further refine your selection. For example, if you select a day of the week (such as Monday), you can further select the weeks in the month (such as **The first Monday** and **The last Monday** of the month). If you choose **Selected Day**, you can then select days in calendar format (such as 3 and 17, or **L** for the last day of the month).

Then set start and end dates (see Step 5 on page 72) and the daily frequency (see Step 6 on page 72).

NOTE: If you configure an AppManager job to run monthly on a particular weekend (such as first weekend, second weekend, etc.), a weekend constitutes a consecutive Saturday and Sunday that falls within the calendar month. |
| **Asynchronous** | This option is selected for Knowledge Scripts that run asynchronously to monitor resources as changes occur. |

5 Set start and end date/times for the job. Available options depend on the schedule type you selected in Step 3 on page 71.

For...	Select...
Start (date or time)	**Now**, or a date and time.
End (date or time)	**No end date**, or a date and time.

6 Set the frequency for the job.

To run the job...	Select...
Once during the day	**Once for the selected period** and then set a start time.
At set intervals during the day	**Every**. Then set the interval — in seconds, minutes, or hours — and a start and end time.

7 Click another tab to change its properties or click **OK** to start the job.

5.4 Setting the Values to Monitor

Knowledge Scripts have default threshold values or other parameters that control what they check and how they behave. The parameters associated with Knowledge Scripts vary, but typically include:

◆ What the Knowledge Script does (for example, notifies you of event conditions, collects data for graphs and reports, or attempts to automatically restart services).

◆ The threshold values to monitor (for example, less than 10% free disk space or CPU utilization greater than 90%).

◆ The files, messages, and executables to monitor (for example, a list of processes that should be running, specific error messages in a log file, specific applications for which to monitor memory usage).

You can modify the parameter values as needed.

You can also add comments to ad hoc jobs. Job comments enable you to organize and group ad hoc jobs and provide useful information about the ad hoc job to other console users. This option is not applicable for policy-based jobs or Knowledge Script Group-based jobs.

If you are creating an ad hoc job to monitor computers across QDBs, you can configure different parameter values for each QDB. However, you must create policy-based jobs with the same parameter values. To resolve this issue, you can configure override values or configure different management groups for each repository.

To create an ad hoc job that runs on agent computers in different repositories, the primary QDB and each of the non-primary QDBs must be running. You can create a policy-based job on agent computers in different QDBs when one of the non-primary QDBs is not running. AppManager will create the policy-based jobs when the QDB is back online.

For more information, see Chapter 7, "Monitoring by Policy," on page 105.

To modify the parameter values of a job:

1 In the **Enterprise Layout** view of the **Navigation** pane, select a **Jobs** view.

2 In the view pane, select the job you want to modify (or on the **Jobs** tab of the **Server Information** pane, select the job you want to modify).

3 In the **Tasks** pane, click **Job Properties**.

   NOTE: You cannot modify the parameter values of a job that is part of a monitoring policy. To modify the parameter values of a monitoring policy job, modify the associated Knowledge Script Group member.

4 Click the **Values** tab.

5 (Optional) To specify a comment for the job, enter a comment in the **Comment** field. This option is not applicable for policy-based jobs or Knowledge Script Group-based jobs. For more information, see Section 5.11.6, "Viewing Job Comments," on page 91.

6 (Optional) If you are monitoring more than one QDB with Control Center, you can configure different parameter values for each QDB by selecting the **Adjust parameters separately for each repository** option.

   If you select this option, you can choose the QDB you want and configure the parameter values for all resources that QDB manages. This option is not applicable for policy-based jobs or Knowledge Script Group-based jobs.

7 Click in the **Value** column to change threshold or monitoring parameter values. Some fields have a browse (**...**) button you can use to make selections.

   For specific information about the current Knowledge Script parameters, click **Help**.

8 Click another tab to change its properties or click **OK** to start the job.

5.5 Setting Override Values

As you gain experience in monitoring your environment, you might find that you want to use a custom property as an override value for a Knowledge Script parameter rather than changing the parameter value for individual jobs. AppManager only overrides the original job parameter value when the job runs on physical computers where you defined the custom property. If the job runs on logical servers where you defined the custom property, AppManager ignores the custom property.

After you add a custom property to a physical computer and a job you configured to use the custom property as a parameter override runs on the computer, AppManager no longer uses the original job parameter value. You can also change the override value by updating the custom property value.

Configuring overrides to a monitoring policy is particularly useful because initially you want to use the same parameter values for all policy-based jobs but over time, you might need to adjust monitoring thresholds for some physical computers in the policy.

You can configure a parameter override for an ad hoc job, but NetIQ Corporation does not recommend this because you can simply change the parameter value of an ad hoc job.

For more information, see the following topics:

- Section 5.5.1, "Getting Started with Overrides," on page 74
- Section 5.5.2, "Enabling an Override for a Parameter," on page 75
- Section 5.5.3, "Configuring an Override Value," on page 76
- Section 5.5.4, "Viewing the Override Value for a Parameter," on page 77
- Section 5.5.5, "Removing a Job Override," on page 77

5.5.1 Getting Started with Overrides

To use a parameter override on a policy-based job, configure a Knowledge Script Group member to include a custom property as an override for a parameter value. This allows you to assign a custom property value as an override value for the parameter on each physical computer. AppManager will use the default parameter value until you add a custom property value to one or more physical computers.

In the **Servers** view, select the physical computers you want and in the **Custom Properties** tab of the **Server Information** pane, add a custom property value that corresponds to the custom property. For more information, see Section 4.5, "About Custom Properties," on page 61.

For example, if you want to monitor total CPU usage, you can configure the NT_CPULoaded Knowledge Script with a default value for the **Severity - Total system CPU** parameter, and add a custom property, **CPUEventOverride**, as an override. When the policy-based job starts, AppManager will monitor the total CPU usage using the default parameter value. If you notice that the total CPU usage on a particular physical computer tends to exceed the threshold, add the **CPUEventOverride**

custom property to the computer and specify a value you want to use as the override value. If the same policy-based job monitors more than one physical computer, you can add the custom property value to multiple computers at the same time.

If you remove a custom property that is used as a parameter override from a physical computer, AppManager restarts any jobs on the computer that use the custom property to use the default parameter value.

5.5.2 Enabling an Override for a Parameter

To override a parameter value on a policy-based job, specify a custom property override for the parameter without specifying a value for the override. After you enable the override on the Knowledge Script Group member, you can add a custom property value to a physical computer to override the default parameter value.

To enable an override for a parameter:

1 In the **Enterprise Layout** view of the **Navigation** pane, select the **Knowledge Scripts** view containing the Knowledge Script Group with the group member you want to modify.

2 In the view pane, select the Knowledge Script Group, and then in the **Tasks** pane click **Knowledge Script Group Properties**.

3 In the Knowledge Script Group members list, select the Knowledge Script you want to configure with an override, and then click **Properties**.

4 In the Job Properties window, select the **Values** tab.

5 On the **Values** tab, click the override selection (**...**) button on the value where you want to assign an override.

6 (Conditional) If you want to use an existing custom property, in the Select Override Value dialog box, click **Select** and then select the custom property you want from the list. The list of available custom properties corresponds to custom properties with a matching data type. For example, if you want to override a parameter value that uses an integer, AppManager only displays custom properties with a data type of **Integer**.

Although custom properties you defined for logical servers might appear in the list of available properties, selecting a custom property for a logical server will have no effect on the job. For the purpose of overrides, AppManager ignores custom properties on logical servers.

7 (Conditional) If you want to create a new custom property, click **Create New**, and then provide a name for the new custom property.

At this point, the custom property does not have a value and AppManager uses the default parameter value.

8 Click **OK** to close the Select Override Value window.

9 On the **Values** tab of the Properties window, AppManager updates the **Override** column to indicate that you configured a custom property override. Note that this indicator does not specify whether you configured an override value.

To override the parameter value, add the custom property with an override value to the physical computer. For more information, see Section 5.5.3, "Configuring an Override Value," on page 76.

10 (Conditional) If you configured an action for a Knowledge Script and you want to enable an override for a parameter of the action, click the **Action** tab to configure exceptions for the action Knowledge Script.

10a On the **Action** tab, click the action you want and click **Properties**.

10b In the **Override** column, click the override selection (**...**) button on the value where you want to assign an override.

10c (Conditional) If you want to use an existing custom property, in the Select Override Value window, click **Select** and then select the custom property you want from the list. The list of available custom properties corresponds to custom properties with a matching data type. For example, if you want to override a parameter value that uses an integer, AppManager only displays custom properties with a data type of **Integer**.

Although custom properties you defined for logical servers might appear in the list of available properties, selecting a custom property for a logical server will have no effect on the job. For the purpose of overrides, AppManager ignores custom properties on logical servers.

10d (Conditional) If you want to create a new custom property, click **Create New**, and then provide a name for the new custom property.

NOTE: AppManager always uses the custom properties on the physical computer to configure the override value for an action Knowledge Script. If you configured an action to run on the management server, configure the custom property for the action on the physical computer. Action Knowledge Scripts do not use custom property exceptions on the management server.

---

11 Click another tab to change its properties or click **OK** to start the job.

5.5.3 Configuring an Override Value

Once you configure a Knowledge Script Group member to use a custom property as a parameter override, you can configure an override value for that parameter on one or more physical computers. For more information about configuring a Knowledge Script parameter to use a custom property override, see Section 5.5.2, "Enabling an Override for a Parameter," on page 75.

To specify an override value for a parameter:

1 In the **Enterprise Layout** view of the **Navigation** pane, select the physical computer where you want to assign a custom property value to use as a Knowledge Script parameter override.

2 In the **Server Information** pane, select the **Custom Properties** tab.

3 In the **Tasks** pane, click **Create New Custom Property**.

4 In the **Name** field, select the custom property you want.

5 In the **Type** field, select the data type for the custom property. The data type must match the data type defined for the custom property assigned as an override.

6 In the **Value** field, specify the value you want to use as an override to the default parameter value.

7 Click **OK**.

5.5.4 Viewing the Override Value for a Parameter

Use the **Values** tab of the Job Properties window to view the override value for a parameter. Keep in mind:

- The **Value** column always displays the parameter value. If you configure a custom property override, you also need to configure an override value. If the override is enabled but does not have a value, AppManager uses the parameter value from the **Values** tab. If you configured an override value, AppManager uses the value of the override, not the parameter value from the **Values** tab.

- The **Override** column indicates whether you configured an override value, but does not display the actual override value. Keep in mind that you can configure a parameter override without an override value. In this case, AppManager uses the parameter value from the **Values** tab.

To view the override value for a parameter:

1 In the **Enterprise Layout** view of the **Navigation** pane, select the **Jobs** view of the management group with the job you want.

2 In the view pane, select the job you want.

3 In the **Tasks** pane, click **Job Properties**.

4 Select the **Values** tab.

5 Click the override selection (**...**) button.

6 In the Select Override Value window, the **Current Value** field displays the value for the override on the physical computer where the job is running.

7 (Optional) If you want to see the value of the same custom property on other physical computers and logical servers, click **Values On All Computers**. The All Values of Custom Property window displays the server name and value for every occurrence of this custom property on other objects. The list is empty if the custom property is not used on any other objects.

5.5.5 Removing a Job Override

You can remove a job override by:

- Removing the override from the monitoring policy. After you remove the override from the monitoring policy, the policy-based jobs will automatically use the default parameter value.

- Removing the custom property from the physical computers where you do not want to override the default parameter value. Note that if you restore the custom property to the computer and you configured the job with an override, the job automatically updates to use the custom property value. After you remove an override, AppManager uses the default parameter value.

5.5.6   Removing an Override from a Monitoring Policy

To update a monitoring policy to remove an override, remove the custom property from the Knowledge Script Group member. After you update the Knowledge Script Group member, AppManager updates the corresponding policy-based jobs to use the default parameter value, and the custom property value remains on the corresponding physical computers.

To remove an override from a monitoring policy:

1 In the **Enterprise Layout** view of the **Navigation** pane, select the **Knowledge Scripts** view containing the Knowledge Script Group with the group member where you want to remove an override.

2 In the view pane, select the Knowledge Script Group, and then in the **Tasks** pane click **Knowledge Script Group Properties**.

3 In the Knowledge Script Group members list, select the Knowledge Script where you want to remove an override, and then click **Properties**.

4 In the Job Properties window, select the **Values** tab.

5 Click the override selection (**...**) button for the parameter where you want to remove the override.

6 In the Select Override Value window, click **Remove**.

7 Click **OK** to save your changes.

Click **OK** again to close the Knowledge Script Group Properties window. AppManager restarts the associated policy-based jobs with the default parameter value.

5.5.7   Removing a Custom Property Value from a Physical Computer

If you want to use the default value for a parameter on a particular physical computer without removing the override for that parameter on the Knowledge Script Group member, you can remove the corresponding custom property value from the computer. When removing a custom property from a physical computer, keep in mind that other jobs on the same computer might be using the custom property value.

To remove a custom property from a physical computer:

1 In the **Enterprise Layout** view of the **Navigation** pane, select the **Servers** view with the computer where you want to remove the custom property value.

2 In the **Servers** view, select the computer where you want to remove the custom property value.

3 In the **Custom Properties** tab of the **Server Information** pane, select the custom property you want to remove, and then click **Delete Custom Property Value**.

AppManager automatically restarts any jobs on the physical computer that used the custom property value as an override to use the default parameter value.

NOTE: You have the option to remove the custom property definition by clicking **Delete Custom Property Definition**. Deleting the custom property definition removes the custom property from the QDB and from every physical computer and logical server where you configured it. AppManager restarts any jobs using the custom property to run with default values. If the custom property is part of a filter for any **Servers** views, AppManager removes that filter from the view.

If you are not sure which custom property value you need to remove, or if you are not sure whether a custom property is used more than once, you can use the Select Override Value window to determine where you configured the custom property and which jobs are using it.

To determine what jobs and physical computers are using a custom property:

1 In the **Enterprise Layout** view of the **Navigation** pane, select a **Servers** view containing a physical computer where a job that is using the custom property you are investigating is running.

2 In the **Server Information** pane, select the **Jobs** tab.

3 On the **Jobs** tab, select a job using the custom property you are investigating.

4 In the **Tasks** pane, click **Job Properties**.

5 In the Job Properties window, select the **Values** tab.

6 Click the override selection (**...**) button for the parameter where you are using the custom property as an override.

7 In the Select Override Value window, click **Jobs/Knowledge Scripts Using Same Override**.

8 The All Jobs/Knowledge Scripts Using Same Custom Property Override window shows all jobs and Knowledge Scripts using the custom property, including the Job ID, Knowledge Script name, computer name, and override value.

5.6 Specifying One or More Corrective Actions

Almost all Knowledge Scripts can initiate responsive actions, such as sending e-mail, running an external program, or sending an SNMP trap. You define the conditions under which to perform the action and what action to take. For example, if you want to receive an e-mail when CPU utilization reaches 75% on a server, create a job that checks for a CPU threshold of 75% and specify an e-mail action.

Actions appear in the **Actions** category of the **Knowledge Scripts** view. The prefix "UX" identifies actions that AppManager can run on a managed UNIX or Linux computer (for example, Action_UXCommand). AppManager can run all other actions on a managed Windows computer or logical server (including the management server or a proxy server). Note that some actions are application-specific. For example, the Action_IISPauseSite Knowledge Script must run on an IIS server.

Actions for managed Windows resources are **Windows actions** and actions for managed UNIX or Linux computers are **UNIX actions**.

The action Knowledge Scripts, such as Action_MAPIMail and Action_Messenger, have built-in severity filtering. Using the **Severity Configuration** parameters, you can specify the range of severities you want the action to execute.

For more information about action Knowledge Scripts, see the Help.

5.6.1 Understanding How Actions Work on Windows Resources

When monitoring a Windows resource, you can configure a job to initiate a different corrective action depending on when AppManager detects the event condition. To do this, configure more than one Windows action and specify a different action schedule. Building upon the example in the previous section, when CPU utilization exceeds the threshold during business hours, an e-mail action runs; during non-business hours, a page action runs.

Use the AppManager Operator Console to configure the scheduled hours of operation for an action schedule. For more information, see the *Operator Console User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

In addition, you can configure an action to run each time AppManager raises an event, only when it creates a duplicate event a specified number of times, or when the event condition no longer exists.

To configure a job to initiate a similar response depending on when AppManager detects the event condition, make a copy of the action Knowledge Script and configure the job to run each action with the corresponding action schedule. For example, if CPU utilization exceeds the threshold during the week, an e-mail action (Action_MapiMail) sends an e-mail notification to the IT operator; if utilization exceeds the threshold on the weekend, an e-mail action (Action_MapiMailOnCallIT) sends an e-mail notification to the on-call IT operator.

5.6.2 Understanding How Actions Work on UNIX and Linux Computers

When monitoring a UNIX or Linux computer, you can configure a Windows action to run on the management server or a UNIX action to run on the managed UNIX or Linux computer. However, you must configure a UNIX action to run each time AppManager raises an event. In addition, you cannot use action schedules to initiate a different corrective action depending on when AppManager detects the event condition.

5.6.3 Configuring Actions for a Job

You can configure actions for a job when creating a new job, by modifying an existing job, or by modifying a Knowledge Script Group member. After you display the Properties window for the job, the process is similar in all cases. The following procedure assumes you are able to display the Properties window.

To configure a responsive or corrective action for a job:

1 Display the Properties window for the Knowledge Script job.

2 Click the **Actions** tab.

3 Click **New** to create a new action and then select an action Knowledge Script from the **Action** list. If you are monitoring a Windows resource, UNIX actions are not available.

NOTE: To see a list of available actions, click the **Actions** category in a **Knowledge Scripts** view.

---

4 In the **Location** column, specify where the action runs:

Select...	To run the action on the...
MS	Management server. Select this option to initiate a Windows action on the management server. If you are monitoring a UNIX or Linux computer, you must select this option to initiate a Windows action.
MC	Managed Windows resource (physical computer or logical server). Select this option to initiate a Windows action on the managed Windows resource. If you are monitoring a UNIX or Linux computer, select this option to initiate a UNIX action on the managed computer.
Proxy	Proxy server. Select this option to initiate a Windows action on a Windows computer with the AppManager agent (netiqmc.exe). This option is not applicable when monitoring UNIX or Linux computers.

5 In the **Type** column, configure an action to run the first time AppManager raises an event (a unique event), after it creates a duplicate child event a specified number of times, or when the event condition no longer exists. Note that you must configure a UNIX action to run the first time AppManager raises an event:

Select...	To run an action...
New event	The first time AppManager raises an event (a unique event). This is the default.
Repeat Event -n	When AppManager creates the specified number of duplicate child events. The default value, 1, permits an action each time AppManager creates a new duplicate child event. Specify the number of times that AppManager must create a new duplicate child event to initiate an action. Note that if you enabled event collapsing, increments to the event count of a child event are not the same as a new duplicate child event. For example, if you enabled event collapsing with the default interval of 20 minutes, when AppManager first detects an event condition, it creates a child event. Subsequent events increment the event count until the 20 minute interval has passed. At this point, AppManager can create a new child event.
Event Down	When the event condition no longer exists. To use this option, check **Generate a new event when original event condition no longer exists** on the **Advanced** tab.

6 In the **Schedule** column, select an action schedule to specify the available hours during which the action can run. When monitoring UNIX and Linux computers, action schedules are not applicable.

For information about viewing and configuring the scheduled hours of operation for an action schedule, use the AppManager Operator Console to view the repository preferences. For more information, see the *Operator Console User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

7 (Conditional) To set the properties for an action Knowledge Script, select a Knowledge Script in the **Action** list and then click **Properties**.

Most actions require you to set some additional properties. For example, if you select an e-mail action, you need to specify an e-mail recipient. For more information about action Knowledge Scripts and their parameters, see the Help.

8 Click **OK** to close the Properties window for the action Knowledge Script.

9 Click **New** to configure another action or click **OK** to start the job.

5.7   Selecting Objects to Monitor

If you run a Knowledge Script on an object that contains other objects (for example, a group that has multiple computers or a disk folder that contains multiple disks on a single computer), the script runs by default on all the objects in the hierarchy that match the Knowledge Script type.

You can select the objects on which you want the job to run.

To select a subset of objects on which to run the Knowledge Script:

1 In the Properties window for the job, click the **Objects** tab.

2 In the list of objects, de-select the objects you do not want the Knowledge Script to monitor. By default, all objects are selected.

3 Click **OK** to start the job.

5.8   Setting Advanced Job Properties

You can specify advanced job properties to filter temporary event conditions. If AppManager raises an event, you can configure the advanced job properties to collapse duplicate events or automatically close the event when the event condition no longer exists.

The following sections describe additional options you can use to configure data collection.

5.8.1   Filtering Temporary Event Conditions

Specify criteria to filter temporary events by configuring the number of times that AppManager must detect a duplicate event condition during a specified number of job iterations before it raises an event. An event condition is considered a **duplicate** when AppManager detects it on the same object name with the same event message, severity, and job ID as a previous event condition.

For example, when the threshold for the number of duplicate event conditions is set to 3 and the number of job iterations is set to 5, if AppManager detects an event condition each time the job runs, it ignores the first two event conditions, raises an event for the third event condition, and starts the time interval for event collapsing.

AppManager ignores the next two event conditions it detects during the interval—only two event conditions are detected during the interval, which falls below the threshold. The event condition count resets when the interval resets, after the fifth job iteration. To raise another event, a duplicate event condition must occur three times during the next five job iterations, and if you enabled event collapsing, AppManager must generate the event within the time interval for event collapsing.

AppManager does not log ignored event conditions in the repository and does not increment the event count of the original child event. It also suppresses actions associated with ignored event conditions.

To specify the number of times AppManager must detect an event condition during a specified number of job iterations:

1 On the Properties window for the Knowledge Script job, click the **Advanced** tab.

2 Enter criteria to specify a threshold for raising an event:

For...	Do this...
Raise event if event condition occurs n times...	Enter a threshold for the number of times AppManager must detect a duplicate event during the specified number of job iterations before raising an event. (The detected event conditions do not need to be consecutive.)
...within M iterations	Enter the number of job iterations to specify the interval. AppManager resets the interval after the given number of job iterations.
	The defaults for the number of duplicate events and job iterations are both 1, which means that AppManager does not ignore duplicate events and raises an event each time it detects an event condition.

3 Click **OK** to start the job.

5.8.2   Changing Event Collapsing Options for One Job

Sometimes a Knowledge Script job detects identical events. To suppress these duplicates and their specified actions, you can set event collapsing options. You can set these options globally, for all jobs, or for individual jobs. This section discusses how to set event collapsing options for each job you run.

NOTE: For more information about duplicate events and global event collapsing options, see .

To change the event collapsing options for a job:

1 On the Properties window for the Knowledge Script job, click the **Advanced** tab.

2 Filter temporary event conditions by specifying the number of event conditions AppManager must detect during a specified number of job iterations. If AppManager detects an event condition the specified number of times during the interval, it raises an event. For example, to configure a job to raise an event if AppManager detects the event condition twice during an interval of five job iterations, set **Raise an event if event condition occurs** to 2 **times** within 5 **job iterations**.

For more information, see .

3 Select **Collapse duplicate events into a single event**.

AppManager logs duplicate events in the repository; however, it suppresses actions associated with collapsed duplicate events, whether the actions are initiated on the managed computer, management server, or proxy computer.

4 Set the **Time interval for event collapsing**. The default time interval is 20 minutes.

AppManager collapses any duplicate events it created within the time interval into the original child event. The time interval can be measured from when AppManager detects the first event condition or when it detected the most recent event condition.

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

NOTE: Event collapsing options set on the **Advanced** tab in the Properties window affect only the job you are about to run. To set global default event collapsing options for all Knowledge Scripts, use the Operator Console to set the QDB preferences. For more information, see the *Operator Console User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

5.8.3 Automatically Closing an Event When the Event Condition No Longer Exists

When you are monitoring a Windows resource that tends to raise events frequently, you can reduce the amount of time you spend managing these events by configuring the job to automatically close an event when the event condition no longer exists.

The first time AppManager detects an event condition, it creates a parent event and a child event; it does not raise additional events as long as the event condition persists, and it does not increment the event count for the child event. When the event condition no longer exists, AppManager creates a new event to indicate the change in the event condition and optionally, closes the original event. You can set this option globally for all jobs or for a particular job. This section discusses how to automatically close events for each job you run.

NOTE: For more information about automatically closing events and global options, see Section 6.10, "How Events Are Automatically Closed When the Event Condition No Longer Exists," on page 103.

To automatically close events for each job you run:

1 In the Properties window for the Windows-based job, click the **Advanced** tab.

2 Select **Generate a new event when original event condition no longer exists**. With this option, AppManager raises an event when the event condition that initiated the original event no longer exists. Note that this option is not available when configuring a job to run on a UNIX or Linux computer.

With this option selected, you can specify the event severity when AppManager raises an event in response to the change in the event condition. By default, the event severity is set to 20.

3 To close the original event when the event condition no longer exists, select **Automatically close original event**.

4 Click **OK** to start the job.

5.8.4 Changing Data Collection Options for One Job

There are several data collection options you can use to optimize how a Knowledge Script job collects data and specify whether data is flagged for upload to the NetIQ Corporation Analysis Center.

You can set data collection options for all jobs or a particular job. This section discusses how to set data collection options for each job you run. For information about configuring global data collection options, see the Operator Console Help.

To change data collection options for a job:

1 In the Properties window for the job, click the **Advanced** tab.

2 Select **Collect data details with data point** to collect details for each data point (such as server name and collection time).

   This option only applies to data collection for AppManager graphs and charts. AppManager displays these details when you want to view the details of a data point in a graph or chart.

3 To control how frequently AppManager collects a data point, update **Collect data every n iterations** to specify the number of times that a job must run to collect data.

   For example, if a Knowledge Script job is set to run at regular intervals of 10 minutes and you set this value to 3 iterations, AppManager creates a data point every third time the job runs, or every 30 minutes. The default is 1 iteration, which means AppManager creates a data point every time the job runs.

4 If you configured the job to create a data point once for a specified number of iterations, select **Calculate average** to create that data point as an average value of all data points measured for all iterations. For example, if you configure the repository to collect a data point every 3rd time a job runs, and the data point for the first iteration is 2, the data point for the second iteration is 5, and the data point for the third iteration is 10, AppManager creates the data point with a value of 5.6666 on the third iteration.

5 To begin collecting data when AppManager raises an event, select **Start collecting data when an event is generated**. For example, if a Knowledge Script job is set to raise an event when CPU usage exceeds 70%, then the job will also start collecting data at that point.

6 When you finish, click **OK**.

5.9 Setting Discovery Options

Discovery Knowledge Scripts allow you to specify the type of discovery you want to perform when a discovery job runs. You can specify a full discovery or a delta discovery. A full discovery performs a complete discovery of all information and sends this information to the QDB each time the discovery job runs. A delta discovery identifies any changes since the last discovery job and only sends changes to the QDB.

5.9.1 Advantages of Delta Discovery

Delta discoveries are more efficient and require fewer system and network resources to perform since they only send information about changes to the QDB. This is the default setting for any new discovery job you run. However, the first time you run a delta discovery job it will perform a full discovery since there is no previous full discovery to compare against. Any subsequent jobs will perform a delta discovery.

It is possible to have a delta discovery job running at the same time as a full discovery job. For example, you might have a delta discovery job that runs daily and a full discovery job that runs weekly. If these two jobs run concurrently, when the agent compares the full discovery to the delta discovery it might not detect any changes since the full discovery includes the same information as the delta discovery. To avoid this error, the agent drops the delta discovery job and only uses the full discovery. This ensures that the job discovers all changes. For more information about delta discovery, see the Help.

5.9.2 Delta Discovery and Clustered Resources

Due to the nature of clustered resources that you monitor with AppManager, you cannot use delta discovery with these resources. This includes discovery of clustered servers and SQL and Exchange installed on clustered servers. Attempting to perform delta discovery on these clustered resources will result in discovery errors, including missing or repeated discovery information. To avoid this problem, configure any discovery Knowledge Scripts you run on clustered resources to perform a full discovery with each discovery. While you might not see the same performance improvements available with delta discovery, this method ensures there are no discovery errors with clustered resources.

5.9.3 Deleting Objects

AppManager never deletes top-level objects in the **Enterprise Layout** view of the **Navigation** pane as the result of a discovery. For example, if you perform a delta discovery for servers in your environment and one or more servers happen to be unavailable when you run the discovery, these servers appear as deleted servers in the delta discovery results. However, the servers are not actually gone, they are simply unavailable. To remove a discovered top-level object from the **Enterprise Layout** view, you must manually delete it. This ensures AppManager does not inadvertently delete objects as the result of a discovery.

Unlike top-level objects, AppManager does delete sub-objects, such as printers, from the E**nterprise Layout** view as the result of delta discovery.

If you manually delete a top-level object from the **Enterprise Layout** view, subsequent delta discoveries will not re-discover the deleted object and restore it to the view. To restore top-level objects that you manually delete, perform a full discovery.

If a previously deleted sub-object is physically restored to your monitored environment, a delta discovery automatically restores it to the **Enterprise Layout** view.

5.10 Customizing Knowledge Scripts

As you gain experience monitoring your environment, you might find that you want to change the default properties for selected Knowledge Scripts rather than changing the parameter values for individual jobs. You can copy a base Knowledge Script to create a derived script with the settings you want. For more information, see "Copying a Knowledge Script" on page 87.

Before changing the default threshold values or schedule for a Knowledge Script, you might want to:

- Run any Knowledge Scripts that collect data related to the threshold you want to change to determine a typical value for your environment.
- Run the Knowledge Script you want to change with the default values to evaluate whether you are seeing too many or too few events.
- Run one or more independent jobs with the values you want to use to test that the results are what you expect before changing the defaults.

5.10.1 Copying a Knowledge Script

You can copy a base Knowledge Script to create a derived Knowledge Script and then change its properties to create a new, customized Knowledge Script. The base script and the derived script share logic, but the derived script has its own settings. For example, a base Knowledge Script might have a default threshold of 10 for a certain parameter while the derived Knowledge Script has a default threshold of 20. The two Knowledge Scripts share the script logic, but the settings are different.

Because a base Knowledge Script and a derived Knowledge Script share logic, when the logic for a base Knowledge Script changes, AppManager can automatically update the logic for derived Knowledge Scripts without updating the settings. This option (**Automatically update Derived KS properties for checked in Base KSs**) is enabled by default for new installations. For more information about enabling or disabling the option, see "Knowledge Scripts Options" on page 42.

If the automatic update option is not enabled, you must manually update the logic for derived Knowledge Scripts when the base script changes. For information about manually updating derived scripts, see "Manually Propagating a base Knowledge Script to Derived Knowledge Scripts and Ad Hoc Jobs" on page 88.

If you create a copy of a derived Knowledge Script, the additional copy is still a derived script of the original base script. For example, if you copy NT_CpuLoaded to My_CpuLoaded and then copy My_CpuLoaded to Your_CpuLoaded, NT_CpuLoaded is the base script for Your_CpuLoaded. Updating NT_CpuLoaded will update both My_CpuLoaded and Your_CpuLoaded.

Derived Knowledge Script names cannot exceed 125 characters. NetIQ Corporation recommends limiting the length of a Knowledge Script name to 120 characters. Limiting the length to 120 characters prevents AppManager from truncating the Knowledge Script job name when running the Knowledge Script from a monitoring policy or as an ad hoc job started by a Knowledge Script Group. For more information, see Section 7.1.2, "How Knowledge Script Groups Work," on page 107.

To copy a base Knowledge Script:

1 In the **Enterprise Layout** view of the **Navigation** pane, click the appropriate **Knowledge Scripts** view.

2 In the view pane, right-click the Knowledge Script that you want to copy and select **Copy Knowledge Script**.

3 Enter a new name and description for the Knowledge Script, and then click **OK**.

By default, AppManager appends the phrase "CopyOf" to the Knowledge Script name to ensure that a new script is created.

To...	Do this...
Display the new Knowledge Script in the same category as the original script	Do not change the prefix that appears before the underscore.
Create a new category for your customized Knowledge Scripts	Type a new name before the underscore. For example, `MySite_`.

The new Knowledge Script is automatically checked in to the primary QDB.

4 In the view pane, double-click the new Knowledge Script you created.

5 In the Properties window, make the changes you want on the **Schedule**, **Values**, **Actions**, or **Advanced** tabs, and then click **OK** to save the changes to the QDB.

Jobs that you create from the derived Knowledge Script will use the properties that you specified.

5.10.2 Manually Propagating a base Knowledge Script to Derived Knowledge Scripts and Ad Hoc Jobs

You can use the Knowledge Script Propagation wizard to view all of the derived Knowledge Scripts and ad hoc jobs that are pending propagation and select specific scripts and ad hoc jobs to process or update all scripts and ad hoc jobs. You can also right-click a base Knowledge Script in a **Knowledge Scripts** view and use the menu options provided. When you use the right-click option to propagate to ad hoc jobs, AppManager updates all of the associated ad hoc jobs in all QDBs. If you want to propagate only to ad hoc jobs running in specific QDBs, use the wizard. When you use the right-click option to propagate to derived Knowledge Scripts, you can choose the scripts to update.

NOTE: If the **Automatically update Derived KS properties for checked in Base KSs** option is enabled, the right-click option to propagate to derived Knowledge Scripts is not available because the propagation occurs automatically.

When you propagate a base script to derived scripts or ad hoc jobs, AppManager merges any additions or deletions in the base script with the target scripts or jobs but retains the existing values, including override values.

To manually propagate a base Knowledge Script to derived Knowledge Scripts and ad hoc jobs using the wizard:

1 On the **Global Tasks** tab, click **Launch KS Propagation Wizard**.

2 (Conditional) To propagate a base Knowledge Script to derived Knowledge Scripts, on the **Propagation to Derived Knowledge Scripts** tab, select the specific base scripts that you want to propagate, or click **Select All** to propagate all pending changes.

3 (Conditional) To propagate a base Knowledge Script to ad hoc jobs, on the **Propagation to Ad Hoc Jobs** tab, expand the script that you want to propagate to view the QDBs where ad hoc jobs are running.

4 Select the QDBs where you want to update ad hoc jobs, or click **Select All** to propagate all pending changes.

5 When you are done making your selections, click **Propagate**, and then confirm that you are ready to propagate.

If you selected to propagate to derived Knowledge Scripts, AppManager updates all direct copies of the script, all Knowledge Script Group members that are based on the script, and all monitoring policy jobs for the script.

If you selected to propagate to ad hoc jobs, AppManager updates the jobs in the QDBs that you selected.

To manually propagate a base Knowledge Script to derived Knowledge Scripts and ad hoc jobs using the right-click option:

1 In the **Enterprise Layout** view of the **Navigation** pane, click the appropriate **Knowledge Scripts** view.

2 In the view pane, right-click the Knowledge Script you want, select **Propagate**, and, depending on where you want to propagate the script, select **To Ad Hoc Jobs** or **To Derived Knowledge Scripts**, and then click **OK**.

(Conditional) If you selected to propagate to derived Knowledge Scripts, select the specific scripts that you want to update or click **Select All** to update all available scripts, and then confirm that you are ready to propagate.

If you selected to propagate to ad hoc jobs, AppManager updates all associated ad hoc jobs in all QDBs.

If you selected to propagate to derived Knowledge Scripts, AppManager updates all direct copies that you selected, all Knowledge Script Group members that you selected, and all monitoring policy jobs for the Knowledge Script Group members that you selected.

5.11 Checking and Changing Job Status

Once AppManager starts a job, the job typically goes through the following states in its life cycle:

This status...	Indicates...
Pending	A temporary state when AppManager first starts a job or when you temporarily stop and restart a job. If a job stays in this state for very long, it might indicate a network communication problem or that an agent computer has been shut down.
	Click the + (plus sign) next to a parent job to see additional status details about child jobs:
	Start indicates the job is in the process of starting.
	Stop indicates the job is in the process of stopping.

This status...	Indicates...
Running	AppManager started the job, but the job might or might not be actively executing its monitoring functions.
	Click the + (plus sign) next to a parent job to see additional status details about child jobs:
	Active indicates the job is currently running on the agent computer, monitoring computer resources according to the schedule and parameters you set. Any corrective actions you defined for the job occur automatically when the event condition is met.
	Scheduled indicates the job has not started executing its monitoring functions. Monitoring takes place according to the schedule you set.
	Inactive indicates the job is currently inactive due to scheduled maintenance. When the maintenance period is over, the job will resume running according to the schedule you set. To prevent jobs from running during scheduled maintenance periods, run the AMADMIN_SchedMaint Knowledge Script. For more information about this Knowledge Script, double-click it and then click **Help** on the **Values** tab in the Properties window.
Stopped	The job has run to completion or has been stopped manually.
Closed	The job is no longer in use but is still available for historical purposes and you can re-open it in the future.
Error	The job encountered a problem (for example, an error in the agent computer's runtime configuration prevented the job from running properly).

5.11.1 Viewing Jobs in the Jobs View

Click the **Jobs** view to get more details about all jobs, including the current status of each job, whether a Knowledge Script Group or a monitoring policy started the job, where the job is running, the name of the Knowledge Script being run, and the time the job was submitted.

The **Jobs** view lists all child jobs. You can group by **Parent Job ID** by dragging the column header to the **Group By** field.

5.11.2 Viewing the Actual Script for a Running Job

To view the actual script that is running on a computer:

1 In the **Enterprise Layout** view of the **Navigation** pane, click the appropriate **Jobs** view.

2 In the **Jobs** view, double-click a child job to view its properties.

3 Click the **View KS** button 🖼.

5.11.3 Changing the Sorting Order

You can click any of the headings in the **Jobs** view to sort all parent entries in that field or toggle the sorting order preference. For example, jobs in the **Jobs** view are normally sorted by job ID in ascending order. To sort the job list by Knowledge Script, click the **Knowledge Script** header. To reverse the sorting order from ascending to descending, click the **Knowledge Script** header again.

5.11.4 Changing Job Status

You can change the status of a particular job or all jobs running on an object in the **Jobs** view.

For a particular job, you can click the job's status icon in the view pane. Clicking a running job stops it; clicking a stopped job, restarts it. You can attempt to restart a job with an Error status, but if the job still encounters an error, the status will return to Error.

To stop or start a particular job:

1 In the **Enterprise Layout** view of the **Navigation** pane, click the appropriate **Jobs** view.

2 (Conditional) To stop a job, in the view pane, click the green status icon for the job you want to stop.

   The status and icon changes to **Pending** (yellow) and then to **Stopped** (red).

3 (Conditional) To start a job, in the view pane, click the red status icon for the job you want to start.

   The status (and icon) changes to **Pending** (yellow) and then to **Running** (green).

5.11.5 Changing Properties of a Running Job

To change the properties of a job that is running:

1 In the **Enterprise Layout** view of the **Navigation** pane, click the appropriate **Jobs** view.

2 In the view pane, select the job you want to modify.

3 In the **Tasks** pane, click **Job Properties**.

4 Change the properties you want in any tab of the Job Properties window (for example, modify the schedule, set a new threshold value, or modify the list of objects that the job monitors).

5 Click **OK**.

   AppManager temporarily changes the job to a pending state and then resumes running it with the new properties.

5.11.6 Viewing Job Comments

Console operators and administrators add job comments. The comments usually include information about the job, for example, information about why the monitoring values of the job were changed.

This information can be added to the properties of an ad hoc job. In the Control Center console, you can group jobs by job comment information. Job comments are not applicable to policy-based jobs or jobs that a Knowledge Script Group starts.

You can view a job comment:

- On the **Values** tab of the Properties window for the job. The **Comments** field contains the latest comment. To view all of the comments for the job, click **History**.

  The Job Comment History window displays the comment information for the job along with information about when the comment was made and who created the comment.

  You can delete job comments from the Job Comment History window by clicking the comment you want and clicking **Delete**.

- In the **Comment** column of the **Jobs** view.

5.11.7 Adding Comments to a Job

Adding comments to a job allows you to record and share information with other console operators and administrators. You cannot add comments to a policy-based job.

When you add comments to a job, AppManager automatically restarts the job and then displays the comment information in the **Comments** field.

To add comments to a job:

1 In the **Enterprise Layout** view of the **Navigation** pane, click the appropriate **Jobs** view.

2 In the view pane, select the job where you want to add comments.

3 In the **Tasks** pane, click **Job Properties**.

4 On the **Values** tab of the Job Properties window, use the **Comments** field to enter the text (up to 256 characters) you want saved with the job. You can also specify a URL to create a hyperlink to a web site.

5 Click **OK** to apply your changes and close the window.

AppManager automatically restarts the job.

5.12 Initiating Maintenance on a Computer

In many environments, you might need to perform unscheduled maintenance on a computer. For example, an organization might have an Apache Web server that must be shut down immediately. In this case, you can temporarily block all jobs, events, and data for a particular computer, including jobs that remotely monitor a computer, by placing the computer in **machine maintenance mode**. For more information about placing computers in maintenance mode, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

5.13 Grouping and Filtering Job Information

By default, AppManager displays all of the jobs in your environment in list form, with each job as a separate entry. Although this provides complete information, as you add more jobs you might want to organize your jobs into groups, or filter the information displayed to make jobs easier to work with. For example, you might want to organize jobs by where they are running or when they were submitted, or limit the jobs displayed to those running on a specific computer or associated with a specific Knowledge Script.

5.13.1 Organizing Jobs into Groups

To organize jobs into groups:

1 In the **Enterprise Layout** view of the **Navigation** pane, click the appropriate **Jobs** view.

2 In the view pane, drag the column you want to group by into the **Group By** field. For example, by **Computer** or **Knowledge Script**.

Once you make your selection, AppManager arranges job information into groups as specified.

3 To expand the detailed job information for any group, click the + (plus sign) to the left of the group.

For more information about customizing the Control Center console, see Section 3.1, "Customizing the Control Center Console Layout," on page 37.

5.13.2 Filtering the List of Jobs

You can select to show or hide jobs in the **Jobs** view by filtering columnar data. For example, you might want to hide running jobs because you are mostly concerned about stopped jobs. Later, if you want to view running jobs, you can select to show them. For more information about filtering, including the use of local and server filters, see "Working with Lists" on page 38.

5.14 Viewing Job Results

Jobs can generate information about:

* The actual job, such as status and Job ID.
* Action, such as status and when the action was submitted.
* Event, such as status and severity.
* Data, for charts and reports.

5.14.1 View Jobs in the Jobs View

To quickly see where jobs are running in your environment, view jobs in the **Jobs** view.

5.14.2 View Results in the Server Information Pane

Use the tabs in the **Server Information** pane to view results.

The **Server Information** pane is only available from a **Servers** view that is configured to display at least one of the following the tabs:

Click this tab...	For information about the...
Events	Events a Knowledge Script job generated. Double-click an event identifier to get more information about the event, to add comments, or view any detailed messages the Knowledge Script job provided.
Jobs	Status of jobs. Double-click a job identifier to get more information about the job and to add comments.
Custom Properties	Custom properties information about the servers in your Control Center environment. For more information, see Section 4.5, "About Custom Properties," on page 61.

5.15 Closing a Job

To close a job:

1 In the **Enterprise Layout** view of the **Navigation** pane, click the appropriate **Jobs** view.

2 In the view pane, select the job you want to close. To select multiple jobs, press **Ctrl** or **Shift** while making your selections.

NOTE

- ◆ If the job is hidden, check to see if a filter is applied to a column in the job list. If the **Filter** button is visible, a filter is applied. Click the button and click **(All)** to remove the filter and show the hidden job.

- ◆ You cannot close a running job. You must first stop the job before you can close it.

3 In the **Tasks** pane, click **Close Job**.

5.16 Deleting a Job

You can delete pending, stopped, or closed jobs. When you delete a job, you delete the job and its events and graph data from the QDB.

NOTE: You can configure each QDB to maintain events and graph data. However, do so from the AppManager Operator Console. At this time, you cannot configure QDB preferences from the Control Center console. For more information, see the *Operator Console User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

To delete a job:

1 In the **Enterprise Layout** view of the **Navigation** pane, click the appropriate **Jobs** view.

2 In the view pane, select the job you want to delete.

If the job is hidden, check to see if a filter is applied to a column in the job list. If the **Filter** button is blue, a filter is applied. Click the button and click **ALL** to remove the filter and show the hidden job.

To select multiple jobs, press the **Ctrl** or **Shift** keys as you make your selections.

3 In the **Tasks** pane, click **Delete Job**.

6 Responding to Events

This chapter provides an overview of how the software inventory and Knowledge Script jobs trigger events and how to view and respond to events in the Control Center console.

6.1 How Knowledge Script Jobs Raise Events

When you create a Knowledge Script job, the parameters you set on the **Values** tab in the Properties window indicate what you want the job to do. If you want the job to send an alert or notify you in some way when a particular condition exists, set the **Event** parameter to **y** (yes). As a Knowledge Script job runs, it checks whether the condition you defined exists – for example, whether a threshold value has been exceeded or a particular process is down.

If a Knowledge Script detects a problem and the **Event** parameter is set to **y**, the job triggers an event, which is:

- Logged in the QDB.
- In most cases, displayed in the form of an event alert in an **Events** view of the Control Center console.

In this state, AppManager considers the event **Open** and requiring your attention.

6.2 How Software Inventory Raises Events

Software inventory raises events under the following conditions:

- If a new agent computer reports inventory to a QDB in which another agent computer has the same AuthoritativeID, software inventory generates an event during the process of reporting the new inventory to the QDB. AppManager associates the event with the new agent computer.
- If a new agent computer reports inventory to a QDB and then AppManager synchronizes the information to a CCDB where another agent computer with the same AuthoritativeID has already been synchronized from another QDB, software inventory generates an event during the synchronization process. AppManager associates the event with the first management server in the QDB that the duplicate agent computer has reported to. In the case of a version 9.1 QDB, AppManager associates the event with the first management server in the primary QDB.

Events that software inventory raises do not have a job ID because they are system events. In the event details, AppManager displays `Software_Inventory` in the `Knowledge Script` field. The easiest way to view software inventory events in an **Events** view is to display the events in a flat view rather than a hierarchical view. In a hierarchical view, the parent event reflects the highest severity child event. Because system events can have different Knowledge Script identifiers (for example, one from AMHealth and one from Software_Inventory), it is easier to identify the source of individual events in a flat view.

6.3   Event Severity Levels

Because some jobs check for serious problems that can severely impact your business while others simply provide informational feedback, events range in severity. The event indicators in the **Events** view reflect at a glance the relative severity of the problem.

AppManager classifies events as follows:

Severity level	Indicator	What it means (typically)
Severe Levels 1 to 10 (default)	Red	A serious problem requires your attention (for example, a computer or service is down).
Warning Levels 11 to 20 (default)	Yellow	Unexpected behavior or minor problems might require your attention, but they are not affecting network up time or preventing users from doing their work (for example, AppManager could not find a counter you want to monitor or an application you want to discover on the selected computer).
Informational Levels 21 to 30 (default)	Blue	AppManager successfully completed a task (for example, restarted a service or completed a discovery).
Diagnostic Levels 31 to 40 (default)	Magenta	AppManager returned diagnostic information (for example, when debugging problems).

You can change the severity levels to better reflect your system management policies.

You configure event severity levels as a QDB preference. At this time, you cannot configure them from the Control Center console. Use the AppManager Operator Console to configure QDB preferences. For more information, see the *Operator Console User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

6.3.1   Calculating Event Severity in the Servers View

Control Center displays the most severe event in the view pane for the **Servers** view under a management group.

You can configure the same server to be associated with two management groups that are associated with two views. For example, you can add a server group to a management group associated with the **SQL** view and to another management group associated with the **NT** view.

The severity of events in both the views could be different for the same server group. If you select the **Servers** view under the **Master** management group, the view pane displays the most severe event for the server across all views. However, if the server is a member of any of the other management groups, and you select the server under those management groups, the view pane displays the most severe event only for the selected view. For example, if there are no events for the management group associated with the **SQL** view, but there are severe events associated with the **NT** view for the same server, then the **NT** view displays the most severe event and the **SQL** view displays **[No Events]**.

6.4 Viewing Events

To see a list of events generated by Knowledge Script jobs, click an **Events** view in a management group. If you are currently in the **Master** management group, the **Events** view lists all events.

You can also view events in a **Servers** or **Service Map** view, if you configured the view to display the **Events** tab in the **Server Information** pane. When viewing events in a **Servers** or **Service Map** view, the **Server Information** pane displays all event information for the selected object while the selected **Servers** or **Service Map** view object reflects the event status for the most severe open event.

6.5 Grouping and Filtering Event Information

By default, Control Center displays all of the events in your environment in list form, with each event as a separate entry. Although this provides complete information, as you add more events you might want to organize your events into groups, or filter the information displayed to make events easier to work with. For example, you might want to organize events by status or severity level.

6.5.1 Organizing Events into Groups

To group events in the view pane:

1. In the **Enterprise Layout** view of the **Navigation** pane, select the **Events** view you want to organize.

2. In the **Events** view, drag the column you want to group by into the **Group By** field, for example, by **Status** or **Severity**.

   Once you make your selection, Control Center arranges event information into groups as specified.

3. To expand the detailed event information for any group, click the **>** (greater-than sign) to the left of the group.

6.5.2 Filtering the List of Events

Filtering events removes events from the list but does not delete the event information. For example, you might want show or hide events in the **Events** view based on their status so they do not clutter the list of events you are currently interested in. You can apply server filters, local filters, and column filters to an **Events** view. For more information about filtering, see "Working with Lists" on page 38.

6.5.3 Viewing Events

The **Events** view lists event information. The first event a particular job generates creates a parent and child event entry. By default, Control Center groups child events by parent event in the **Events** view and displays event messages in a separate row. You can change the appearance of an **Events** view depending upon your requirements. For example, you can display all events in a flat view rather than grouped by parent event.

To change the appearance of events:

1. In the **Enterprise Layout** view of the **Navigation** pane, select the **Events** view you want to modify.

2. In the **Tasks** pane, click **View Properties** > **General**.

3 Modify the settings on the **General** tab and any other tabs as needed.

4 After completing your changes, click **OK**.

For more information about customizing the Control Center console, see Section 3.1, "Customizing the Control Center Console Layout," on page 37 or see the Help.

6.5.4 Changing the Sorting Order

You can click any of the headings in the **Events** view to sort all event entries in that field or toggle the sorting order preference. For example, Control Center normally sorts events in the **Events** view by last occurrence and in ascending order. To sort the event list by severity, click the **Severity** header. To reverse the sorting order from ascending to descending, click the **Severity** header again.

6.6 Viewing Details about an Event

The Event Properties window includes more information (such as when the event first occurred and the user running the job) than the summary information in the view pane. You can also change the status of events in the Event Properties window.

To view the details of an event:

1 In the **Enterprise Layout** view of the **Navigation** pane, select the appropriate **Events** view.

2 In the view pane, select the parent or child event you want to investigate.

3 In the **Tasks** pane, click **Event Properties**.

4 To browse through event details for other events, click **Previous** or **Next**. If you selected a parent event, this selects the previous or next parent. If you selected a child event, this selects the previous or next child event for the same parent event.

Depending on the Knowledge Script and the event, Control Center might include additional tabs to display more information.

For information about...	See...
Viewing event details (**Details** tab)	Section 6.6.1, "Details Tab," on page 98.
Viewing event actions (**Action** tab)	Section 6.6.2, "Actions Tab," on page 99.
Viewing event comments (**Comments** tab)	Section 6.6.3, "Comments Tab," on page 99.
Adding comments to an event (**Comments** tab)	Section 6.7, "Adding Comments to an Event," on page 100.

5 (Optional) To acknowledge and close events, select the corresponding option. Closing an event automatically acknowledges it. After you close an event, you can delete it.

6 After you finish viewing event properties and acknowledging or closing events, click **OK**.

6.6.1 Details Tab

AppManager records detailed information about what Knowledge Scripts are monitoring in your environment and the conditions that triggered each event. For example, event details for the NT_MemByProcess Knowledge script include the process name and ID of each process as well as

the amount of memory utilized. If you enabled delta discovery, discovery Knowledge Scripts record any discovered objects that have changed, the type of change (Add, Delete, Update), and any old and new values.

To view event details:

1 In the **Enterprise Layout** view of the **Navigation** pane, select the appropriate **Events** view.

2 In the view pane, select the parent or child event you want to investigate.

3 In the **Tasks** pane, click **Event Properties**.

4 Select the **Details** tab to view any detailed information for the event.

5 To browse through event details for other events, click **Previous** or **Next**. If you selected a parent event, this selects the previous or next parent. If you selected a child event, this selects the previous or next child event for the same parent event.

6.6.2 Actions Tab

When AppManager detects events, Knowledge Scripts can initiate a variety of actions, such as sending a page or e-mail, running an external program, or sending an SNMP trap.

If you defined an action for a Knowledge Script job and AppManager performed the action, the Event Properties window includes an **Action** tab, which tells you the results of the action, the script that was run, and the time AppManager performed the action.

To view information about event actions:

1 In the **Enterprise Layout** view of the **Navigation** pane, select the appropriate **Events** view.

2 In the view pane, select the child event you want to investigate (the **Action** tab is only available for child events).

3 In the **Tasks** pane, click **Event Properties**.

4 Select the **Action** tab to view any action status information for the event.

5 To browse through event details for other events, click **Previous** or **Next**. If you selected a parent event, this selects the previous or next parent. If you selected a child event, this selects the previous or next child event for the same parent event.

6.6.3 Comments Tab

Console operators and administrators add event comments, which usually include information about what caused the event or how it was resolved. An event comment can include a hypertext link, and you can click the link to jump to the corresponding URL.

NOTE: If you configured an AppManager report Knowledge Script to raise an event upon successful completion, the event comment can include a hyperlink to the actual report. For information about how to configure report script event comments to include a hyperlink to the actual report, see Chapter 10, "Running Reports," on page 151.

To view a comment:

1 In the **Enterprise Layout** view of the **Navigation** pane, select the appropriate **Events** view.

2 In the view pane, select the parent or child event you want to investigate.

3 In the **Tasks** pane, click **Event Properties**.

4 Click the **Comments** tab.

5 To browse through event details for other events, click **Previous** or **Next**. If you selected a parent event, this selects the previous or next parent. If you selected a child event, this selects the previous or next child event for the same parent event.

6.7 Adding Comments to an Event

Adding comments to an event allows you to record and share information with other console operators and administrators about what caused the event or how you resolved the problem.

You can add comments to parent and child events, individually.

To add comments to an event:

1 In the **Enterprise Layout** view of the **Navigation** pane, select the appropriate **Events** view.

2 In the view pane, select the parent or child event you want to investigate.

3 In the **Tasks** pane, click **Event Properties**.

4 Click the **Comments** tab.

5 Enter the text (up to 255 characters) you want to save with the event. You can also specify a URL to create a hyperlink to a web site.

6 Click **Apply** or click **OK** to apply your changes and close the window.

When you add a comment to an event, the comment indicator on the **Events** tab in the **List** pane changes. For more information, see Section 6.6.3, "Comments Tab," on page 99.

7 To browse through event details for other events, click **Previous** or **Next**. If you selected a parent event, this selects the previous or next parent. If you selected a child event, this selects the previous or next child event for the same parent event.

6.8 Changing the Status of an Event

Once a Knowledge Script detects a problem and generates an event, Control Center adds it to the **Events** view as an open event.

To respond to an open event and change its status, either acknowledge or close the event. How you use the Acknowledged or Closed status depends on your system management policies. In general, you can acknowledge an event to indicate you are aware of the problem or that the problem is being addressed; then close it when the problem has been resolved.

6.8.1 Acknowledging and Closing Events

Once a Knowledge Script generates an event, AppManager lists the event as Open in the **Events** view until you respond to it. Acknowledging the event changes the status of the event in the view pane to Acknowledged (so other administrators know the event occurred but is being addressed). You can also close events whether or not you acknowledge them first.

You can acknowledge or close events by:

◆ Individually acknowledging or closing child events in the view pane (you can also acknowledge or close all child events at once by acknowledging or closing a parent event).

- Acknowledging or closing all events associated with an application server, a group of servers, or all servers in a view.
- Individually acknowledging or closing an event after viewing event details in the Event Properties window.

To acknowledge or close an event from the view pane:

1 In the **Enterprise Layout** view of the **Navigation** pane, select the appropriate **Events** view.

2 In the view pane, select the parent or child event you want to investigate.

3 In the **Tasks** pane, click **Acknowledge Event** or **Close Event**.

If you select a...	Then...
Parent event	You acknowledge the parent and all child events.
Child event	You only acknowledge the child event.

You can also acknowledge or close any events associated with a particular server.

To acknowledge or close all events associated with an application server:

1 In the **Enterprise Layout** view of the **Navigation** pane, select the appropriate **Servers** view.

2 In the view pane, select the server you want to investigate.

3 In the **Tasks** pane, click **Event Tasks > Acknowledge All Events** or **Event Tasks > Close All Events**.

   AppManager acknowledges or closes all parent and child events on that server.

You can also acknowledge or close events from the Event Properties window. For more information about event properties and acknowledging or closing events, see Section 6.6, "Viewing Details about an Event," on page 98.

NOTE: To configure a job to automatically close an event when the event condition no longer exists, see Section 5.8.3, "Automatically Closing an Event When the Event Condition No Longer Exists," on page 84.

6.8.2 Deleting an Event

To delete closed events:

1 In the **Enterprise Layout** view of the **Navigation** pane, select the appropriate **Events** view.

2 In the **Events** view, select the event you want to delete.

   To select multiple events, press the **Ctrl** or **Shift** keys as you make your selections.

3 In the **Tasks** pane, click **Delete Event**.

NOTE: When you select multiple events, you can delete, acknowledge, or close all of the events at once. If you attempt to delete multiple events and your selection includes open or acknowledged events, AppManager does not delete those events. To delete an open or acknowledged event, first close the event.

6.9 What Are Duplicate Events?

The first event a job generates creates both a parent entry and a child event entry. AppManager lists subsequent events from the same job as additional child event entries under the parent event, and an event count tracks the number of child events. (The count for the parent entry is the sum of the count for each child entry.)

A Knowledge Script job might detect **unique** events (for example, events with a different severity or message) or identical events. AppManager considers an event identical, or a **duplicate**, when a new open child event with the same object name, event message, severity, and job ID as a previous event occurs. As with unique events, AppManager logs duplicate events in the repository and, if event collapsing is not enabled, lists them as additional child event entries under the parent event. If you specified actions, AppManager initiates them for each duplicate event.

Although duplicate events are typically valid, it usually is not useful to receive multiple alerts caused by the same problem or condition. In addition, you probably do not want these identical events to initiate duplicate actions, such as repeated e-mail notification or messenger dialog boxes.

To filter temporary event conditions (not just collapse them), you can specify the number of duplicate event conditions that AppManager must detect during a specified number of job iterations before it raises an event. This option is useful in reducing the number of trivial events reported to the management server. AppManager does not log filtered event conditions in the repository and does not increment the event count of the original child event. AppManager also suppresses actions associated with ignored events.

After AppManager raises an event, it can collapse subsequent duplicate events into a single event. Instead of creating new child event entries, AppManager collapses duplicate events into the original child event and increases the child event count. AppManager logs the duplicate events in the repository; however, it suppresses actions associated with collapsed duplicate events, whether it initiates the actions on the managed computer or management server.

AppManager collapses duplicate events within a specified time interval (the default is 20 minutes). You can configure the **event collapsing time interval** to begin:

- When AppManager raises the first event. AppManager collapses all duplicate events within the time interval (static period of time) into one event.

- Each time AppManager generates an event (it is not a static period of time). For example, using the default time of 20 minutes, if a job generates duplicate events every five minutes, then the 20-minute interval is restarted every five minutes, which means it never effectively expires – unless you set an option to have events **ignored**.

After AppManager closes the original child event, or after the event collapsing time interval expires, it creates a new child event and, if you specified an action on the management server, initiates the action when it detects the event condition.

You can configure event collapsing options globally (so that they apply to all Windows-based Knowledge Script jobs) and for individual Windows-based jobs. For more information about where to set the options for all jobs or an individual job, see the *Operator Console User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

The following example explains how event collapsing works when a job detects a problem and raises an event and continues to detect the same condition every 5 minutes for a total of 12 identical events.

Without event collapsing enabled, AppManager lists a parent entry and 12 child entries. The count for each child entry is 1; the count for the parent entry is 12. AppManager logs all 12 events in the repository and initiates 12 instances of the specified action.

With event collapsing enabled but without ignoring events (using the default values of 20 minutes for **Time interval for event collapsing** and the event condition must occur 1 time during 1 iteration), the first event creates a parent entry and one child entry, and begins the 20-minute time interval. Because the duplicate events take place at five-minute intervals, the 20-minute event collapsing interval never expires (it restarts after each duplicate event). And, because the number of events and the number of job iterations is set to 1, AppManager does not ignore duplicate events; instead, it collapses each into the initial child entry.

AppManager lists a parent entry and one child entry; the count for each is 12. AppManager logs all 12 events in the repository but only initiates one instance of the specified action.

With event collapsing enabled and with events ignored (using 20 minutes for **Time interval for event collapsing** and the event condition must occur 5 times within 5 iterations), AppManager ignores the first four events. The fifth event generates a parent entry and one child entry, and begins the 20-minute time interval. The next four events (six through nine) take place during the 20-minute interval and AppManager ignores them. The next event (ten) generates a new child event and restarts the time interval and the job iteration interval. AppManager ignores the eleventh and twelfth events.

AppManager lists a parent entry and two child entries. The count for each child entry is 1; the count for the parent entry is 2. AppManager only logs the two listed child events in the repository and initiates their actions. It ignores the other events and actions.

NOTE

 ◆ You can configure a job to either collapse duplicate events or automatically close events when the event condition no longer exists; you cannot do both. For more information, see Section 6.10, "How Events Are Automatically Closed When the Event Condition No Longer Exists," on page 103.

 ◆ For information about changing event collapsing options for individual Windows-based jobs, see Section 5.8.2, "Changing Event Collapsing Options for One Job," on page 83.

6.9.1 Setting Global Event Collapsing Options

At this time, use the Operator Console to configure QDB preferences to set global, default event collapsing options for all Knowledge Script jobs. For more information, see the Operator Console Help.

6.10 How Events Are Automatically Closed When the Event Condition No Longer Exists

For some event conditions, it is more useful to raise an event when AppManager first detects the condition and then raise a second event when the event condition stops. You can configure an AppManager job to do this; you can also choose to have AppManager close the first event automatically.

With the job you configured to raise an event when an event condition is detected and automatically close the original event when the condition no longer exists, AppManager creates a parent event and a child event with a status of open; if it detects subsequent duplicate event conditions, it suppresses additional child events and does not increase the event count. When the event condition no longer exists, AppManager creates a new event to indicate the change in the event condition and changes the status of the original event to Closed.

For example, the first time a job runs, an event condition does not exist; AppManager does not raise an event. The next time the job runs, AppManager detects an event condition and raises a parent event and a child event.

The job iterates several times, and each time the event condition exists—AppManager does not create additional child events and does not increase the child event count; there is only one child event in the **Events** tab of the **List** pane.

On the next iteration the job runs and AppManager does not detect the event condition. AppManager changes the status of the child event to Closed and creates a new informational event that indicates the change in the event condition status.

If you double-click the event that indicates the change in status of the event condition (for example, event 582), in the Event Properties window, the **Message** tab indicates that the original event (event 581) has been closed.

If the Knowledge Script job detects the event condition again, AppManager creates a new child event.

NOTE

- To automatically close an event when the original event condition no longer exists, configure the job to raise an event with a specified event severity level each time AppManager does not detect the event condition. This event is for informational purposes only and its event severity should be set accordingly. To prevent the parent event from having an Open status when the original event is closed, you can use the Operator Console to configure the QDB preferences to automatically close all open events based on severity level. For more information, see the *Operator Console User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

- You can configure a job to either automatically close events when the event condition no longer exists or collapse duplicate events; you cannot do both. For information about event collapsing, see Section 6.9, "What Are Duplicate Events?," on page 102.

- To configure an individual job to automatically close an event when the event condition no longer exists, you can do so from the Advanced tab in the Knowledge Script Properties dialog box. For more information, see Section 5.8.3, "Automatically Closing an Event When the Event Condition No Longer Exists," on page 84.

- Use the Operator Console set global default options to automatically close events when the event condition no longer exists. For more information, see the *Operator Console User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

6.10.1 Setting Global Options for Automatically Closing Events When the Event Condition No Longer Exists

At this time, use the AppManager Operator Console to set global options for all jobs. For more information, see the *Operator Console User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

7 Monitoring by Policy

This chapter provides an overview of how to use the Control Center console to initiate policy-based monitoring.

7.1 About Policy-based Monitoring

A **monitoring policy** uses a set of pre-configured Knowledge Scripts to automatically monitor resources as they appear in a management group. A monitoring policy enables you to efficiently and consistently monitor all of the resources in your environment.

For example, you can create a monitoring policy to monitor physical disk resources for a particular group of computers. The monitoring policy monitors the discovered physical disk resources on each computer in the group using the same monitoring values. If the disk configuration for a managed computer changes (for example, adding a G: drive), the next time AppManager discovers physical disk resources, the monitoring policy automatically monitors the G: drive. If you add a computer to the group, the monitoring policy automatically monitors the discovered physical disk configuration. If you remove a physical disk from the managed computer and rediscover resources, the monitoring policy automatically stops monitoring the resource.

A monitoring policy is implemented through one or more Knowledge Script Groups. A **Knowledge Script Group** is a set of pre-configured Knowledge Scripts which you can run as part of a monitoring policy for a group, list of ad hoc computers, or view, or as a set of **monitoring jobs** on a physical computer or logical server.

The advantage of using a Knowledge Script Group to monitor by policy is that AppManager automatically monitors matching resources—a monitoring job that an individual Knowledge Script starts only monitors particular resources.

A monitoring policy also enforces the same monitoring values on all monitored resources. However, Control Center allows you to change the monitoring values for a particular policy-based job. This is useful when a resource experiences a temporary or acceptable event condition that is outside the threshold levels that the monitoring policy permits.

If you have a resource that is consistently outside of the policy you created, you can override the monitoring values on that resource. For more information, see "Setting Override Values" on page 74.

Alternatively, you can stop and restart a policy-based job. For more information, see "Stopping and Restarting Policy-based Jobs" on page 115.

You can monitor resources from the **Master** and **AM Logical Servers** management groups, but NetIQ Corporation recommends that you do not implement a monitoring policy on these management groups.

To remove a resource from a policy, either change the policy, reconfigure the management group to not include the resource, or delete the resource from the database.

7.1.1 How Monitoring Policies Work

To better understand how a monitoring policy works, keep in mind:

- A monitoring policy is implemented at the management group level through one or more Knowledge Script Groups. A **Knowledge Script Group** is a pre-configured set of Knowledge Scripts collected together into Knowledge Script Group members.

  A **Knowledge Script Group member** is a derived Knowledge Script, which means that it is a copy of a base Knowledge Script and shares logic with the base script but has its own settings. For example, the NT_CpuLoaded Knowledge Script can be a member of several Knowledge Script Groups, with the member in each Knowledge Script Group configured to use different monitoring values. Because each member is derived from the base script NT_CpuLoaded, if changes occur to the base script and the **Automatically update Derived KS properties for checked in Base KSs** option is enabled, each member is automatically updated, along with the monitoring policy jobs that were created from that member. Otherwise, you must manually propagate updated base scripts to the derived Knowledge Script Group members. When you propagate the base script to the group members, AppManager automatically updates the monitoring policy jobs that were created from those members.

  For more information about the automatic update option and the differences between base Knowledge Scripts and derived Knowledge Scripts, see "About Knowledge Scripts and Jobs" on page 67.

  For information about manually propagating base Knowledge Scripts to Knowledge Script Group members, see "Manually Propagating Base Knowledge Scripts to Knowledge Script Group Members" on page 111.

- A monitoring policy automatically creates policy-based jobs to monitor matching resources in the management group. A **policy-based job** is an instance of a Knowledge Script Group member in the primary QDB that has been replicated to the QDB running the job on agent computers.

  When you apply a monitoring policy to a management group that includes logical servers, the policy-based job runs on the agent computers for the logical servers but only monitors the logical servers. For example, if a physical computer contains virtual machines A and B and you create a management group that only includes virtual machine A and apply a monitoring policy to that management group, the policy-based job runs on the physical computer but only monitors virtual machine A.

  Unlike an ad hoc job, you **cannot** modify the monitoring values of a policy-based job by modifying the policy-based job itself. To change the monitoring values of a policy-based job, you must update the parameter values of the Knowledge Script Group member.

- If a policy-based job encounters an error while attempting to monitor a resource, the monitoring policy attempts to restart the job. AppManager can only restart monitoring jobs; it does not automatically restart ad hoc jobs. If AppManager cannot automatically restart a policy-based job, or if you scheduled the policy-based job to run once, you can manually restart the job after its status changes to Stopped.

- You can configure exceptions to monitoring values in a policy-based job. This is useful when a resource that a policy manages raises unnecessary events. For more information, see Section 5.5, "Setting Override Values," on page 74.

- Unlike ad hoc jobs, you cannot add comments to a policy-based job. For more information about job comments, see Section 5.11.6, "Viewing Job Comments," on page 91.

- If you are configuring a monitoring policy for a group of resources that are managed by more than one QDB, unlike ad hoc jobs, you cannot configure different parameter values for each QDB. To resolve this issue, you can configure override values or configure different management groups for each QDB.

- NetIQ Corporation recommends that you implement a monitoring policy in the Control Center console. If you implement a monitoring policy in the Operator Console, the policy definition is not visible from the Control Center console and you cannot manage it from Control Center. For example, if you use the Operator Console to configure a monitoring policy for the NT view, a management group with the NT view as a member does not display the monitoring policy.

NOTE: A job that is not part of a monitoring policy is an ad hoc job. An ad hoc job monitors a particular resource on a physical computer or logical server. Unlike policy-based jobs, you must manually propagate Knowledge Script properties to ad hoc jobs. In addition, ad hoc jobs do not automatically monitor resources that AppManager discovers after the job starts.

7.1.2 How Knowledge Script Groups Work

If a Knowledge Script Group starts a job, AppManager prepends the Knowledge Script job name with the identifier of the Knowledge Script Group that started the job. This information does not appear in the **Knowledge Scripts** view of the Control Center console.

TIP: To avoid truncating the Knowledge Script job name when running a custom Knowledge Script as part of a monitoring policy or as an ad hoc job started by a Knowledge Script Group, NetIQ Corporation recommends that you limit the length of the Knowledge Script name to 120 characters. The maximum length for a Knowledge Script is 125 characters.

7.1.3 Reporting Considerations

Do **not** run reports as part of a monitoring policy. You should not run the same report on more than one report agent as part of a monitoring policy.

7.1.4 Charting Considerations

To select the data streams you want, the Add Data wizard of the Chart Console lists the corresponding Knowledge Script job names. For policy-based jobs and ad hoc jobs a Knowledge Script Group starts, AppManager prepends the Knowledge Script Group identifier to the Knowledge Script job name.

7.1.5 Viewing Policy-based Jobs in the Jobs View

Policy-based jobs appear in the **Jobs** view. The **Job Type** column displays **Monitoring Policy** for monitoring policy jobs and **Ad Hoc (Knowledge Script)** for ad hoc monitoring jobs.

The **Jobs** view does not display the Knowledge Script Group identifier for policy-based jobs or ad hoc jobs a Knowledge Script Group starts. For more information, see Section 7.1.2, "How Knowledge Script Groups Work," on page 107.

7.2 Creating a Knowledge Script Group

Create a Knowledge Script Group to configure a monitoring policy or to create a group of ad hoc jobs on a particular resource. By default, a Knowledge Script Group appears in the **KSG** category of the **Knowledge Scripts** view.

In the **Master** management group, use the **Knowledge Scripts** view to browse the list of all Knowledge Scripts and to create a Knowledge Script Group. Depending on the discovered resources, the available Knowledge Script categories will vary. For example, if the management group does not include discovered SQL Server resources, the SQL Knowledge Scripts are not displayed.

To create a Knowledge Script Group:

1 In the **Enterprise Layout** view of the **Navigation** pane, select the **Knowledge Scripts** view of the **Master** management group.

2 In the **Tasks** pane, click **Create New Knowledge Script Group**.

3 In the Create New Knowledge Script Group window, type a Knowledge Script Group name and description.

NOTE: By default, AppManager prefixes the Knowledge Script Group name with `KSG_`, which means the Knowledge Script Group appears in the **KSG** category in the **Knowledge Scripts** view. To create the Knowledge Script Group in another category, prefix the name of the Knowledge Script Group using *Category_*. For example, to create a Knowledge Script Group named NT Basics in the NT category, name the Knowledge Script Group `NT_NT Basics`.

---

4 Select an item from the **Knowledge Script Type** list:

Click...	To select Knowledge Scripts that...
Regular	Monitor discovered resources.
Discovery	Discover resources.
Install	Install or update AppManager agent software on a remote computer. For more information, see Chapter 9, "Deploying AppManager to Agent Computers," on page 131.

5 In the **Available Knowledge Scripts** column, select one or more Knowledge Scripts and then click **Add**. The Knowledge Scripts appear in the **Knowledge Script Group Members** column.

6 To configure the properties for a member of the Knowledge Script Group, click a Knowledge Script in the **Knowledge Script Group Members** list and then click **Properties**.

7 In the Properties window, enter properties for the Knowledge Script Group member, and then click **OK**.

In many cases, you can simply use the default schedule and values. To make changes:

Click this tab...	To...
Schedule	Change the default schedule for running the job. For more information, see Section 5.3, "Setting the Schedule," on page 70.
Values	Adjust the threshold and other parameters for the job. For more information, see Section 5.4, "Setting the Values to Monitor," on page 72 and Section 5.5, "Setting Override Values," on page 74.

Click this tab...	To...
Actions	Specify an action you want AppManager to take. For more information, see Section 5.6, "Specifying One or More Corrective Actions," on page 79.
Advanced	Specify whether to filter temporary event conditions and whether to collapse duplicate events for this job into a single event or automatically acknowledge them. Additionally, you can specify data collection options for this job. For more information, see Section 5.8, "Setting Advanced Job Properties," on page 82.

Any changes you make only affect monitoring policies that use this Knowledge Script Group. The changes do not affect the default properties of the Knowledge Script.

8 When you finish configuring the Knowledge Script Group, click **OK**.

7.2.1 Creating a Group of Ad Hoc Jobs

Use a Knowledge Script Group to create a group of ad hoc jobs on a physical computer or logical server. You can group the resulting jobs together but you must configure them individually.

Note that if you are managing more than one QDB, you might see duplicate entries in the **Parent Job ID** column. These entries are not actually duplicates; the Enterprise Job ID ensures each job has a unique ID.

An Enterprise Job ID is a unique identifier for a job in Control Center in the same way that a Job ID is a unique identifier for a job in a QDB.

To create a group of ad hoc jobs, run a Knowledge Script Group in the **Jobs** view or the **Servers** view. Note that you can run Discovery Knowledge Scripts in any management group, including a management group that is not based on the **Master** view.

To create a group of ad hoc jobs:

1 In the **Enterprise Layout** view of the **Navigation** pane, select the **Knowledge Scripts** view you want.

2 In the view pane, browse the list to find the Knowledge Script Group you want.

3 In the **Tasks** pane, click **Create New job**.

4 In the Select Servers window, select the physical computers you want to monitor and click **OK**.

5 To view or change the objects that a Knowledge Script monitors, click a member Knowledge Script in the list and click **Objects**.

6 Expand the list of available objects to select (to monitor) or deselect (not monitor) an object, and then click **OK**.

7 When you are done configuring the group, click **OK** to start ad hoc jobs.

NOTE: For information about configuring the QDB preferences to set the number of times to restart an ad hoc job with an Error status, see the Operator Console Help.

7.3 Changing a Knowledge Script Group

You can update a Knowledge Script Group to add or remove Knowledge Script Group members and to change the job properties of an existing group member.

When you update the properties of a Knowledge Script Group member, AppManager automatically propagates any changes you make to corresponding policy-based jobs. You must manually propagate changes to jobs that are not part of a monitoring policy. For information about manually propagating changes, see "Manually Propagating Base Knowledge Scripts to Knowledge Script Group Members" on page 111.

If you add a member to a Knowledge Script Group, it might take up to five minutes to propagate your changes to corresponding policy-based jobs.

To change a Knowledge Script Group:

1 In the **Enterprise Layout** view of the **Navigation** pane, select the **Knowledge Scripts** view you want.

2 In the view pane, browse the list to find the Knowledge Script Group you want.

3 In the **Tasks** pane, click **Knowledge Script Group Properties.**

4 In the Knowledge Script Group Properties window, you can add or remove Knowledge Script Group members:

To...	Do this...
Remove a Knowledge Script Group member	Click a Knowledge Script in the **Knowledge Script Group Members** list and click **Remove**. Or, click **Remove All** to remove all members. When you close the window, AppManager stops and deletes any associated policy-based jobs based on that member.
Add a Knowledge Script Group member	Click a Knowledge Script in the **Available Knowledge Scripts** list and click **Add**. When you close the window, any groups or views that use the Knowledge Script Group to monitor by policy automatically monitor corresponding resources with the new Knowledge Script.

5 Select a Knowledge Script Group member and click **Properties** to update its monitoring values. In many cases, you can simply use the default schedule and values. To make changes:

Click this tab...	To...
Schedule	Change the default schedule for running the job. For more information, see Section 5.3, "Setting the Schedule," on page 70.
Values	Adjust the threshold and other parameters for the job. For more information, see Section 5.4, "Setting the Values to Monitor," on page 72 and Section 5.5, "Setting Override Values," on page 74.
Actions	Specify an action you want AppManager to take. For more information, see Section 5.6, "Specifying One or More Corrective Actions," on page 79.
Advanced	Specify whether to filter temporary event conditions and whether to collapse duplicate events for this job into a single event or automatically acknowledge them. Additionally, you can specify data collection options for this job. For more information, see Section 5.8, "Setting Advanced Job Properties," on page 82.

6 When you are satisfied with your changes, click **OK**.

7.3.1 Manually Propagating Base Knowledge Scripts to Knowledge Script Group Members

If the **Automatically update Derived KS properties for checked in Base KSs** option is enabled, when a base Knowledge Script is updated, AppManager automatically updates the derived Knowledge Scripts for that base script, including all Knowledge Script Groups where the derived script is a member. However, you must manually propagate the updated Knowledge Script Group members to any ad hoc jobs that were started by those members. For more information, see "Propagating Knowledge Script Group Members to Ad Hoc Jobs" on page 112.

If the automatic update option is not enabled, you must first manually propagate the base Knowledge Script to the Knowledge Script Group members, and then propagate the updated members to any ad hoc jobs that were started by those members.

With both automatic and manual propagation, AppManager merges any additions or deletions in the base script with the target scripts or jobs but retains the existing values, including override values.

For manual propagations, you can use the Knowledge Script Propagation wizard to view all of the derived Knowledge Scripts that are pending propagation and select specific scripts to process or update all scripts. You can also right-click a base Knowledge Script in a **Knowledge Scripts** view and use the menu options provided.

To manually propagate a base Knowledge Script to a Knowledge Script Group member using the wizard:

NOTE: This procedure is only necessary if the automatic update option is NOT enabled.

1 On the **Global Tasks** tab, click **Launch KS Propagation Wizard**.

2 On the **Propagation to Derived Knowledge Scripts** tab, select the specific base scripts that you want to propagate, or click **Select All** to propagate all pending changes.

3 When you are done making your selections, click **Propagate**, and then confirm that you are ready to propagate.

AppManager updates all direct copies of the script, all Knowledge Script Group members that are based on the script, and all monitoring policy jobs for the script.

To manually propagate a base Knowledge Script to a Knowledge Script Group member using the right-click option:

NOTE: This procedure is only necessary if the automatic update option is NOT enabled.

---

1 In the **Enterprise Layout** view of the **Navigation** pane, click the appropriate **Knowledge Scripts** view.

2 In the view pane, right-click the Knowledge Script you want, select **Propagate**, and select **To Derived Knowledge Scripts**.

3 Select the members that you want to update or click **Select All** to update all available scripts, click **OK**, and then confirm that you are ready to propagate.

AppManager updates all direct copies that you selected, all Knowledge Script Group members that you selected, and all monitoring policy jobs for the Knowledge Script Group members that you selected.

7.3.2   Propagating Knowledge Script Group Members to Ad Hoc Jobs

After a Knowledge Script Group member is updated from a derived Knowledge Script, you must propagate the updated member to any ad hoc jobs that were started by that member.

For manual propagations, you can use the Knowledge Script Propagation wizard to view all of the ad hoc jobs that are pending propagation and select specific jobs to process or update all ad hoc jobs. You can also right-click the Knowledge Script Group in a **Knowledge Scripts** view and use the menu options provided. When you use the right-click option to propagate to ad hoc jobs, AppManager updates all of the associated ad hoc jobs in all QDBs. If you want to propagate only to ad hoc jobs running in specific QDBs, use the wizard.

To propagate a Knowledge Script Group member to ad hoc jobs using the wizard:

1 On the **Global Tasks** tab, click **Launch KS Propagation Wizard**.

2 On the **Propagation to Ad Hoc Jobs** tab, expand the members that you want to propagate to view the QDBs where ad hoc jobs are running.

3 Select the QDBs where you want to update ad hoc jobs, or click **Select All** to propagate all pending changes.

4 When you are done making your selections, click **Propagate**, and then confirm that you are ready to propagate.

AppManager updates the jobs in the QDBs that you selected.

To propagate a Knowledge Script Group member to ad hoc jobs using the right-click option:

1 In the **Enterprise Layout** view of the **Navigation** pane, click the appropriate **Knowledge Scripts** view.

2 In the view pane, right-click the Knowledge Script Group that includes the member you want to propagate, select **Propagate**, and select **To Ad Hoc Jobs**.

3 Select the member you want to propagate or click **Select All** to select all members that are pending propagation, click **OK**, and then confirm that you are ready to propagate.

AppManager updates all ad hoc jobs that were started by the members you selected in all QDBs.

7.3.3 Copying a Knowledge Script Group

Copy a Knowledge Script Group to create a new Knowledge Script Group with a copy of each Knowledge Script Group member. In the copied Knowledge Script Group, the base Knowledge Scripts for the members of the original group are also the base scripts for the members of the copied group.

When you copy a Knowledge Script Group, AppManager prepends a new Knowledge Script Group identifier to any Knowledge Script jobs the Knowledge Script Group creates. For more information, see Section 7.1.2, "How Knowledge Script Groups Work," on page 107.

To copy a Knowledge Script Group:

1 In the **Enterprise Layout** view of the **Navigation** pane, select the **Knowledge Scripts** view you want.

2 In the view pane, select the Knowledge Script Group you want.

3 In the **Tasks** pane, click **Copy Knowledge Script Group**.

4 In the Copy Knowledge Script Group window, enter the new name for the Knowledge Script Group.

NOTE: To copy the Knowledge Script Group into a different category, change the prefix (*category_*) of the name. For example, to copy a Knowledge Script Group from the KSG category into a new category named KSG 2000, prefix the name with `KSG 2000_`.

---

5 (Optional) Provide a description for the copied Knowledge Script Group.

6 When you finish, click **OK**.

7.3.4 Deleting a Knowledge Script Group

To delete a Knowledge Script Group:

1 In the **Enterprise Layout** view of the **Navigation** pane, select the **Knowledge Scripts** view you want.

2 In the view pane, select the Knowledge Script Group you want.

3 In the **Tasks** pane, click **Delete**.

NOTE: You cannot delete a Knowledge Script Group that is configured to run as part of a monitoring policy. In this case, remove the Knowledge Script Group from any monitoring policies and then delete the Knowledge Script Group. For information about removing a Knowledge Script Group from a monitoring policy, see Section 7.5, "Changing a Monitoring Policy," on page 115.

7.4 Creating a Monitoring Policy

You can create a monitoring policy to automatically discover and monitor resources in a management group. Implementing a monitoring policy on a management group provides a powerful and flexible way to monitor the resources in your environment. For more information, see Chapter 4, "Managing a Group of Physical Computers or Logical Servers," on page 53.

AppManager implements a monitoring policy with one or more Knowledge Script Groups. Before you create a monitoring policy, create and properly configure a Knowledge Script Group. For more information, see Section 7.2, "Creating a Knowledge Script Group," on page 108.

When you create a monitoring policy, AppManager automatically monitors all matching objects. Be aware that if an object belongs to multiple members of a management group and matches a monitoring policy running on that management group, AppManager creates a policy-based job for each instance of the object. For more information about defining the members of a management group, see Section 4.3, "Working with Management Groups," on page 56.

When you create a monitoring policy, it might take up to one minute for the policy-based jobs to start. If you subsequently add or remove resources, it might take up to one minute for policy-based jobs to start or stop.

If necessary, you can stop and restart a policy-based job or configure override values for a particular parameter.

NOTE: You cannot use a monitoring policy to run reports; a Knowledge Script Group that was created with the **Install KS** option cannot be added to a monitoring policy. For more information, see Section 7.2, "Creating a Knowledge Script Group," on page 108.

To create a monitoring policy:

1 In the **Enterprise Layout** view of the **Navigation** pane, right-click the management group you want to monitor by policy.

2 In the **Tasks** pane, click **Management Group Properties > Policies.**

3 On the **Policies** tab of the Management Group Properties window, click **Add**.

   Note that from the Operator Console, you cannot remove a Knowledge Script Group that was added in the Control Center console. If you select a Knowledge Script Group that was added from the Control Center console, the **Remove** button is enabled but does not work.

4 Select one or more Knowledge Script Groups from the list. To select multiple items, use the **Shift** or **Ctrl** keys. Click **OK** when you are done making your selections.

5 You can configure the number of times that the monitoring policy should attempt to restart policy-based monitoring jobs:

Click...	To...
Always restart error jobs	Restart a policy-based monitoring job until its job status changes to Running. **NOTE:** The monitoring policy can only restart policy jobs.
Restart error jobs n times	Specify the number of retry attempts. If the specified number of retry attempts fails to restart the job, the job status remains Error.

NOTE: If AppManager cannot restart a policy-based job automatically, or if the policy-based job is scheduled to run once, when the job status is Stopped, you can manually restart the job. For more information, see Section 5.11.4, "Changing Job Status," on page 91.

6 When you are done configuring the policy, click **OK**. The monitoring policy creates policy-based jobs to monitor matching resources.

7.5 Changing a Monitoring Policy

You can change how a monitoring policy works by updating all jobs in the policy or by modifying a particular job.

To update all jobs in the monitoring policy, update the corresponding Knowledge Script Group members. You can change a monitoring policy by:

- Adding or removing Knowledge Script Groups.
- Changing the properties of existing members in a corresponding Knowledge Script Group.

For information about updating the Knowledge Script Group members, see Section 7.3, "Changing a Knowledge Script Group," on page 110.

You can modify a particular policy-based job by:

- Temporarily stopping a policy-based job. For example, if a resource in the management group is experiencing an unusually high number of events and you are aware of the problem, you can simply stop the job. For more information, see Section 5.11.4, "Changing Job Status," on page 91.
- Overriding the monitoring values of the policy-based job on a particular resource. For more information, see Section 5.5, "Setting Override Values," on page 74.

7.6 Stopping and Restarting Policy-based Jobs

Stopping and restarting a policy-based job is the same as stopping an ad hoc job. For more information, see Section 5.11.4, "Changing Job Status," on page 91.

8 Using Service Map Views

This chapter describes what **Service Map** views are and how you can use them to more effectively view and manage the resources in your monitored environment.

8.1 Understanding Service Map Views

A **Service Map** view provides a visual representation of the relationships between objects you monitor with AppManager and displays event conditions on these objects that might disrupt any interactions between objects. **Service Map** views also allow you to respond to those events. You can use a single **Service Map** view to visually represent and monitor all of the computing and network infrastructure that underlies a particular business process. For example, you can create a **Service Map** view of the Web servers, database servers, and databases that customers use to make purchases. From a **Service Map** view, you can also:

 ◆ Manage events.
 ◆ Manage jobs.
 ◆ View details on discovered resources.
 ◆ Manage custom properties.
 ◆ Manage data streams.

By creating a visual representation of the relationships between the components in your infrastructure, you can easily monitor the entire infrastructure and develop a more immediate understanding of how problems with one component are likely to affect the business processes that these components support. To continue the previous example, if you identify critical events on a database server, this could indicate communication problems with the Web servers providing order information to the database server, which could in turn create problems with customers attempting to place orders.

Service Map views can also represent:

 ◆ The computers, servers, and mail stores that support corporate e-mail.
 ◆ The computers, servers, and databases that make up your customer relationship management system.
 ◆ The domain controllers, sites, and domains of your Active Directory.

Service Map views can also contain *embedded views*, including other **Service Map** views, **Events** views, and **Servers** views.

The discovered objects in a **Service Map** view indicate the most severe open event condition associated with that object or one of its child objects. Embedded views also indicate the most severe open event conditions associated with that view. When you see an indication of a problem, you can perform additional tasks, such as acknowledge events, change corresponding job properties, or initiate diagnostic or corrective action.

8.2 Working with Service Map Views

The following sections provide details about how to manage events, jobs, and custom properties for discovered objects in a **Service Map** view. For more information, refer to the following topics:

8.2.1 Displaying a Service Map View

To display a **Service Map** view, select the **Service Map** view you want from the **Enterprise Layout** view of the **Navigation** pane. The **Service Map** view is displayed in the view pane.

The **Service Map** view consists of:

- Icons that represent discovered objects.
- Icons that represent embedded views.
- Drawing objects that help you understand the relationships in the **Service Map** view.

Each icon indicates the type of object or embedded view. By default, each icon includes a label that can provide additional information. The label for discovered objects can vary depending on the type of object. For more information, see Section 8.3.2, "Adding and Deleting Objects," on page 121. Embedded views can be labeled with the original name of the view or a different name.

8.2.2 Understanding Event Indicators

The icon for an AppManager object or embedded view changes to reflect its most severe Open event condition. If the object is a parent, for example a computer object, the object icon changes to indicate the most severe Open event condition associated with itself or one of its children (such as the memory and CPU objects).

The event severity level for an object is determined by the repository preference settings for the corresponding QDB. To view the current event severity levels, use the Operator Console to log into the QDB you want. Then from the **File** menu select **Preferences**, click the **Repository** tab, and then click **Severity**.

A **Service Map** view provides visual indicators for the following conditions:

- No events associated with this object or its children
- Severe event
- Warning event
- Informational event
- Diagnostic event

8.2.3 Managing Events and Jobs

From a **Service Map** view, you can manage events and jobs for:

- **A selected AppManager object.** Select an object and associated events and jobs are displayed in the **Events** and **Jobs** tabs of the **Server Information** pane.

- **All objects in an embedded Service Map, Servers, or Events view.** Select the embedded view and associated events and jobs are displayed in the **Events** and **Jobs** tabs of the **Server Information** pane.

The **Events** tab of the **Server Information** pane enables you to manage events as you normally would from a **Servers** or **Events** view. You can also open an embedded view to manage events from the actual view. To open an embedded view, double-click the embedded view icon or right-click the icon and click **Open Embedded View**. For information about managing events, see Chapter 6, "Responding to Events," on page 95.

The **Jobs** tab of the **Server Information** pane enables you to manage jobs from a **Service Map** view as you would from a **Servers** or **Jobs** view. Note that from a **Service Map** view, you cannot create new jobs. For information about managing jobs, see Chapter 5, "Running Monitoring Jobs," on page 67.

To navigate between views, use the **Back** and **Forward** buttons on the toolbar.

NOTE: If you do not see the **Server Information** pane, it might be collapsed. To expand the **Server Information** pane, click the **Expand** button in the lower right corner of the **Service Map** view.

8.2.4 Viewing Detailed Information about Discovered Resources

From a **Service Map** view, you view discovery details for:

- **A selected AppManager object.** Select an object and associated discovery details are displayed in the **Details** tab of the **Server Information** pane.

- **All objects in an embedded Service Map or Servers view.** Select the embedded view and associated discovery details are displayed in the **Details** tab of the **Server Information** pane. Note that you cannot view discovery details from an embedded **Events** view.

The **Details** tab of the **Server Information** pane enables you to view discovery details from a **Service Map** view as you would from a **Servers** view. You can also open an embedded **Service Map** view or **Servers** view to display discovery details from the actual view. To open an embedded view, double-click the embedded view icon or right-click the icon and click **Open Embedded View**. For information about managing events, see Chapter 6, "Responding to Events," on page 95.

To navigate between views, use the **Back** and **Forward** buttons on the toolbar.

NOTE: If you do not see the **Server Information** pane, it might be collapsed. To expand the **Server Information** pane, click the **Expand** button in the lower right corner of the **Service Map** view.

8.2.5　Managing Custom Properties

From a **Service Map** view, you manage custom properties for:

- **A particular server.** Select a server object and associated custom properties are displayed in the **Custom Properties** tab of the **Server Information** pane.

- **All servers in an embedded Service Map or Servers view.** Select the embedded **Service Map** view or **Servers** view to display all associated custom properties information in the **Custom Properties** tab of the **Server Information** pane. Note that you cannot manage custom properties from an embedded **Events** view.

The **Custom Properties** tab of the **Server Information** pane enables you to manage custom properties from a **Service Map** view as you would from a **Servers** view. You can also open an embedded **Service Map** view or **Servers** view to manage custom properties from the actual view. To open an embedded view, double-click the embedded view icon or right-click the icon and click **Open Embedded View**. For information about managing custom properties, see Section 4.5, "About Custom Properties," on page 61.

To navigate between views, use the **Back** and **Forward** buttons on the toolbar.

8.2.6　Managing Data Streams

From a **Service Map** view, you manage data streams for:

- **A particular server.** Select a server object and any data streams are displayed in the **Charts** tab of the **Server Information** pane.

- **All servers in an embedded Service Map or Servers view.** Select the embedded **Service Map** view or **Servers** view to display all associated data streams in the **Charts** tab of the **Server Information** pane.

The **Charts** tab of the **Server Information** pane enables you to manage data streams from a **Service Map** view as you would from a **Servers** view. You can:

- Locate and chart data streams related to specific servers, jobs, and events.

- Obtain detailed information about data points of interest.

- Obtain real-time updates based on the update interval you specify.

- Print, email, or export charts and data stream details.

For more information about the **Charts** tab, see Section 11.3, "Working with the Charts Tab," on page 166.

8.3　Designing Service Map Views

To design a **Service Map** view, create a new **Service Map** view or check out an existing **Service Map** view from the CCDB. The Control Center console provides a variety of tools for designing the **Service Map** view.

The **Enterprise Layout** view of the **Navigation** pane enables you to select the objects, **Events** views, **Servers** views, and other **Service Map** views you want to embed into the **Service Map** view. When embedding a view, you can embed views from other management groups to provide the most flexible filtering mechanism and display only the objects you want in the corresponding view. To embed an object or view, drag the item onto the view pane.

The view pane enables you to edit the contents of a **Service Map** view. While a **Service Map** view is checked out, the workspace banner indicates the **Service Map** view is "Under Construction" and displays the name and description of the **Service Map** view.

The Service Map ribbon enables you to add rectangles, ellipses, rounded rectangles, text boxes, custom node objects, and links (lines or arrows) between objects. You can also set a background image.

8.3.1 Creating Service Map Views

Use the Control Center console to create and edit **Service Map** views. From the **Enterprise Layout** view of the **Navigation** pane, you can create a new **Service Map** view or check out an existing **Service Map** view from the CCDB. Note that you can continue to use other Control Center views while a **Service Map** view is checked out. You can also create a new **Service Map** view and embed it in another **Service Map** view.

To create a Service Map view:

1 In the **Enterprise Layout** view of the **Navigation** pane, right-click a management group or folder where you want to create the **Service Map** view, and then click **New > Service Map.**

2 In the **General** tab, specify the options you want for the **Service Map** view. For more information about these options, see the Help.

3 In the **Grid** tab, specify the grid options you want for the **Service Map** view. For more information about grid options, see the Help.

4 Click **OK**.

To change the properties of a Service Map view:

1 In the **Enterprise Layout** view of the **Navigation** pane, right-click the **Service Map** view you want, and then click **View Properties > General**.

2 In the **General** tab of the Service Map View Properties window, update the options you want for the **Service Map** view. For more information about these options, see the Help.

3 In the **Grid** tab, update the grid options you want for the **Service Map** view. For more information about grid options, see the Help.

4 Click **OK**.

8.3.2 Adding and Deleting Objects

To add discovered objects to a **Service Map** view, drag and drop objects onto the view pane from the **Enterprise Layout** view of the **Navigation** pane. You can also add objects from the **Servers** group in the **Tasks** pane. Add discovered objects to a **Service Map** view to manage events, jobs, custom properties, and data streams and view discovery details for the object. If you want to represent an object that is not discovered, you can use a custom node. For more information, see Section 8.3.10, "Adding a Custom Node," on page 125.

The default object name uses the format *<computer>\<object>*. For example, an IIS server installed on WebServer01 has the default name WebServer01\IIS Server:WebServer01. The memory object for the computer WebServer01 is named WebServer01\Memory.

To add objects to a **Service Map** view, the **Service Map** view must be checked out of the CCDB. Objects do not display event status information until after you check the **Service Map** view into the database.

After you add AppManager objects to the workspace, you can visually connect them to other AppManager objects and embedded views using lines and arrows. For more information, see Section 8.3.8, "Connecting AppManager Objects," on page 124.

To delete an object from a **Service Map** view, right-click the object and then click **Delete**. This removes the object from the **Service Map** view.

If an object is deleted from the **Enterprise Layout** view of the **Navigation** pane, the corresponding object in any **Service Map** views is not deleted. Instead the object is displayed with a Deleted icon. If the same object is re-discovered, the object is not updated in any **Service Map** views and continues to indicate the object has been deleted. You must manually delete the object from any **Service Map** views and add it again.

8.3.3 Modifying Object Properties

Once you add an object to a **Service Map** view, you can change the properties of the object, such as the object name, label color, and icon size. You can also use custom icons.

To change the properties of an object in a Service Map view:

1 Check out the Service Map with the object whose properties you want to modify.

2 Right-click the object you want to modify, and then click **Object Properties**.

3 In the Object Properties window, make your changes. For more information, see the Help.

4 Click **OK**.

8.3.4 Embedding a View

You can embed a **Service Map** view, **Servers** view, or **Events** view in a **Service Map** view. To embed a view, drag and drop the **Servers**, **Events**, or **Service Map** view you want from the **Enterprise Layout** view of the **Navigation** pane onto the view pane. To embed a view, the **Service Map** view must be checked out of the CCDB.

Embedded views offer the following capabilities:

◆ **Service Map** views provide "drill down" access to other **Service Map** views that give more detailed information. For example, to reduce the complexity of a **Service Map** view, you can create a top-level **Service Map** view with embedded **Service Map** views that have additional details.

◆ **Servers** views enable you to manage a dynamic list of events, jobs, and custom properties, and view discovery details. Take advantage of the advanced filtering capabilities provided by rule-based management groups and embed a **Servers** view from a management group that filters only the objects that correspond to the **Service Map** view. As the rule-based management group filters matching objects, the embedded **Servers** view reflects those changes.

For example, in a **Service Map** view that displays a business process that relies upon SQL Server databases named DB_Commerce, instead of adding each database object, embed a **Servers** view that belongs to a rule-based management group which filters databases named DB_Commerce. If your business infrastructure adds or removes a DB_Commerce database, the embedded **Servers** view automatically reflects that change.

◆ **Events** views enable you to manage a dynamic list of events. Take advantage of the advanced filtering capabilities provided by rule-based management groups and embed an **Events** view from a management group that filters only the objects that correspond to the **Service Map** view. As the rule-based management group filters matching objects, the embedded **Events** view reflects corresponding events.

For example, you can create a **Service Map** view that functions as a simple warning system for the availability of a Web site you are hosting. If the monitored Web site becomes unavailable, the related object will indicate an event condition.

Note that you can embed **Events**, **Servers**, and **Service Map** views from any management group.

Once you add a view to a **Service Map** view, you can change the properties of the view, such as the name, label color, and icon size. You can also use custom icons.

To change the properties of a view embedded in a Service Map view:

1 Check out the Service Map with the view whose properties you want to modify.

2 Right-click the view you want to modify, and then click **Object Properties**.

3 In the Sub View Properties window, make your changes. For more information, see the Help.

4 Click **OK**.

8.3.5 Editing Objects in a Service Map

Once you add objects to a **Service Map** view, you can edit those objects using the tools available on the **Service Map** tab of the ribbon. After you select an object, you can:

◆ Edit the properties of the object by right-clicking to select **Object Properties**.

◆ Move the object by dragging it with the mouse.

◆ Copy the object to the clipboard.

◆ Paste the object to the same or different **Service Map** views.

◆ Delete the object.

Note that you can undo or redo an action by clicking **Undo** and **Redo** in the **Edit** group on the **Service Map** tab of the ribbon.

To select multiple objects, press **Ctrl** while making your selections or drag the cursor around the objects you want.

8.3.6 Showing and Hiding the Grid

Use the grid to help you align objects in the workspace. With the grid displayed, when you move an object in the workspace, the object is automatically aligned to the nearest grid point. By automatically aligning objects, also known as "snap to grid," you can quickly and easily align objects.

By default, the workspace is configured to display the grid. To hide or show the grid, right-click the workspace and click **Show/Hide Grid**.

8.3.7 Adding and Formatting Text

You can add text to a **Service Map** view. Note that most objects that appear in a **Service Map** view can be configured to display corresponding label text.

After you add text, you can configure the text properties to specify font size, color, and automatic line wrapping.

To add text:

1 Check out the Service Map where you want to add text.

2 On the **Service Map** tab, in the **Tools** group, click **Text**.

3 In the view pane, click on the Service Map where you want to place the text.

4 Type the text you want to add to the **Service Map** view.

5 When you finish adding text, click on the view pane outside the text box.

6 (Conditional) If you need to format the text, right click the text box and select **Object Properties**.

   6a Modify the text properties, such as font and background colors, text wrapping, and font size.

   6b Click **OK**.

8.3.8 Connecting AppManager Objects

Connect AppManager objects and embedded views to indicate relationships and direction. You can connect AppManager objects and embedded views using the **Link with Left Arrow**, **Link with Right Arrow, Link with No Arrows**, and **Link with Double Arrows** link tools. You cannot use the link tools to draw lines or to create links between shapes you have drawn, such as an ellipse, rectangle, or rounded rectangle.

After connecting objects, you can configure the link properties to:

   ◆ Add a label to the link.
   ◆ Change the width and color of the link.
   ◆ Change the link end style.
   ◆ Use a curved line.

To create a link:

1 With the **Service Map** view checked out, on the **Service Map** tab of the ribbon, in the **Tools** group, click the **Link with Left Arrow**, **Link with Right Arrow, Link with No Arrows**, or **Link with Double Arrows** link tool.

2 In the view pane, click the objects you want to connect. The link is created using the default link properties.

Once you add a link to a **Service Map** view, you can change the properties of the link, such as the label, link color, and link width. You can also change end styles.

To change the properties of a link in a Service Map view:

1 Check out the Service Map with the links whose properties you want to modify.

2 Right-click a link you want to modify, and then click **Object Properties**.

3 In the Object Properties window, make your changes. For more information, see the Help.

4 Click **OK**.

8.3.9 Drawing an Ellipse, Rectangle, or Rounded Rectangle

Use the ellipse, rectangle, and rounded rectangle tools to draw shapes.

To draw a shape:

1 With the **Service Map** view checked out, on the **Service Map** tab of the ribbon, in the **Tools** group, select the **Ellipse**, **Rectangle**, or **Rounded Rectangle** tool.

2 Click and drag the tool in the view pane to create the shape.

3 To resize the shape, select the pointer and drag an object handle.

Once you add a shape to a **Service Map** view, you can change the properties of the shape, such as color, fill type, border width, and transparency.

To change the properties of a shape in a Service Map view:

1 Check out the Service Map with the shape whose properties you want to modify.

2 Right-click a shape you want to modify, and then click **Object Properties**.

3 In the Shape Properties window, make your changes. For more information, see the Help.

4 Click **OK**.

8.3.10 Adding a Custom Node

Use a custom node to represent an AppManager object that is not available from the **Enterprise Layout** view of the **Navigation** pane. This object can be used as a placeholder.

To add a custom node, on the **Service Map** tab, in the **Tools** group, click **Custom Node,** and then click in the workspace where you want to add the custom node. The custom node is created with a default label.

After you add a custom node, you can edit its properties.

To change the properties of a custom node:

1 Check out the **Service Map** view with the custom node whose properties you want to modify.

2 Right-click the custom node you want to modify, and then click **Object Properties**.

3 In the Custom Node Properties window, make your changes.

4 Click **OK**.

8.3.11 Adding a Background Image

To clarify the relationships between objects, you can also add a background image such as an underlying geographical map or floor plan. A background image is anchored to the top-left corner of the map, and is displayed in its original dimensions. If you need to resize the image, use an image-manipulation program, and then reload it.

You can add a background image from a physical disk, the Windows clipboard, or the CCDB. After you add a background image, it is stored in the CCDB.

You can use images in the following formats:

- BMP
- EMF
- GIF
- JPG
- PNG
- TIFF
- WMF

NOTE: If you want the Control Center console to load the **Service Map** views faster, use the JPG or PNG formats for the background image.

To add or change a background image:

1 With the **Service Map** view checked out, on the **Service Map** tab of the ribbon, in the **Tasks** group, click **Set Background Image** or right-click the view pane and click **Set Background Image**.

2 In the Background Image Properties window, select an option to load the image.

For...	Do this...
File	Click this option and click **Browse** to select the file from a physical disk and store it in the CCDB.
Database	Click this option and click **Browse** to select the file from a list of files in the CCDB.
Windows Clipboard	Click this option and click **Paste** to select an item from the Windows clipboard and store it in the CCDB.
None	Click this option to not display a background image.

3 Click **OK**.

8.3.12 Checking a Service Map View Into and Out of the CCDB

To save your changes, check the **Service Map** view into the CCDB. To check in a **Service Map** view, on the **Service Map** tab of the ribbon, in the **Tasks** group, click **Check In**. In the **Enterprise Layout** view of the **Navigation** pane, you can also right-click the **Service Map** view and click **Check In**.

When you check a **Service Map** view in, the object icons change to reflect event conditions. If you check in a **Service Map** view that references a large number of objects, some objects in the **Service Map** view might remain gray while information about their event conditions is retrieved from the CCDB. Once that information is retrieved, the object icons changes to reflect event conditions. During this time, the **Service Map** view is operable.

To check a **Service Map** view out of the CCDB, on the **Service Map** tab of the ribbon, in the **Tasks** group, click **Check Out**. In the **Enterprise Layout** view of the **Navigation** pane, you can also right-click the **Service Map** view and click **Check Out**. In the view pane, you can now edit the **Service Map** view.

Note that while a **Service Map** view is checked out, it cannot be checked out by another user. To determine who has checked out a **Service Map** view, display the Service Map view properties. For more information, see Section 8.3.1, "Creating Service Map Views," on page 121.

After you check out a **Service Map** view, event status for the objects in the view is not displayed.

8.3.13 Reverting to the Previous Version of a Service Map View

When you check in a **Service Map** view, the revision number of the view is updated.

While you have a **Service Map** view checked out, you can revert to the most recent revision number. For example, if you have made a number of changes that you do not want to implement, you can revert to the last revision of the **Service Map** view.

To revert to the last revision of a **Service Map** view, on the **Service Map** tab of the ribbon, in the **Tasks** group, click **Revert**. You can also right-click the **Service Map** view in the view pane, and then click **Revert**. The last revision of the **Service Map** view is displayed in read-only mode.

8.3.14 Exporting a Service Map View to an Image File

The `ServiceMapImageExporter.exe` command line program enables you to export a **Service Map** view to an image file. This program is located in:

`<install_dir>\NetIQ\AppManager\Control Center\bin`

This command line program is useful when you want to view the status of several **Service Map** views, for example, as part of a presentation. You can use Microsoft Task Scheduler to run the `ServiceMapImageExporter.exe` command line program and export all of your **Service Map** views.

To configure Microsoft Task Scheduler to run more frequently than on a daily basis, configure the Advanced properties for the task. Alternatively, you can use the NTAdmin_RunDos Knowledge Script to run the `ServiceMapImageExporter.exe` command line program.

To run this utility from a command prompt, first change directories to:
`<install_dir>\NetIQ\AppManager\Control Center\bin`.

Here are some examples of how to run `ServiceMapImageExporter.exe`:

- For Windows authentication:
  ```
  Servicemapimageexporter.exe /trust /S <server_name> /V "AppManager\Master\New
  Service Map View 1"
  ```
- For SQL authentication:
  ```
  Servicemapimageexporter.exe /U <sql_user> /P <password> /s <server_name> /V
  "AppManager\Master\New Service Map View 1"
  ```

For different image formats:

- For PNG format:
  ```
  Servicemapimageexporter.exe /trust /s <server_name> /V "AppManager\Master\New
  Service Map View 1" /I PNG
  ```
- For JPEG format:
  ```
  Servicemapimageexporter.exe /trust /s <server_name> /V "AppManager\Master\New
  Service Map View 1" /I JPEG
  ```
- For BMP format:
  ```
  Servicemapimageexporter.exe /trust /s <server_name> /V "AppManager\Master\New
  Service Map View 1" /I BMP
  ```
- For GIF format:
  ```
  Servicemapimageexporter.exe /trust /s <server_name> /V "AppManager\Master\New
  Service Map View 1" /I GIF
  ```
- For TIFF format:
  ```
  Servicemapimageexporter.exe /trust /s <server_name> /V "AppManager\Master\New
  Service Map View 1" /I TIFF
  ```

In the examples offered, `<server_name>` indicates the name of the Control Center repository server. Information about how to configure the `ServiceMapImageExporter.exe` program is available from the command line by typing `ServiceMapImageExporter.exe` and pressing **Enter**.

8.3.15 Viewing Revision History for a Service Map View

The revision history helps you keep track of when and by whom changes have been made to a **Service Map** view.

To see the revision history for a Service Map view:

1 In the **Enterprise Layout** view of the **Navigation** pane, select the **Service Map** view you want.

2 On the **Service Map** tab of the ribbon, in the **View** group, click **Revision History**.

3 The Revision History window displays the following information:

Column	Explanation
Revision	The number of times the map has been revised. Note that you can only revert to the most recent version of a **Service Map** view.
Action Time	The date and time of a particular action on the **Service Map** view.
User	The user who initiated the action on the **Service Map** view.
Action	The action taken on the **Service Map** view (for example, Created, Checked Out, Checked In).

4 Click **OK** to close the window.

8.4 Using Security Settings

The Control Center console provides the following permissions for **Service Map** views:

- ◆ Allowed to access Service Map views
- ◆ Allowed to create and modify Service Map views
- ◆ Allowed to delete Service Map views

Note that **Events** view and **Servers** view permissions apply to embedded **Events** and **Servers** views. If a user does not have permission to access **Events** views or **Servers** views, the embedded **Events** or **Servers** view is displayed with a Deleted icon and the **Server Information** pane does not display the corresponding **Events** or **Jobs** tab. For more information about configuring Control Center security, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

9 Deploying AppManager to Agent Computers

This chapter provides an overview of how to use the Control Center console to deploy the AppManager agent and modules to Windows computers in your environment.

For more information about installing the AppManager agent on Windows computers, including installing the AppManager agent on a local computer and performing a silent installation of the AppManager agent, see the *Installation Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2). For more information about installing the agent on a UNIX or Linux computer, see the NetIQ UNIX Agent documentation, which is included with the AppManager UNIX agent installation kit.

9.1 What Is Deployment?

The Control Center console allows you to quickly and easily deploy the AppManager agent and AppManager modules to the remote Windows computers in your environment. Deployment provides:

- Automated detection of new agent computers and applications.
- Automated deployment of agents and modules.
- Rapid response to changing environments.
- Multiple, firewall-friendly deployment servers.
- Customized deployment schedules to meet corporate maintenance windows.

To deploy agents or modules, you create or enable deployment rules, then approve the deployment tasks created by the deployment rules. Deployment tasks include the required credentials to run on the agent computers. Each AppManager agent provides the details about applications installed on the agent computer so that in the future you can easily deploy additional modules and updates to the agent computer.

Deploying an agent discovers and adds the agent computer to the default **Master** management group. You can add agent computers to other management groups based on the resources you want to monitor.

Deployment also enables you to upgrade or uninstall agents and modules.

Instead of using the Control Center console to deploy the UNIX agent, you use UNIX Agent Manager. For information about deploying the UNIX agent, see the NetIQ UNIX Agent documentation, which is included with the AppManager UNIX agent installation kit.

9.2 How Does Deployment Work?

This section describes how you can use the Control Center console to deploy agents and modules on agent computers. Control Center needs the following to successfully deploy agents on a Windows computer:

- Deployment rules

◆ Deployment Service and deployment server

◆ Deployment Web Service and Web Depot

You will follow the same basic steps to deploy agents and modules on Windows agent computers.

To deploy agents and modules on agent computers:

1 Verify the default deployment credentials, schedule, and notification emails. For more information, see Section 3.2.3, "Deployment Options," on page 42.

2 If necessary, check in an installation package. The Control Center console installs with a default set of installation packages, and you can add or upgrade installation packages as they become available. For more information, see Section 9.4, "Managing Installation Packages," on page 135.

3 Create a deployment rule or modify an out-of-the-box deployment rule, and then enable the rule. For more information, see Section 9.5, "Working with Deployment Rules," on page 137.

4 Approve the deployment task created by the deployment rule. For more information, see Section 9.6, "Managing Deployment Tasks," on page 145.

5 View the results. For more information, see Section 9.7, "Viewing the Results," on page 148.

9.2.1 Deployment Rules

Once you install the Control Center deployment components, you use the Control Center console to configure a **deployment rule**.

A **deployment rule** consists of a list of agent computers, a set of installation packages you want to install, the credentials needed to perform the installation, and a deployment schedule. For your convenience, Control Center includes default rules to install some of the out-of-the-box **installation packages**, including the NetIQ AppManager Windows Agent. For more information, see Section 3.2.3, "Deployment Options," on page 42.

The Control Center console provides an easy-to-use wizard interface to help you create **deployment rules**. Deployment rules allow you to identify computers where you want to install agents and modules based upon a variety of criteria, including the following:

◆ Agent

◆ Domain

◆ Specific computers

◆ Computer list file

◆ Management groups in the Control Center console

AppManager will not install any modules unless you first install the AppManager agent on the computer you want to monitor. For more information about creating deployment rules, see Section 9.5.1, "Creating or Modifying Rules," on page 137.

9.2.2 Enabling Deployment Rules and Generating Deployment Tasks

You must first enable a deployment rule before you can generate deployment tasks with the rule. To enable a rule, you need to verify the credentials, schedule, and email notification (if any) for the rule. If an installation package requires configuration, you must also configure the package before you can enable a deployment rule. For more information, see Section 9.5.11, "Enabling and Disabling Deployment Rules," on page 142.

Once you enable a rule, the Deployment Service generates a deployment task for each installation package on each agent computer. For example, if a rule has five agent computers and three packages to install, the Deployment Service generates a maximum of 15 deployment tasks. A deployment task is not generated if:

- An installation package is already installed on an agent computer.
- The agent computer does not pass pre-deployment checks for the installation package.

You must approve each deployment task created by the deployment rule before the deployment tasks will run.

9.2.3 Deployment Service and Deployment Server

The **Deployment Service** communicates with the CCDB to process deployment rules and tasks. Tasks run on the agent computer using credentials provided in the deployment rule.

The computer where you install the Deployment Service is the **deployment server**. If you configure more than one Deployment Service to work with your CCDB, you can configure a deployment rule to deploy the agent or modules by using a certain deployment server.

The Deployment Service must be configured to run as the Local System account. If the Deployment Service is not configured to run as the Local System account, you might encounter problems with the Background Intelligent Transfer Service (BITS), which the Deployment Service uses to communicate with the Web Depot.

NOTE: Multiple Deployment Services allow for collocating with network segmented (firewall) AppManager agents in your environment.

For more information about specifying the Deployment Service, see Section 9.5.7, "Deployment Service," on page 141.

9.2.4 Deployment Web Service and Web Depot

The agent uses the **Deployment Web Service** to communicate with the deployment server. The **Deployment Web Service** consists of two web services that you install on a Microsoft Internet Information Services (IIS) server. The computer where you install the Deployment Web Service and check in the installation packages is the **Web Depot**.

The **Deployment Web Service** performs the following actions:

- Checks in installation packages to the Web Depot.
- Distributes installation packages to the Deployment Services. The Deployment Web Service uses Microsoft Background Intelligent Transfer Service (BITS) server extensions to distribute installation packages to the Deployment Service.
- Communicates with Deployment Services that are across a firewall.

The following figure illustrates a simplified view of this architecture:

* Web server only communicates with the repository
  when acting as a proxy for the deployment service.
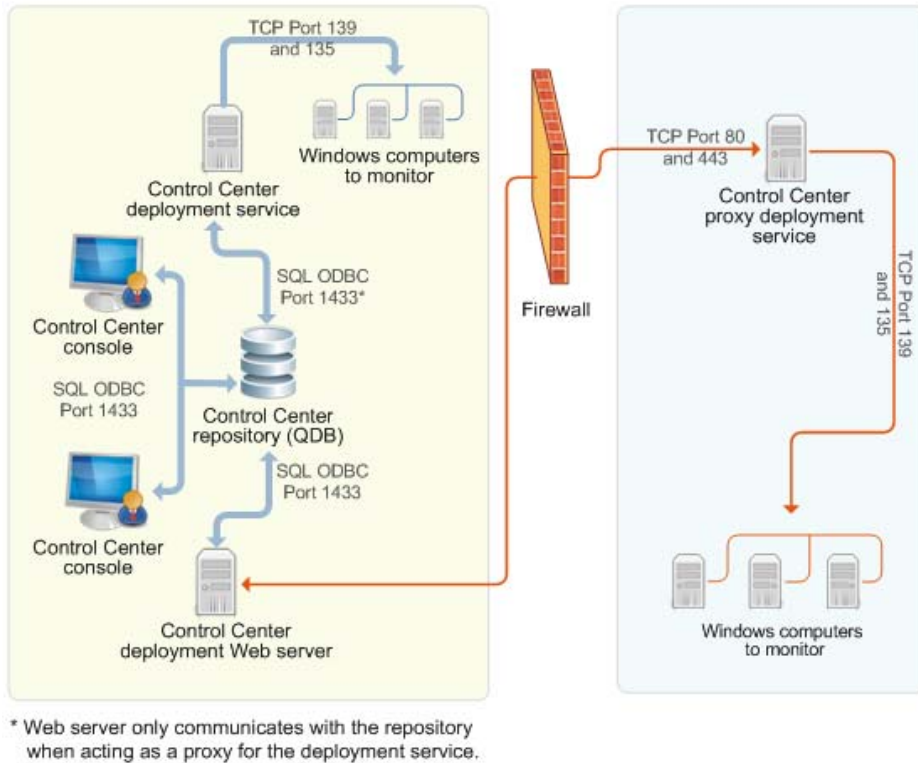
For more information about AppManager ports, see the *Installation Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

9.2.5 Communication Flow Between the Remote Deployment Components

When you approve a deployment task:

- ◆ The Deployment Service checks the CCDB and identifies the approved task.
- ◆ The Deployment Service retrieves the corresponding installation package from the Web Depot on the deployment Web server and runs the installation package according to the configuration of the deployment rule on the agent computer.
- ◆ The Deployment Service scans the list of agent computers. For default computer detection interval, this happens after you enable the rule. For non-default computer detection interval, the interval must pass after you enable the rule, before the Deployment Service scans the list of computers. If a computer is found that matches the rule, the Deployment Service generates a deployment task. The deployment task only runs if the task is scheduled to run at a particular time or if the task must be manually approved.
- ◆ The Deployment Service updates the task status to **Active** while the task is being processed and then **Completed** once the installation has completed.
- ◆ The agent computer reports software inventory information back to the management server. For example, if you installed the AppManager agent, software inventory information would include the version of the agent.

If you have version 7.*x* agents in your environment, those agents report software inventory information back to the Deployment Web Service.

   ◆ The management server forwards the software inventory information from the AppManager agent to the QDB, and the command queue service manages a process that synchronizes the information from the QDB to the CCDB.

   For version 7.*x* agents, the Deployment Web Service forwards the software inventory information from the AppManager agent to the CCDB.

   ◆ The Control Center console retrieves the software inventory information from the CCDB.

The following figure illustrates a simplified view of the communication flow between version 8.0 remote deployment components:
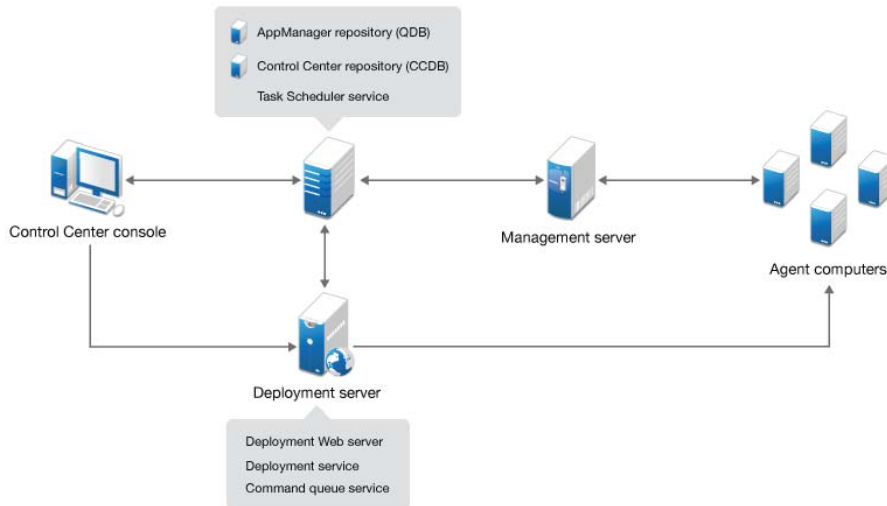
9.3 Deployment Preferences

The Control Center console allows you to specify certain deployment options, such as default credentials and email notification lists and how frequently the Deployment Service re-runs rules to look for new agent computers. For more information, see .

9.4 Managing Installation Packages

An installation package is an installation program that is run by the Deployment Service as part of a deployment task. You can perform the following tasks with Installation packages:

   ◆ Install, reinstall, uninstall, or upgrade the AppManager agent.

   ◆ Install, upgrade, or uninstall a module.

To install the AppManager agent, you should select the NetIQ AppManager Agent package and the *AppManager for Application* packages that correspond to the applications you want the agent to monitor. For example, to install the AppManager agent and Microsoft SQL Server module, you would need the deployment package for the NetIQ AppManager Windows Agent and the AppManager for SQL Server packages. Control Center creates a separate deployment task for each package.

NOTE: The deployment tasks for a module are not created until after you install the AppManager agent and it appears in the Software Inventory list.

You specify whether to install or uninstall agents and modules when you create or modify a deployment rule. Deployment rules can either install or uninstall agents and modules. The same rule cannot install some components and uninstall others.

The installation package consists of an XML file that you might need to check in to Control Center. By default, all packages available with the current release of AppManager are checked in when you install Control Center. If you download a module released later than the AppManager release, check in the XML file included in the download package. For more information, see Section 9.4.3, "Checking in Installation Packages," on page 136.

9.4.1 Viewing Installation Packages

In the **Navigation** pane, click **Deployment** and then click **Packages** to view a list of available packages.

To view the properties of a package, right-click the package you want and click **Deployment Package Properties**.

9.4.2 Checking for Updated Deployment Packages

Before deploying an installation package, you can check to be sure you have the latest version of the module in the deployment package. This is an easy way to be sure you are deploying only the most recent module software and it can help you avoid needing to upgrade modules later on.

To check for updated deployment packages:

1 In the **Navigation** pane, select the **Deployment** view.

2 Click **Packages**.

3 On the **Main** tab of the ribbon, click **Tasks > Packages View Tasks > Check for Deployment Package Updates**. A browser window displays any available module updates for all deployment packages.

4 Click the download icon to download and install the updated modules you want.

9.4.3 Checking in Installation Packages

If you download and want to deploy a module later than the version included with the AppManager installation kit, check in the module installation package. Before you can check in an installation package, ensure that you have configured the Control Center console with the name of the IIS Server where you have installed the Deployment Web Service.

The name of the Web Depot computer is configured at the time of installation. For more information about configuring the Deployment Web Service, see Section 3.2.3, "Deployment Options," on page 42.

To check in additional packages after installation:

1 Log on to the Control Center console using an account that is a member of a user group with deployment permissions. For more information about permissions, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

2 In the **Navigation** pane, select the **Deployment** view.

3 Click **Packages**.

4 In the Tasks pane, click **Check in Deployment Packages**.

5 Navigate to the folder where you have saved the `XML` file for the module and select the file.

6 Click **Open**. The Deployment Package Check in Status window displays the status of the package check in.

9.5 Working with Deployment Rules

Deployment rules are how you define the agent and module installation packages you want to install and the computers where you want to run the installation packages. You also define the following items in deployment rules:

- ◆ The credentials to use for installation
- ◆ The Deployment Service to use
- ◆ The deployment schedule
- ◆ Email notification of deployment success or failure

You can create new deployment rules or modify existing deployment rules, including out-of-the-box deployment rules installed when you install the Control Center console. To create or modify deployment rules, you use the Deployment Rule Wizard.

9.5.1 Creating or Modifying Rules

You can create a new rule, modify a rule you created previously, or modify one of the out-of-the-box deployment rules.

Control Center provides out-of-the-box rules for some installation packages, including the NetIQ AppManager Windows Agent. The default rules use the naming convention DefaultRule - *application* where *application* is the name of an application on an agent computer. If there is no out-of-the-box deployment rule, create a deployment rule.

If you want to use an out-of-the-box rule, you can modify the properties of the rule. For more information about modifying a rule, see Section 9.5.14, "Changing the Properties of a Deployment Rule," on page 144.

To create or modify a rule:

1 Log on to the Control Center console using an account that is a member of a user group with deployment permissions. For more information about permissions, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

2 In the **Navigation** pane, click **Deployment**.

3 In the **Navigation** pane, click **Rules**.

4 (Conditional) If you want to create a new rule, in the **Tasks** pane, click **Create New Deployment Rule**.

5 (Conditional) If you want to modify an existing rule, click the rule you want to modify, and then in the **Tasks** pane click **Edit Deployment Rule**.

6 In the Deployment Rule Wizard, configure the rule. In some cases, you can use the default values. To make changes, click the links on the left pane of the wizard. For more information about the links, see the Help.

7 Click **Finish**.

8 If you want the rule to begin creating deployment tasks, ensure that the rule is enabled. For more information, see Section 9.5.11, "Enabling and Disabling Deployment Rules," on page 142.

9.5.2 Configuring Installation Packages

In the Deployment Rule Wizard, click **Packages** to configure the installation packages you want to install.

To configure installation packages:

1 (Conditional) If you want to install certain packages on the agent computer, select the packages you want to install.

2 (Conditional) If you want to uninstall certain packages from the agent computer, select the packages you want to uninstall and then select the **Uninstall the selected packages from the target computers** option.

3 (Conditional) If the AppManager agent is not already installed on the agent computer, include the **NetIQ AppManager Windows Agent** installation package and the application-related installation packages.

4 (Optional) If you want the installer to skip the pre-installation check for packages on the agent computer, select the **Skip > pre-deployment > check** option.

5 (Conditional) If the **Requires Configuration** column is set to **Yes**, configure the selected package. To configure the selected package, click the configuration link in the list. The Configuration Wizard prompts you for the required configuration information. For more information about configuring the agent installation package, see the *Installation Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

NOTE

⬧ If the configuration of a deployment package requires you to specify a management server, use the NetBIOS name of the management server. You cannot use the IP address.

⬧ Before you install the AppManager agent, consider the permissions required to run Knowledge Scripts on the agent computer. For information about the permissions required to run a particular Knowledge Script, see the Help for that script.

9.5.3 Deployment Credentials

In the Deployment Rule Wizard, click **Credentials** to specify the user account you want to use to run the installation package on the agent computer.

The credentials you provide to run the installation package are different than the credentials used by the AppManager agent to run particular Knowledge Scripts.

To run an installation package on an agent computer, for example, to install the AppManager agent, provide a domain user account that is member of the **Administrators** group on the agent computer. When configuring a rule, you can use the Windows user account that is currently configured as the default for deployment, or provide different logon information.

Select an option to specify the deployment credentials you want to use:

- **Use configured default authentication credentials.** The Control Center console can be configured to use default authentication credentials for all deployment rules. Click **Configure** to configure the default credentials. For more information, see Section 3.2.1, "Console Options," on page 41.

- **Use different authentication credentials.** You can provide a different set of credentials for this rule in place of the default credentials.

If you specify invalid credentials, the Control Center console does not raise a warning until **after** you enable the deployment rule. In the **Deployment Rule** view, the **Status Details** column displays information about any problems with the configuration of the rule.

9.5.4 Agent Computers

In the Deployment Rule Wizard, click **Target Computers** to choose a method for selecting computers. After you choose a selection method, use the **Inclusion Filters** and **Exclusion Filters** tabs to configure filters to further refine the computers you want to include and exclude, respectively.

To generate a deployment task for a computer, the computer must match both the target selection method and the inclusion filter. If you specify an exclusion filter, the computer must not be selected by the exclusion filter.

TIP: If the package you want the rule to install is already installed on the agent computer, a deployment task is not generated.

Select an option to choose a selection method:

- **Agent.** Selects AppManager 7.0 (or later) agents currently managed by Control Center. This option is useful for installing updates to the AppManager agent.

- **Domain.** Searches for computers in the Active Directory domain you specify. In the **Domain Name** field, type the fully-qualified domain name or NETBIOS name of the domain you want.

- **Specific Computers.** Click **Add** to add a computer name. You can specify the names or IP addresses of the computers you want. At this time, only IPv4 addresses are supported.

  Click **Remove** to remove a computer from the list.

- **Computers List File.** Specify the path and name of a file on the deployment server that contains an XML description of the computers you want, for example:

```
<ScannedMachines>

  <ScannedMachine Name="Machine1" Platform="Windows"/>

  <ScannedMachine Name="Machine2" Platform="Windows"/>

  .

  .

  .

</ScannedMachines>
```

  You can specify the names or IP addresses of the computers you want. At this time, only IPv4 addresses are supported.

◆ **Management Groups.** Click **Add** to add a Control Center management group. This option is useful when you want to install a package on the computers in a management group. Use this option to leverage the power and flexibility of rule-based management groups to select the computers you want. For more information, see Section 4.1, "About Management Groups," on page 53.

To remove a management group from the list, select the management group and click **Remove**.

9.5.5 Inclusion Filters

In the Deployment Rule Wizard, use the **Inclusion Filters** tab to configure the criteria you want to use to further refine the list of agent computers you want to include.

To generate a deployment task for a computer, the computer must match both the target selection method and the inclusion filter. If you specify an exclusion filter, the computer must not be selected by the exclusion filter.

Select a criteria you want and click the link to configure each criteria:

◆ **Domains.** Include computers in a domain by selecting the domains you want from the list.

◆ **IP Addresses.** Include computers by typing the IP address of each computer you want. At this time, only IPv4 addresses are supported.

◆ **IP Address Ranges.** Include computers that fall within a range of IP addresses by specifying the starting and ending IP address. At this time, only IPv4 addresses are supported.

◆ **Computer Names.** Include computers by typing the name of each computer you want or by browsing the list of computers.

◆ **Operating Systems.** Include computers that run a supported version of the Microsoft Windows operating system.

◆ **Organizational Units.** Include computers that belong to one or more organization units in Active Directory by specifying the distinguished name of the organizational unit, for example:
`OU=Domain Controllers,DC=corporate,DC=local`

◆ **Regular Expressions.** Include computers that match all specified regular expressions, for example, **EXCHSERV0[1-5]** only selects computers named **EXCHSERV01**, **EXCHSERV02**, and so on, up to **EXCHSERV05**.

◆ **Wildcard Expressions.** Include computers that match one or more specified wildcard expressions. Supported expressions include **?** (one character) and * (more than one character).

◆ **Remote Registry Values.** Include computers with one or more matching registry values. For more information, see the Help.

9.5.6 Exclusion Filters

To exclude a computer that matches the target selection method and the inclusion filter, specify an exclusion filter on the **Exclusion Filters** tab. This tab uses the same filters as the **Inclusion Filters** tab. If the computer is not selected by both the target selection method and the inclusion filter, you do not need to specify an exclusion filter.

NOTE: To generate a deployment task for a computer, the computer must match both the target selection method and the inclusion filter, if one was specified. The Deployment Service does not generate a deployment task for a computer that matches the exclusion filter.

9.5.7 Deployment Service

In the Deployment Rule Wizard, click **Deployment Service** to select the Deployment Server you want to use to deploy the task. For example, you might have a Deployment Service that is used to deploy installation packages to a particular corporate office. For information about installing the Deployment Service, see the *Installation Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

9.5.8 Deployment Schedule

In the Deployment Rule Wizard, click **Deployment Schedule** to specify when to run the installation on the agent computers.

Depending on the packages you are installing, and where you are installing them, you might need to specify a custom schedule. For example, if you have a maintenance window on your Exchange servers, you can schedule a deployment package to be installed during that time.

NOTE: If an agent computer is in maintenance mode, a deployment task will run at its scheduled time. Enabling maintenance mode on an agent computer does not prevent the deployment task from running.

Select an option to specify the deployment schedule you want to use:

- **Use default deployment schedule.** The Control Center console can be configured to use a default deployment schedule. Click **Configure** next to the **Use default deployment schedule** option to configure the default schedule. For more information, see Section 3.2.3, "Deployment Options," on page 42.

- **Create custom deployment schedule.** Select this option to create a custom deployment schedule. You can choose to deploy tasks immediately upon approval or based upon a specified schedule, once the task is approved.

Option	What it does
Deploy upon approval	This option requires the deployment task to be approved before it runs.
Deploy according to the following schedule	Specifies when to deploy the task. Selecting this option changes the deployment task status to **Scheduled**. Scheduled tasks must also be approved.

For information about configuring a custom deployment schedule, see the Help.

9.5.9 Email Notification

In the Deployment Rule Wizard, use the **Notification** tab to specify who will be sent an email message about the status of a deployment task. You can choose to send email notification to a list of recipients when a deployment task succeeds or fails.

Configuring the Notification From Email Address

Before you can configure recipients of email notifications, you might want to configure the **Notification From** email address that Control Center uses to send the email notifications. Otherwise spam blockers might block the email notifications. The default value for the From address is empty.

To configure the Notification From email address:

1 Open the following file:

    <Install_folder>\AppManager\ControlCenter\bin\DeploymentService.exe.config.

2 Specify an email address to use as the Notification From email address in the `NotificationEmailFromAddress` parameter.

When the Deployment Service sends an email notification to the recipients, the email displays the value you specify in the `DeploymentService.exe.config` file as the From address.

3 Close and save your changes.

4 Restart the NetIQ AppManager Control Center Deployment Service for your changes to take effect.

Configuring the List of Notification Recipients

Select an option to specify the email notification option you want to use:

◆ **Use default list of notification recipients.** The Control Center console can be configured to use a default list of notification recipients. You can also change the default list of recipients. Click **Configure** to configure the recipient list. For more information, see .

◆ **Use custom list of notification recipients**. You can use a specific list of recipients rather than using the default list of recipients.

To add recipients to the custom list, type the email address of the recipient in the **New Email Recipient Address** text box and click **Add**.

To remove recipients from the custom list, select the recipient in the custom list and click **Remove**.

9.5.10 Viewing Summary Information for a Deployment Rule

In the Deployment Rule Wizard, use the **Summary** tab to view a summary of how the rule is configured.

When you create a new rule, you can also view the summary information in the **Rules** view. Click the rule for which you want to view summary information in the **Summary** pane.

Summary information for the default, out-of-the-box rules is not displayed until after you configure the rule.

9.5.11 Enabling and Disabling Deployment Rules

To allow the Deployment Service to generate a deployment task for each agent computer, enable the deployment rule.

NOTE: If Control Center generated deployment tasks before you made changes to or enabled the rule, delete the previous deployment tasks.

When you update a deployment rule, Control Center creates new deployment tasks for the deployment rule. However, Control Center does not update or remove any deployment tasks that it previously created for the deployment rule.

You cannot enable a rule until you have properly configured it. For example, you might need to configure an installation package before you can enable the rule.

After you enable a deployment rule, the associated Deployment Service will execute the rule at the next interval. By default, this interval is every 5 minutes. For information about the deployment interval, see Section 9.3, "Deployment Preferences," on page 135.

When you configure a rule to install one or more packages, the rule remains **Enabled**. The rule will generate new tasks only for those computers that match the rule requirements. However, if you configure a rule to uninstall one or more packages, the rule is automatically **Disabled** after the list of agent computers is evaluated.

You can also disable a deployment rule to prevent the rule from running or generating new deployment tasks. Note that matching deployment tasks are only generated if the tasks have not been previously generated or if they have been deleted while in **Waiting for Approval** state.

To enable a deployment rule:

1. Log in to Control Center with an account that has permissions to create, delete, and modify rules.
2. In the **Navigation** pane, click **Deployment**.
3. Click **Rules** to display a list of rules.
4. Select the rule you want and in the **Tasks** pane, click **Enable Deployment Rule**.

9.5.12 Refreshing the List of Deployment Rules

To view the rules that you create or modify you need to refresh the list of rules.

To refresh the list of rules, click **Refresh Current View** on the ribbon or press **F5**.

9.5.13 Viewing Rule Evaluation History

After you create and enable a deployment rule, AppManager evaluates the rule for processing and displays the status in the **Rule History** pane. To view the evaluation history, click **Rule History** in the **Deployment** view of the **Navigation** pane.

The **Rule Processing History** portion of the **Rule History** pane indicates whether processing was successful. If AppManager was not able to create a deployment task for the rule, it provides details about the error in the **Task Comment** column.

If the evaluation status changes the next time that AppManager evaluates the rule, AppManager updates the rule history. AppManager does not update the rule history if the evaluation status does not change.

To delete all or a portion of the rule history, click **Delete all rule history** or **Delete selected rule history** in the **Tasks** pane.

9.5.14 Changing the Properties of a Deployment Rule

Before you update a rule, consider whether you need to change the rule. If the change applies:

- ◆ To a few agent computers, you can edit the deployment tasks for those computers. For more information, see Section 9.6.7, "Configuring the Installation Package for Deployment Tasks," on page 147.
- ◆ To most of the agent computers, update the rule. You must delete existing deployment tasks created by the rule before new deployment tasks with the updated configuration are generated.

To change the properties of a rule:

1 Log in to Control Center with an account that has permissions to create, delete, and modify rules.

2 In the **Navigation** pane, click **Deployment**.

3 Click **Rules** to display a list of rules, and then click the rule you want to modify.

4 In the **Tasks** pane, click **Edit Deployment Rule** to start the Deployment Rule Wizard and edit the selected rule. For more information about fields on a window, see the Help.

5 After you edit the rule, enable the rule. For more information, see Section 9.5.11, "Enabling and Disabling Deployment Rules," on page 142.

9.5.15 Copying a Deployment Rule

As a convenience you can copy an existing rule and then modify it to meet your needs. After you copy a rule, its status is **Disabled**.

To copy a rule:

1 Log in to Control Center with an account that has permissions to create, delete, and modify rules.

2 In the **Navigation** pane, click **Deployment**.

3 Click **Rules** to display a list of rules, and then click the rule you want to copy.

4 In the Tasks pane, click **Make Deployment Rule Copy**.

The name of the copied rule is prefixed with **Copy of** *<rule name>* where *<rule name>* is the original rule name.

9.5.16 Deleting a Deployment Rule

After you delete a rule, you cannot restore it. If you want to temporarily prevent the rule from generating deployment tasks, consider disabling it. For more information, see Section 9.5.11, "Enabling and Disabling Deployment Rules," on page 142. When you delete a rule, you are prompted to confirm you that want to delete the rule.

You can disable the confirmation to delete a rule by configuring the Control Center options. For more information, see Section 3.2.3, "Deployment Options," on page 42.

To delete a rule:

1 Log in to Control Center with an account that has permissions to create, delete, and modify rules.

2 In the **Navigation** pane, click **Deployment**.

3 Click **Rules** to display a list of rules, and then click the rule you want to delete.

4 In the **Tasks** pane, click **Delete Deployment Rule** to delete the rule.

9.6 Managing Deployment Tasks

A **deployment task** indicates an installation package you want to install on a certain computer based on the criteria met in a deployment rule. When the agent computer meets the criteria for a deployment rule, the Deployment Service generates a deployment task.

Before you can deploy a module to an agent computer, the AppManager agent must be installed. For example, if you configure a rule to install the AppManager agent and module for Microsoft Internet Information Services (IIS), the Deployment Service does not generate the deployment task for IIS until after you deploy the AppManager agent on the agent computer.

TIP: If a package is already installed on the agent computer, the Deployment Service does not generate a deployment task. For example, if you configure a rule to install the AppManager agent and module for Microsoft SQL Server, if the same version of the agent is already installed on the agent computer, the Deployment Service only creates the deployment task to install AppManager for Microsoft SQL Server.

9.6.1 Viewing Deployment Tasks

To view a list of all deployment tasks, log in to the Control Center console with an account that has permissions to work with tasks and in the **Navigation** pane select the **Deployment** view. Then select **Tasks**. The **Deployment\Tasks** view displays a list of all deployment tasks. You can group by **Computer** to view all deployment tasks for an agent computer.

You can configure how to display this information. For more information, see Section 3.2.3, "Deployment Options," on page 42.

9.6.2 Approving Deployment Tasks

You must approve deployment tasks, including scheduled deployment tasks, before Control Center installs the installation package on the agent computer identified in a deployment task. For more information, see Section 9.5.8, "Deployment Schedule," on page 141.

To approve a deployment task:

1 Log in to Control Center with an account that has permissions to work with tasks.

2 In the **Navigation** pane, click **Deployment**.

3 Click **Tasks** to view a list of available deployment tasks.

4 Click the task you want to approve and in the **Tasks** tab, click **Approve Deployment Task**.

To select multiple tasks, press **Ctrl** or **Shift** while making your selections. The task will become **Active** or **Scheduled**. After the task is completed, its status will be updated to indicate whether it ran successfully. If you configured email notification, the task will notify the list of email recipients once it completes.

9.6.3 Rejecting Deployment Tasks

You cannot run a deployment task until you approve it. If you do not want to run a task that is **Waiting for Approval**, you can reject the task. When you reject a task, its status changes to **Rejected**.

After you reject a deployment task, you cannot approve or configure the task. To configure a task that is **Rejected**, delete the task and wait for the deployment rule to generate a new task.

If you update the deployment rule that generated the deployment task, the Deployment Service does not generate a new task until you delete the **Rejected** task. For more information about deleting a task, see Section 9.6.4, "Deleting Deployment Tasks," on page 146.

To reject a deployment task:

1 Log in to Control Center with an account that has permissions to work with tasks.

2 In the **Navigation** pane, click **Deployment.**

3 Click **Tasks** to view a list of available deployment tasks.

4 Click the task you want and in the **Tasks** tab click **Reject Deployment Task**.

9.6.4 Deleting Deployment Tasks

You can delete a task to remove it from the list. If you delete a task that is **Scheduled** or **Waiting for Approval,** and the rule that generated the task is enabled, the Deployment Service automatically generates a new deployment task. You cannot delete an **Active** task.

You can configure the Control Center console to automatically delete **Completed** deployment tasks and to hide tasks that are older than a specified period. For more information, see Section 3.2.3, "Deployment Options," on page 42.

When deleting a task, you are prompted to confirm that you want to delete the task. You can disable the confirmation to delete a task by configuring the Control Center options. For more information, see Section 3.2.3, "Deployment Options," on page 42.

To delete a deployment task:

1 Log in to Control Center with an account that has permissions to work with deployment tasks.

2 In the **Navigation** pane, click **Deployment**.

3 Click **Tasks** to view a list of available deployment tasks.

4 Click the task you want to delete.

To select multiple tasks, press **Ctrl** or **Shift** while making your selections. If the tasks in the list are grouped, for example, by Computer, first expand the group and then select the tasks. You cannot select the group to delete more than one task.

5 In the **Tasks** tab, click **Delete Deployment Task.**

9.6.5 Viewing the Deployment Rules for Deployment Tasks

You might want to view the deployment rule that generated a deployment task you want to run, reject, or delete.

To view information about the rule that generated a deployment task:

1 Log in to Control Center with an account that has permissions to work with deployment tasks.

2 In the **Navigation** pane, click **Deployment**.

3 Click **Tasks** to view a list of available deployment tasks.

4 Click the task you want and in the **Tasks** tab click **View Rule**.

9.6.6 Changing the Schedule for Deployment Tasks

You might want to configure a different schedule for a deployment task after the Deployment Service generates the task.

To change the schedule for a deployment task:

1 Log in to Control Center with an account that has permissions to work with deployment tasks.

2 In the **Navigation** pane, click **Deployment**.

3 Click **Tasks** to view a list of available deployment tasks.

4 Click the task you want and in the **Tasks** tab click **Schedule Deployment Task**.

5 In the Edit Task Schedule window, update the schedule. For more information, see the Help.

6 Click **OK**.

9.6.7 Configuring the Installation Package for Deployment Tasks

If the installation package for the deployment task requires configuration, you can configure the installation package either before or after you approve the task. For example, if you approve a task and it encounters a problem, you can configure the installation package and then re-activate the task. For more information, see Section 9.6.9, "Re-activating a Deployment Task," on page 148.

You cannot configure tasks that are **Running**, **Rejected**, or **Completed**.

To configure a deployment task:

1 Log in to Control Center with an account that has permissions to work with deployment tasks.

2 In the **Navigation** pane, click **Deployment**.

3 Click **Tasks** to view a list of available deployment tasks.

4 Click the task you want and in the **Tasks** tab click **Configure**. Note that only tasks that require configuration are selectable for configuration. For more information, see the Help.

9.6.8 Changing the Credentials for a Task

If a deployment task fails because of the credentials, you can change the credentials and re-activate the task. For more information, see Section 9.6.9, "Re-activating a Deployment Task," on page 148.

To change the credentials for a deployment task:

1 Log in to Control Center with an account that has permissions to work with deployment tasks.

2 In the **Navigation** pane, click **Deployment**.

3 Click **Tasks** to view a list of available deployment tasks.

4 Click the task you want and in the **Tasks** pane click **Change Credentials**.

5 In the Package Credentials window, update the credentials. For more information, see the Help.

6 Click **OK**.

9.6.9 Re-activating a Deployment Task

If a deployment task encounters an error, you can update the task properties and then re-activate the task to enable it to run on its specified schedule. For example, if the credentials for the deployment task are invalid, update the credentials for the task and then re-activate the task.

To re-activate a deployment task:

1 Log in to Control Center with an account that has permissions to work with deployment tasks.

2 In the **Navigation** pane, click **Deployment**.

3 Click **Tasks** to view a list of available deployment tasks.

4 Click the deployment task you want and in the **Tasks** tab click **Reactivate Deployment Task**.

   If you configured email notification, AppManager notifies the list of email recipients after the deployment task is completed.

9.6.10 Viewing the Properties of a Deployment Task

You can view the properties of a deployment task to check when it is scheduled to run or review any other information about the task.

To view the properties of a deployment task:

1 Log in to Control Center with an account that has permissions to work with deployment tasks.

2 In the **Navigation** pane, click **Deployment**.

3 Click **Tasks** to view a list of available deployment tasks.

4 Click the task you want and in the **Tasks** tab click **Deployment Task > Properties**.

   In the Deployment Task Properties window, information about the task is displayed. For more information, see the Help.

9.6.11 Refreshing the List of Deployment Tasks

When you create, enable, and approve a deployment rule, the Deployment Service generates a deployment task. If you want to view the deployment task that the Deployment Service generates, you need to refresh the **Tasks** view. To refresh the list of tasks, in the ribbon click **Refresh Current View** or press **F5**.

9.7 Viewing the Results

After the deployment tasks complete successfully, the Control Center console automatically updates the results. If you:

 Installed the AppManager agent, Control Center automatically displays the agent computer in the management groups with matching criteria. For more information, see Section 4.1, "About Management Groups," on page 53.

 Installed module support, the **Details** tab of the **Servers** view displays the discovered application resources. For more information, see Section 3.5, "Viewing Detailed Information about Discovered Resources," on page 46.

9.8 Viewing the Software Inventory

The Software Inventory list includes information about the packages that are installed, including a list of managed objects, or module support, and version information, for all agent computers. The list does not include information about AppManager 6.x (and earlier) agents.

To view software inventory information, in the **Navigation** pane, click **Deployment** and then click **Software Inventory**.

9.9 Troubleshooting Deployment Tasks

This section provides some tips for troubleshooting deployment tasks.

9.9.1 Agent Computer Not Appearing in the Servers View

When you install the AppManager agent, if the installation was successful but the agent computer does not appear in the **Servers** view of the **Master** management group, you might need to add the computer and re-run Discovery. For more information, see Section 3.4, "Adding a Computer to the Master View of a QDB," on page 44.

9.9.2 Deployment Tasks Not Being Generated After Enabling a Deployment Rule

- If you enable a rule but do not see the deployment tasks you expect, it might be due to one of the following reasons:
    - The corresponding tasks have already completed. Look for tasks in the **Tasks** view with a status of **Completed**.
    - The installation package is already installed on the agent computer. The Deployment Service does not generate a deployment task if the installation package is already installed on the agent computer. For example, if you configure a rule to install the AppManager agent and module for Microsoft SQL Server, if the same version of the agent is already installed, the Deployment Service generates only the deployment task for Microsoft SQL Server. Check the **Software Inventory** list of installed AppManager components on the agent computer.
    - There might be an error in the configuration of the installation package. Review the **Status Details** column for the deployment rule for any errors. For example, invalid credentials might be specified in the configuration.
    - No agent computers were selected. Review the rule to verify the agent computers are specified.
- When you finish configuring the deployment rule, the Deployment Service generates the deployment tasks, but the tasks are not deployed correctly because the agent computer does not support the packages you want to install.
- You need to wait until the deployment task installs the AppManager agent. The Deployment Service does not generate deployment tasks for modules until the agent installation is complete.
- If the agent computer does not have software inventory information for the AppManager agent, the AppManager agent cannot communicate with the deployment Web server computer or you specified the wrong the deployment Web server computer name when you configured the agent installation package.

9.9.3 Avoiding Recreating a Deployment Task After Deleting the Task

If you do not want a task to be recreated after you delete it, do one of the following:

 ◆ Disable the associated deployment rule that generates the deployment task.

 ◆ For those tasks you do not want to run that are associated with a deployment rule you do not want to disable, reject the task to change its status to **Rejected**.

9.9.4 Deployment Tasks Going Into Error State

Deployment tasks can enter an **Error** state for a variety of reasons, such as an incorrect installation package configuration. Review the status details for a task in an **Error** state by clicking the task and in the **Tasks** tab clicking **Deployment Task Properties**. The Deployment Task Properties window also includes information from the pre-installation check.

Another common source of errors with deployment tasks is providing incorrect credentials in the deployment rule. The Deployment Rule Wizard does not validate the credentials you specify. To validate the credentials for a deployment task, click the task and in the **Tasks** tab click **Change Credentials**.

9.9.5 Generating New Deployment Tasks

 ◆ To generate new deployment tasks after you update a rule, delete any existing deployment tasks. After you delete the existing tasks, the Deployment Service can generate new deployment tasks.

 ◆ If you uninstall the AppManager agent manually from the local computer, it can take up to 72 hours for the Deployment Service to create a new deployment task for the installation package. For more information about changing this interval, see Section 9.3, "Deployment Preferences," on page 135.

 ◆ If you encounter the following error when installing the AppManager Windows agent, view the `msierror.log` file on the agent computer to determine the errors:

   ```
   The Windows Installer encountered a fatal error, so the installation could not
   complete.
   ```

   Typically this error occurs when:

    ◆ The agent computer does not have enough free disk space for Control Center to install the AppManager agent. The AppManager agent requires approximately 30 MB of free disk space.

    ◆ The specified installation path does not exist. If you specify an invalid installation path, Control Center installs the agent in the default location,
      `C:\Program Files\NetIQ\AppManager`.

The `msierror.log` file is typically located in `%SystemRoot%\temp\WmiRel`. To access this file on a remote computer, you must have Administrator privileges to
`\\MACHINENAME\ADMIN$\Temp\WmiRel\msierror.log`.

10 Running Reports

This chapter describes how to use the standard reports provided with AppManager.

10.1 Understanding AppManager Reporting

AppManager reports include data in tabular and graphical format from a QDB. Examples of the types of AppManager reports include:

- **Performance-related reports** to help you understand the performance of your applications and systems.
- **Trend analysis reports** to help you plan for system or application upgrades.
- **Service level agreement reports** to quantify the availability of services and service response time.
- **Event summary reports** to provide an overview of the events being created by AppManager so that you can develop a plan of action.
- **Configuration reports,** to provide an overview of the system details being monitored and configuration details, such as event threshold settings.

AppManager reports use the same Knowledge Script-based mechanism that you use to run monitoring jobs that collect data and raise events.

An AppManager report queries data from a single QDB. At this time you cannot report on data from more than one QDB *in the same report*. To report on AppManager data from more than one QDB in the same report, use AppManager Analysis Center. For more information, see the Analysis Center product documentation.

AppManager reports include application-specific reports and generic reports. For more information, see the following sections.

10.1.1 About Application-specific Reports

AppManager reports are included with some Knowledge Script categories to provide **application-specific reporting**. Application-specific reports are included with the corresponding application Knowledge Scripts and are prefixed by Report_. For example, in the **Knowledge Scripts** view of the Control Center console, the Report_CPULoad report appears in the NT Knowledge Script category.

Application-specific reports only use data collected by a particular Knowledge Script. To extend the previous example, the Report_CPULoad Report Script only reports on data collected by the NT_CPULoaded Knowledge Script.

To determine the data source for a report, double-click the Report Script you want and in the **Values** tab of the Properties window, click **Help** to display the Help.

Application-specific reports only include data from standard out-of-the-box Knowledge Scripts and do *not* include data from copies of Knowledge Scripts. For information about reporting on data collected by a copy of an AppManager Knowledge Script, see .

10.1.2 About Generic Reports

Generic reports are located in the ReportAM category of the **Knowledge Scripts** view.

Generic reports enable you report on a variety of areas, including:

- **Performance information** Includes reports on average hourly and daily values, most recent values, or values by computer or data stream.

- **Job and event information** Includes reports that list events by computer, monitored application, Knowledge Script, and event status and severity.

- **Application information** Includes reports on your AppManager installation and discovered applications on your managed computers.

- **Hardware information** Includes reports on some of the hardware components on your computers, provides an overview of network interfaces (including the manufacturer, IP address, and subnet mask), logical and physical disks (including used and free space), and printers (including port and driver information).

  AppManager Control Center also includes **best practice** reports. These reports are recommended for all customers and are located in the StartPoint category. For complete information about the StartPoint and ReportAM AppManager reports, including information about the data source for a report, see the Help.

10.1.3 Understanding How AppManager Reports Work

An AppManager report consists of a Report Script that runs on the AppManager report agent. The **report agent** is an extension of the AppManager agent and is responsible for:

- Querying a QDB.

- Generating the HTML-based report.

To run a report, the AppManager report agent must be:

- Installed and discovered. Note that the report agent is not automatically discovered during installation. Use the Discovery_ReportAgent Knowledge Script to discover the report agent.

- Configured to have a QDB as a data source. By default, the AppManager report agent installation process configures a single QDB as the data source for the report agent. After installation, you can configure the report agent to report on data in additional QDBs, however, this is **not** recommended.

NetIQ Corporation recommends that you use a separate report agent for each QDB data source, and install each report agent on a dedicated server. For more information, see the *Installation Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2). Here are some considerations:

- Although you can configure a report agent to have more than one data source, at this time, an AppManager report can only include data from a particular QDB. An AppManager report cannot include data from more than one QDB.

- Depending on the amount of information in each report and the number and frequency of reports you run, the report agent can consume significant system resources.

10.1.4 Discovering the Appmanager Report Agent

When installing the AppManager Windows agent, the AppManager setup program includes an option to install the report agent. The AppManager setup program does not automatically install the report agent.

After you install the report agent, discover it by running the Discovery_ReportAgent Knowledge Script. If you have not discovered a report agent, the **Knowledge Scripts** view of the Control Center console does not display the ReportAM category or application-specific Report Scripts.

For information about installing and discovering the AppManager report agent, see the *Installation Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

Rediscover the report agent to:

 * Update the list of Knowledge Script categories upon which you want to report. For example, if you discover the report agent and then discover a new Knowledge Script category, such as IIS, rediscover the report agent to run IIS reports. For more information, see Section 10.1.3, "Understanding How AppManager Reports Work," on page 152.

 * Update the QDB data source for a report agent. At this time, it is not recommended that a report agent be configured with more than QDB data source.

To view the details for an AppManager report agent:

1 In the Control Center console, expand the **Enterprise Layout** view of the **Navigation** pane to select a management group that contains the report agent computer.

2 In a **Servers** view, click the report agent computer.

3 In the **Details** tab, expand the list of details to view report agent information.

   In particular, the details include the QDB with which the report agent communicates. If you are not sure which report agent you need, use the details to find the correct report agent.

4 Under **Report Agent > AM Repositories**, the QDB data source is displayed with a list of available Knowledge Script categories on which you can report.

10.2 Managing Data Collected for Reports

You can manage data collected for reports by configuring jobs to not collect detail data for reports. Not all reports use detail data and collecting this information increases the size of the QDB.

At this time, configure the QDB to collect detail data for reports. If you configure the repository preference to not collect detail data reports, you cannot override this preference in the properties of a running job. Instead, configure the **Advanced** tab job properties for the jobs that you do not want to collect detail data for reports.

10.3 Running a Report

When you use the Control Center console to run a report, ensure the following points:

 * Create a management group that only includes your report agents.

 * Run reports on an ad hoc basis. Do **not** run reports as part of a monitoring policy. You should not run the same report on more than one report agent as part of a monitoring policy.

◆ Configure each report agent to query a single QDB data source. AppManager reports can only include data from a single QDB.

◆ The list of discovered Knowledge Script categories under the QDB data source is up-to-date. If necessary, rediscover the report agent to update the list of Knowledge Script categories upon which you can report.

◆ The data you want to report on is in the QDB. If you want to report on data collected by a new Knowledge Script job, make sure the job runs at least once and is configured to collect data. Not all Knowledge Scripts are configured by default to collect data.

◆ At a minimum, configure a report to identify the data streams and time frame you want. Most reports do **not** have a default configuration that will generate a report.

◆ (Conditional) If you have more than one report agent, configure the report to select the report agent with the correct QDB data source. If you attempt to run a Report Script on more than one report agent, the report runs on the first selected report agent in the **Objects** tab of the job Properties window.

◆ Run reports on the report agent from a management group that you have configured to select the **Master** view or the **Report** view. If you run a report from a management group that is not configured with the **Master** or **Report** view as one of its members, you cannot see the reports in the list of available Knowledge Scripts.

10.3.1 Starting a Report

Starting a report is similar to starting a Knowledge Script job. For more information, see Section 5.2, "Starting a New Job," on page 69.

To start a report:

1 In the Control Center console, expand the **Enterprise Layout** view of the **Navigation** pane to select a management group that contains the report agent you want.

  Make sure the report agent is configured to report on the QDB where you want to run the report. Use the **Details** tab of the **Server Information** pane to verify the QDB with which the report agent communicates.

2 In the **Knowledge Scripts** view of the management group, click to expand the categories and browse reports.

3 To start a report, right-click a Report Script and click **Create a new job**.

To create...	Do this...
An application-specific report	Expand a Knowledge Script category and select a report from the list. Reports are prefixed with Report_.
A generic report	Expand the ReportAM category and select a report from the list.

4 In the Server Selection window, select the report agent computer and click **OK**.

5 In the Properties window, select a tab to configure its properties:

Click...	To...
Schedule	Configure the default schedule for running the report. For more information, see Section 10.3.2, "Setting the Schedule," on page 155.

Click...	To...
Values	Configure the parameters of the report. For information about the report, including the data source for the report and information about the configuring the parameters in the Values tab, click **Help** to display the Help. In most cases, you **cannot** use the default values. For more information, see Section 10.3.3, "Setting the Report Values," on page 156.
Actions	Configure an action to run, for example, after the report completes successfully. You can use the Action_SMTPMailRpt Knowledge Script to email the first page of a report. The first page contains hyperlinks to the other pages in the report. To run this action, first configure the **URL mapping** parameter on the report agent.For information about the Action_SMTPMailRpt Knowledge Script, see the Help.
Objects	Configure the report agent to run the report. For more information, see Section 10.3.4, "Selecting a Report Agent to Run the Report," on page 156.
Advanced	Configure advanced properties. Advanced properties are not applicable to Report Scripts.

10.3.2 Setting the Schedule

Use the **Schedule** tab to set the schedule for your report.

When scheduling reports, try not to overload the report agent. The resources necessary to generate a report increase as the amount of data in each report increases, and as the complexity and number of charts increases.

Reports should be scheduled for a time of day when network traffic is lighter and when your QDBs are less busy, in order to speed the transfer of data and to maximize the SQL Server resources available to your Report Scripts.

Beyond choosing off hours, it is best to schedule small batches of reports to run consecutively (rather than simultaneously) with some amount of time between them. The time and system resources required to generate a report are directly proportional to the amount of data in that report.

By default, the report agent is configured to process up to three reports at a time. You can use the AMAdmin_ConcurrentRpt Knowledge Script to change the number of reports concurrently processed by the agent. Keep in mind that even if you are running the agent on an especially robust computer, it is not recommended to ever configure the agent to process more than five reports at a time.

Under circumstances where you are using the default configuration of the agent, you could, for example, schedule the first three reports to run at 2:00 A.M., the next three to run at 2:05 A.M., and so on. You will need to monitor how long it is taking for each report to complete so you can adjust the schedules accordingly.

To set the schedule for a report:

1 Display the Properties window for the Report Script job.

If the report is already running, double-click the job identifier in the **Jobs** tab of the **Server Information** pane; or select the job, right-click, and then click **Properties**.

2 Click the **Schedule** tab and set the report schedule.

For more information about the scheduling options, see Section 5.3, "Setting the Schedule," on page 70 or click the **Help** button to display the Help.

3 Click another tab to change its properties or click **OK** to start the job.

10.3.3 Setting the Report Values

Reports have default parameters that control how they select and display data. Typically, configure the Values tab to specify the data source and time frame.

If you do **not** select a data source and configure a time frame, your report will raise an event indicating there was "No data for report."

For reports that display data in graphical format, you might also need to select a chart type. For information about selecting a chart type, see the Report Script Help.

You can configure a report with properties that can be sorted, including Author, Component, Company, and Custom Fields. For more information, see Section 10.4.1, "Tips for Configuring Reports," on page 157.

To set the report values:

1 Display the Properties dialog box for the Report Script job.

If the report is already running, double-click the job identifier in the **Jobs** tab of the **Server Information** pane; or select the job, right-click, and then click **Properties**.

Click the **Values** tab to change the default properties of the report.

For...	Do this...
Data source	Select the data streams and time frame you want. Some reports allow you to configure the report style which determines how the first page of the report is organized, for example, by computer or data stream, and the aggregation interval. To reduce the size of the report, limit the amount of data returned by reporting on a particular data stream for a group of computers.
Report Settings	Specify the following: ◆ Chart style, if applicable. For more information, see the Report Script Help. ◆ A sub-folder where you want generate the report. The sub-folder is relative to the report agent's base output path and is useful for limiting report access.
Event Notification	Raise events associated with generating the report, and to set severity levels for those events.

2 For complete information about configuring a report, click the **Help** button to display the Help.

3 Click another tab to change its properties or click **OK** to start the job.

10.3.4 Selecting a Report Agent to Run the Report

When you run a Report Script, the **Objects** tab of the Job Properties window displays a list of available report agents and data sources.

◆ Make sure each report agent is configured to query a single QDB data source. AppManager reports can only include data from a single QDB.

◆ Make sure the list of discovered Knowledge Script categories under the QDB data source is up-to-date. If necessary, rediscover the report agent to update the list of Knowledge Script categories upon which you can report. For more information, see Section 10.1.4, "Discovering the Appmanager Report Agent," on page 153.

- Make sure that the data you want to report upon is in the QDB. If you want to report on data collected by a new Knowledge Script job, make sure the job runs at least once and is configured to collect data. Not all Knowledge Scripts are configured by default to collect data.

- If you have more than one report agent, be sure to configure the report to select the report agent with the correct QDB data source. If you attempt to run a Report Script on more than one report agent, the report runs on the first selected report agent in the **Objects** tab of the Job Properties window.

To select the report agent to run the report:

1 Display the Properties window for the Report Script job.

 If the report is already running, double-click the job identifier in the **Jobs** tab of the **Server Information** pane; or select the job, right-click, and then click **Properties**.

2 In the **Objects** tab of the Properties window, select the QDB data source on the report agent you want to run the report.

 Note that the data source information for the report agent information is displayed using the following syntax:

 AM Repositories:*server:database:category*

 where:

 - *server* is the name of the server that hosts the QDB

 - *database* is the name of the QDB

 - *category* is the name of the Knowledge Script category the report agent uses to query the repository

3 Click another tab to change its properties or click **OK** to start the job.

10.4 Performing Additional Tasks

After you run a few reports, you might find you need to perform additional tasks:

- Section 10.4.1, "Tips for Configuring Reports," on page 157
- Section 10.4.2, "Configuring Where the Report Agent Generates Reports," on page 158
- Section 10.4.5, "Maintaining the Size of the Report Folder," on page 160

10.4.1 Tips for Configuring Reports

Before you decide on the final configuration of a report, experiment with some different configurations. Run the report with different parameter settings and study the results. Does it include all the data you want? Is there extraneous data? Are the charts readable? Is the result appropriate for the audience?

When you are preparing a report, keep in mind the following:

- Limit the scope of the report. Rather than creating one report for all the Exchange Servers in your enterprise, create separate reports by location or department. If you want to report on physical memory usage, select only that data stream, rather than using all data returned by the NT_MemUtil Knowledge Script. By limiting the scope, you reduce the time it takes to generate a report, reduce the report size, and make the report easy to understand.

- Use an appropriate time range. To prepare an ad hoc report, a specific time range is probably appropriate. To prepare a report to run on a regular basis, use a sliding time range. Be sure the time range coincides with the times during which data was collected.

- Be sure you can see all data streams in the chart. If an area or bar chart obscures some of the data streams, use a ribbon or line chart instead. If the rotation of the chart does not facilitate easily reading the data, use a different rotation. If there are too many data streams illustrated in a chart, use a different style for the report (*by computer* as opposed to *all data streams on one page*), or further limit the scope of the report.

    Use the Chart Console to see the relative values of data streams before you configure the charts in your reports. Once you understand those relationships, you can better determine how to present data in a report. For example, if there is a great deal of disparity in value between data streams, a ribbon or line chart would be a better choice than an area or bar chart for letting you see all the data. For more information, see the Report Script Help.

10.4.2 Configuring Where the Report Agent Generates Reports

The default output path of the report agent is configured during installation to:

`<InstallDir>:\NetIQ\Common\Report`

Use the AMAdmin_SetReportPaths Knowledge Script to configure the report agent computer to use a different output path. For more information, see the Help.

The report agent output path is used as a base path for generating reports. To limit access to reports, you can configure each report to generate into a sub-folder under the report agent base path and configure permissions on each sub-folder.

10.4.3 Reporting on Data Collected by a Copy of a Knowledge Script

Application-specific reports are hard-coded to generate reports based on data collected by a particular Knowledge Script. For example, the NT_Report_CPULoad Report Script only includes data collected by the NT_CpuLoaded Knowledge Script.

If you use the NT_Report_CPULoad Report Script to display data collected by a copy of the NT_CPULoaded Knowledge Script, for example, NT_MYCPULoaded, the Report Script raises an event to indicate "No data for report."

To resolve this problem, you will need to:

- Create a copy of the Report Script.
- Edit the Report Script to include data collected by the new Knowledge Script instead of the original, out-of-the-box Knowledge Script.

To copy and check out a Report Script:

1 In the Control Center console, right-click the report you want and click **Copy KS**.

2 In the Copy Knowledge Script window, create a new name for the report.

3 Click **OK**.

4 Right-click the copy of the Knowledge Script and click **Check Out KS**.

5 In the Check Out Knowledge Script window, select a folder location to save the Report Script.

6 Click **Save**. The Developer Console displays the editable script. You are now ready to edit the Report Script.

To edit the Report Script to use a different data source and check the script into the repository:

1 In the Developer Console, click **View > Properties**.

2 In the Properties window, click **Parameters**.

3 In the **Parameters** tab, click **PRM_KSList** from the list of parameters and click **Modify**.

4 In the Modify Parameter window, edit the **Default Value** field to specify the name of the Knowledge Script that the report will use to collect data.

5 Click **Save**.

6 Click **OK** to close the Script Properties window.

7 Save your changes to local disk by clicking **File > Save**.

8 To check your changes into the QDB, click **Tools** > **Check In Knowledge Script**.

10.4.4 Configuring Folder Permissions

In addition to limiting report access to authenticated users, you can restrict report access by creating sub-folders in the `Report` folder, setting permissions for those folders based on the intended audience, and then generating reports to the appropriate folders.

For example, create a report sub-folder for account managers and a sub-folder for IT personnel, and set permissions on those folders that restrict access based on group membership.

When a user from one group connects to `AMReports` via the URL, only the reports available to that account are visible. See your Windows documentation for information about managing accounts and setting folder permissions.

Generating a Report to a Sub-folder

After you have configured report sub-folders with permissions, configure the output folder location for each report to specify the sub-folder you want.

To write reports to a specific folder:

1 Start a Report Script job.

2 Click the **Values** tab of the Properties window.

3 In the **Select output folder** parameter, click **Browse [...]**.

4 Select **Specific name of folder**.

5 In the **Folder** field, enter the path to the output folder.

The Report Script name (for example, AvgValueByMin) is the default folder name for the report, and is added to the base output path of the report agent. If you want the report available only to IT personnel, include the IT folder in the name. For example, IT\AvgValueByMin.

IT\AvgValueByMin is appended to the base output path of the report agent, and the report is generated in that folder.

10.4.5 Maintaining the Size of the Report Folder

The size of the `Report` folder depends upon number of reports you generate and how long you keep reports.

The size of a report increases as you include more:

- Data points
- Charts

A report can range from a few kilobytes to several megabytes.

You can monitor and maintain the size of the `Report` folder by deleting expired reports. For more information, see the following topics:

10.4.6 Collecting Information about the Report Folder Disk

The size of the `Report` folder can be monitored indirectly by monitoring the space on the disk where the `Report` folder is located.

Use the NT_LogicalDiskSpace Knowledge Script to monitor the disk and collect data about the percentage of space that is in use and the amount of space (in MB) that is available.

10.4.7 Reporting on the Used and Available Disk Space for Reports

After you have data about the available disk where the `Report` folder is located, you can then generate periodic reports about the available and used space on that disk.

For example, if your Report folder is on `C:\Program Files\NetIQ Corporation\AppManager\Web\Report`, the following data streams are collected:

- `Ldsk: C:AVAIL MB`
- `Ldsk: C:USED %`

You can then use a report such as ReportAM_AvgValueByDay to generate periodic reports about the values of those two data streams.

10.4.8 Deleting Expired Reports

The **Output Settings** for a report enables you to configure an expiration date. With an expiration date specified, you can run the AMAdmin_DeleteExpiredReports Knowledge Script to periodically scan the `Report` folder and delete expired reports. For more information about the AMAdmin_DeleteExpiredReports Knowledge Script, see the Help.

10.4.9 Exporting Report Content

By modifying a Report Script, you can add the option to export the contents of a report to an XML file or to a TXT file in which the report data are represented in a comma-separated format.

Two modifications to the report script are necessary. You must:

- Add four new parameters to the XML section of the script.
- Add code to the script.

NOTE: These modifications are possible only for Report Scripts that use `NETIQREPORT.dll`, referenced in the script as `NETIQREPORT.ReportObject.`

When you export the contents of a report, the XML or TXT file is output with the same name and to the same directory as the report.

This example uses the ReportAM_AvgValueByHr script. The modified script is renamed ReportAM_AvgValueByHrEXPORT.

This section discusses the following topics:

- Section 10.4.10, "Adding the New Parameters," on page 161
- Section 10.4.11, "Adding the Additional Block of Code," on page 163

10.4.10 Adding the New Parameters

The four new parameters include a group folder, **Export settings**, and the following three parameters:

- **Select output type**, which has the following options:
  - Report (create standard report)
  - Export (export content, do not create standard report)
  - Both (create standard report and export content)

- **Select export format**, which has the following options:
  - XML (exports to an XML file)
  - CSV (exports to TXT file)

- **Export file column delimiter, leave blank for TAB**, which lets you enter the column delimiter you want to use for data. This parameter is valid only if you select CSV as the value for the previous parameter.

The following procedure assumes a familiarity with copying Knowledge Scripts, checking them in and out of the QDB, and editing different portions of a script. For more information about copying scripts and checking them in and out of the QDB, see the *Operator Console User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

To add the new group of parameters in the appropriate place:

1 Make a copy of ReportAM_AvgValueByHr, rename it ReportAM_AvgValueByHrEXPORT, and check it out of the QDB.

2 Use Notepad to open `ReportAM_AvgValueByHrEXPORT.qml.`

3 Scroll through the XML portion at the beginning of the script until you see the following parameter definition:

```
<Param name="ReportSetting">
    <Desc>Report settings</Desc>
    <Type>String</Type>
    <Size>128</Size>
    <ReqInput>0</ReqInput>
    <Folder>2</Folder>
    <NoQuote>0</NoQuote>
</Param>
```

4 Enter the following parameter descriptions into the script so they immediately precede `<Param name="ReportSetting">`.

```
- folder for export settings
  <Param name="Export">
    <Desc>Export settings</Desc>
    <Type>String</Type>
    <Size>10</Size>
    <ReqInput>0</ReqInput>
    <Folder>1</Folder>
    <NoQuote>0</NoQuote>
  </Param>
 - to specify export, report, or both
  <Param name="PRM_OUTPUT">
    <Desc>Select output type</Desc>
    <Delim>,</Delim>
    <Type>String</Type>
    <Size>10</Size>
    <Value>Report</Value>
    <ReqInput>0</ReqInput>
    <I_Type>I_COMBOBOX("Report","Export","Both")</I_Type>
    <Folder>0</Folder>
    <NoQuote>0</NoQuote>
  </Param>
   - export type, i.e. XML or CSV
  <Param name="PRM_EXPTYPE">
    <Desc>Select export format</Desc>
    <Delim>,</Delim>
    <Type>String</Type>
    <Size>10</Size>
    <ReqInput>0</ReqInput>
    <I_Type>I_COMBOBOX("XML","CSV")</I_Type>
    <Parent>Export</Parent>
    <Folder>0</Folder>
    <NoQuote>0</NoQuote>
  </Param>
   - if CSV, column delimiter
  <Param name="PRM_EXPDELIM">
    <Desc>Export file column delimiter, leave blank for TAB</Desc>
    <Type>String</Type>
    <Size>1</Size>
    <ReqInput>0</ReqInput>
    <Parent>Export</Parent>
    <Folder>0</Folder>
    <NoQuote>0</NoQuote>
  </Param>
```

5 Save the script, and keep it open in Notepad so you can make the necessary additions to the code.

10.4.11   Adding the Additional Block of Code

In order to implement the values for the parameters you have added to the script, you have to add code to wrap the call in the script that generates the report, and within that section you have to repeat the call that generates the script.

The call to generate the script can contain any of the following:

- ◆ .MakeDrillDownReportV1
- ◆ .MakeDrillDownReportV2
- ◆ .MakeDrillDownReportV3
- ◆ .MakeChart
- ◆ .MakeTable
- ◆ .MakeChartAndTable

To insert the additional code:

1 Scroll through the XML portion of the script to the `Sub Main ( )` routine, and continue until you see the following block of code:

```
With ReportObj
.CHILDSUBTITLE2 = subTitle2
Select Case PRM_DISPLAYTYPE
Case "By computer"
.ChildColumnFormat(1,300) ="%.2f"
Case "By data stream"
.ColumnFormat(1,3) ="%.2f"
.ChildColumnFormat(1,300) ="%.2f"
Case "By computer and data stream"
.ColumnFormat(1,3) ="%.2f"
.ChildColumnFormat(1) ="%.2f"
Case "All data streams on one page"
.ColumnFormat(1,300) ="%.2f"
Case "By Knowledge Script"
.ColumnFormat(1,3) ="%.2f"
.ChildColumnFormat(1,300) ="%.2f"
End Select
If (PRM_DISPLAYTYPE <> "All data streams on one page") Then
.Filter = StatsFilterObj
bHasData = .MakeDrillDownReportV1
(DataSourceObj.RecordSet, IncludeType)
Else
StatsFilterObj.Recordset = DataSourceObj.Recordset
bHasData = .MakeChartAndTable
(StatsFilterObj.Recordset, IncludeType)
End If
End With
```

2 Insert the new code just after `End Select` (the new code and the repeated call to generate the report are in bold type):

```
With ReportObj
.CHILDSUBTITLE2 = subTitle2
Select Case PRM_DISPLAYTYPE
Case "By computer"
.ChildColumnFormat(1,300) ="%.2f"
Case "By data stream"
.ColumnFormat(1,3) ="%.2f"
.ChildColumnFormat(1,300) ="%.2f"
Case "By computer and data stream"
.ColumnFormat(1,3) ="%.2f"
.ChildColumnFormat(1) ="%.2f"
Case "All data streams on one page"
.ColumnFormat(1,300) ="%.2f"
Case "By Knowledge Script"
.ColumnFormat(1,3) ="%.2f"
.ChildColumnFormat(1,300) ="%.2f"
End Select
'beginning of new code
If (PRM_Output = "Export") Then
bHasData = .Export(ADORecObj, PRM_EXPTYPE, PRM_EXPDELIM)
Elseif (PRM_OUTPUT = "Report") Then
'beginning of call to make report
If (PRM_DISPLAYTYPE <> "All data streams on one page") Then
.Filter = StatsFilterObj
bHasData = .MakeDrillDownReportV1
(DataSourceObj.RecordSet, IncludeType)
Else
StatsFilterObj.Recordset = DataSourceObj.Recordset
bHasData = .MakeChartAndTable
(StatsFilterObj.Recordset, IncludeType)
End If
Else
bHasData = .Export(ADORecObj, PRM_EXPTYPE, PRM_EXPDELIM)
Set ADORecObj = ADOCmdObj.Execute
'repeat of call to generate report
If (PRM_DISPLAYTYPE <> "All data streams on one page") Then
.Filter = StatsFilterObj
bHasData = .MakeDrillDownReportV1
(DataSourceObj.RecordSet, IncludeType)
Else
StatsFilterObj.Recordset = DataSourceObj.Recordset
bHasData = .MakeChartAndTable
(StatsFilterObj.Recordset, IncludeType)
End If
End If
End With
```

3 Save the script and check it in to the QDB.

11 Collecting and Analyzing Data in Control Center

This chapter describes how to collect and view data from the QDBs you manage with Control Center.

11.1 Collecting Data

AppManager allows you to collect data from the QDBs in your Control Center environment and generate charts and reports for data analysis.

You can use most Knowledge Scripts to collect data from QDBs and generate charts. You can view charts on the **Charts** tab in Control Center or in the AppManager Chart Console. The **Charts** tab provides quick access to collected data directly from Control Center. The tab is available in the **Server Information** and **Service Map Object Information** panes. For more information about the **Charts** tab, see Section 11.3, "Working with the Charts Tab," on page 166. The Chart Console runs as a standalone application. For more information about the Chart Console, see Section 11.4, "Working with the Chart Console," on page 170.

You can use some Knowledge Scripts to generate application-specific and generic reports based on data from a single QDB. For more information about generating basic AppManager reports, see Chapter 10, "Running Reports," on page 151. You can use NetIQ AppManager Analysis Center to generate reports that include data from more than one QDB. For more information about using Analysis Center, see the Analysis Center product documentation.

To configure a Knowledge Script job to collect data and generate charts, enable the data collection parameter on the **Values** tab of the Job Properties window. Each time the job runs, it collects time and value pairs called **data points**. A series of data points is a **data stream**.

If a job collects more than one kind of information, it creates several data streams. For example, the NT_CpuResource Knowledge Script collects four data points (User CPU, Number of processes, All threads, and Interrupts for all CPU), which appear as separate data streams in charts.

The QDB stores each data stream a Knowledge Script job generates so you can view data later. A chart is a view of one or more data streams.

As a job runs, AppManager constantly appends new data to the data stream. By default, AppManager keeps the data points that jobs collect for a given day in a data table for immediate display in real-time charts and graphs for 30 days, and each day removes any data tables that are more than 30 days old. For information about changing the default setting, see Section 3.2.8, "Global Preferences Options," on page 44.

11.2 Analyzing Data

When deciding whether to generate charts or reports and which tools to use for data analysis, consider how you intend to use the data. You can use the data you collect from QDBs to accomplish the following objectives:

- Troubleshoot specific problems

The **Charts** tab is the best option for analyzing data you collect to diagnose a specific occurrence of a problem. For example, a virtual machine is experiencing memory problems and you want to determine which processes are using the most memory. Run the NT_TopMemoryProcs Knowledge Script with a high resolution (for example, every five minutes) and configure the job to collect data. The job generates a data stream for memory utilization, and data collection returns the process name, ID, and memory utilization for each data point. The **Charts** tab provides quick access to a graphical representation of the data stream and allows you to view detailed information about data points of interest.

For more information about using the **Charts** tab, see Section 11.3, "Working with the Charts Tab," on page 166.

◆ Monitor key performance indicators

The Chart Console is the best option for monitoring performance indicators because it allows you to create charts from specific data streams and add them to chart groups from which you can quickly assess overall performance. If necessary, you can obtain detailed information about areas of concern. For example, you want to monitor your IIS, Websphere, and SQL Servers for trends that indicate potential problems and be able to obtain information about specific servers for troubleshooting purposes. Run the IIS_HealthCheck, WebSphereAppSrvUNIX_HealthCheck, and SQL_Connectivity Knowledge Scripts and configure the jobs to collect data. In the Chart Console, create charts from the data streams for the servers you want to monitor and create a group that contains only those charts. You can scan the chart snapshots as a group to quickly identify areas of concern, then select individual charts to obtain detailed information for troubleshooting purposes.

For more information about using the Chart Console, see Section 11.4, "Working with the Chart Console," on page 170.

◆ Identify trends based on historical information

Reports are the best option for obtaining historical data on a schedule to identify trends.

The application-specific and generic reports AppManager provides are the best option for obtaining short-term, static data. For more information about basic AppManager reports, see Chapter 10, "Running Reports," on page 151.

NetIQ Analysis Center is the best option for obtaining long-term data. For more information about using Analysis Center, see the Analysis Center product documentation.

11.3 Working with the Charts Tab

The **Charts** tab allows you to generate charts and analyze data directly from Control Center. You can use the **Charts** tab to perform the following functions:

◆ Locate and chart data streams related to specific physical computers, logical servers, jobs, and events

For more information about locating and charting specific data streams, see Section 11.3.1, "Viewing Data Streams," on page 167 and Section 11.3.3, "Generating Charts," on page 168.

◆ Obtain detailed information about data points of interest

For more information about obtaining detailed information, see Section 11.3.4, "Obtaining Detailed Information," on page 169.

◆ Obtain real-time updates based on the update interval you specify

For more information about the options you can set, see "Working with Lists" on page 38.

◆ Print, email, or export charts and data stream details

For more information about printing and exporting chart information, see Section 11.3.5, "Printing or Exporting Chart Information," on page 170.

11.3.1 Viewing Data Streams

The data streams created by data collection jobs are available on the **Charts** tab in the **Server Information** and **Service Map Object Information** panes. The following table describes the options for viewing data streams.

To view data streams for...	Do this...
Multiple physical computers or logical servers	1. In the **Navigation** pane, select the **Servers** view or **Service Map** view that includes the physical computers or logical servers for which you want to view data streams. 2. In the **Servers** view grid or on the service map, select the physical computers or logical servers. 3. In the **Server Information** pane or the **Service Map Object Information** pane, click the **Charts** tab.
One physical computer or logical server	1. In the **Navigation** pane or on a service map that includes the physical computer or logical server, select the computer or logical server. 2. In the **Server Information** pane or the **Service Map Object Information** pane, click the **Charts** tab.
Specific events	1. In the **Server Information** pane or the **Service Map Object Information** pane, click the **Events** tab. 2. Select the events. 3. On the **Event View Tasks** context menu, click **View Related Data Streams**. The related data streams are highlighted on the **Charts** tab.
Specific jobs	1. In the **Server Information** pane or the **Service Map Object Information** pane, click the **Jobs** tab. 2. Select the jobs. 3. On the **Job View Tasks** context menu, click **View Related Data Streams**. The related data streams are highlighted on the **Charts** tab.

11.3.2 Filtering Data Streams

There are two ways to filter data on the **Charts** tab. You can filter what data streams are available, and you can filter the data available in a given chart.

To filter the data streams available on the **Charts** tab, click **Local Filters**. For more information about applying local filters, see "Working with Lists" on page 38.

To filter the data available within a chart, you can either restrict the data to a specific time range or to a specific number of data points. To restrict the data to a specific time range, click **Time Filter** on the **Charts** tab and then specify the time range you want in the View Properties window. The time filter applies to all charts available on the **Charts** tab for the selected **Servers** view. For more information about the time filter, see Section 11.3.3, "Generating Charts," on page 168.

To restrict the data stream to a specific number of data points, on the **Main** tab of the ribbon, in the **Tools** group, click **Options**, then in the **Charts** group of the **Console** folder select the **Chart only the most recent n data points for each data stream** option. The data points filter applies to all charts in all **Servers** views. For more information about chart options, see Section 3.2.1, "Console Options," on page 41.

If any charts are filtered, the Chart title displays "(filtered)" to indicate a filter is applied to the data.

11.3.3 Generating Charts

When you select the checkbox next to a data stream on the **Charts** tab, AppManager automatically charts the data stream and displays the results on the tab. You can chart more than one data stream at a time, and you can include more than one data stream in a single chart. The following options for controlling chart contents are available from the **Data Stream Tasks** context menu:

◆ **Delete Data Stream**

The availability of this option depends on the Control Center permissions for the user. For more information about setting permissions in Control Center, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

◆ **Chart Data Stream**

◆ **Do Not Chart Data Stream**

◆ **Chart Only Selected Data Streams**

Once you generate a chart, options for controlling the chart appearance are available on the **Charts** tab and from the **Chart Tasks** context menu. The following table describes the options for controlling the chart appearance.

Option	Location	Description
View Type	**Chart Tasks** context menu and **Charts** tab	Provides the following options for charting multiple data streams: ◆ **Combined** - Generates a single chart of multiple data streams. ◆ **Tabbed** - Generates a separate chart for each data stream and displays each chart on a separate tab. ◆ **Stacked** - Generates a separate chart for each data stream and displays the charts in stacked panes on the **Charts** tab. You also have the option to make each chart the same height. ◆ **Time-Aligned** - Generates a separate chart for each data stream and aligns the time axes so you can correlate spikes in different data streams.

Option	Location	Description

Chart Type	**Chart Tasks** context menu and **Charts** tab	Provides the following chart style options:
		
		    ◆ **Line**
		
		    ◆ **Area**
		
		    ◆ **Bar**
		
		    ◆ **Spline**
		
		    ◆ **Step Line**

Show Enhanced Charts/Show Simple Charts	**Charts** tab	Allows you to turn on or off charting the data streams selected in the **Charts** tab grid with a thicker line and including or hiding data point markers
Show Legend/Hide Legend	**Chart Tasks** context menu and **Charts** tab	Allows you to show or hide the chart legend You might want to hide the legend when viewing charts of multiple data streams to reduce the amount of scrolling required. You might want to show the legend when exporting a chart. This option is not available for the Tabbed and Stacked view types.
Time Filter	**Charts** tab	Allows you to chart only the data points collected during a specific time period This option applies to all charts in the selected view. If you select a time filter, the chart title displays "(filtered)" to indicate you applied a filter to the data.
Color	**Charts** tab grid	Allows you to change the data stream color To change the color, click the **Color** cell in the grid and select the desired color.

11.3.4 Obtaining Detailed Information

You can use the **Charts** tab to obtain detailed information about data points of interest for troubleshooting purposes.

To obtain detailed information:

1 On the **Charts** tab, select the data stream associated with the data points for which you want more information.

2 On the **Data Stream Tasks** context menu, select **Data Stream Properties**.

The Data Stream Properties window lists the data stream properties and each data point in the data stream.

3 In the list of data points, select the data point of interest and click **Show Data Details**.

The Data Details window provides details about the time and value pair.

11.3.5 Printing or Exporting Chart Information

You can print, export, or email charts and data stream details. You can export chart information in a variety of file formats or email information as an attachment in a variety of file formats. The following table describes the options for printing and exporting chart information.

To print, export, or email...	Do this...
All generated charts	1. On the **Chart Tasks** context menu, click **Print/Export/Email** > **All Charts**. 2. Choose the appropriate print, export, or email options.
Specific charts	1. On the **Chart Tasks** context menu, click **Print/Export/Email** > **Selected Charts**. 2. Select the charts to print, export, or email and click **OK**. 3. Choose the appropriate print, export, or email options.
Charts tab grid details for all collected data streams	1. On the **Data Stream Tasks** context menu, click **Print/Export/Email** > **All Data Streams**. 2. Choose the appropriate print, export, or email options.
Charts tab grid details for specific data streams	1. On the **Charts** tab, select the data streams you want to print, export, or email. 2. On the **Data Stream Tasks** context menu, click **Print/Export/Email** > **Selected Data Streams**. 3. Choose the appropriate print, export, or email options.
Time and value information for all data points in a data stream	1. On the **Charts** tab, select the data stream. 2. On the **Data Stream Tasks** context menu, click **Data Stream Properties**. 3. Click **Print/Export/Email**. 4. Choose the appropriate print, export, or email options.
Details for a specific data point	1. On the **Charts** tab, select the data stream. 2. On the **Data Stream Tasks** context menu, click **Data Stream Properties**. 3. Select the data point and click **Show Data Details**. 4. Click **Print/Export/Email**. 5. Choose the appropriate print, export, or email options.

11.4 Working with the Chart Console

The Chart Console provides functionality not available on the **Charts** tab. The Chart Console allows you to perform the following functions:

- Create chart groups from which you can quickly assess overall performance.

  For more information about creating groups, see .

- Customize the chart appearance through options that allow you to:
  - Orient your view of the data using real-time rotation, zoom, and pan.

- Re-order data streams within a chart to improve readability.
- Customize the chart color scheme.
- Mix and match chart styles across different data streams in a chart. Proportional graphing between streams correctly displays different monitoring intervals.

For more information about the customization options, see Section 11.4.5, "Customizing Charts," on page 174.

- Identify when a data stream breaches a high or low threshold by creating static threshold indicators.

For more information about creating threshold indicators, see Section 11.4.6, "Setting Thresholds," on page 175.

Use AppManager Security Manager to grant users permission to use the Chart Console. For more information about using Security Manager, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

11.4.1 Starting the Chart Console

You can start the Chart Console from the Control Center console, the Windows desktop, or the Operator Console.

To start the Chart Console from the...	Do this...
Control Center console	1. In a **Servers**, **Jobs**, or **Events** view, right-click an item in the view pane and select **Utilities** > **AppManager Utilities** > **Chart Console**. 2. Select the data streams you want to chart. 3. Specify a chart title, select a color scheme, and click **Finish**.
Windows desktop	1. Start the Chart Console from the AppManager program folder. 2. Specify the name of the SQL Server and, if applicable, instance that hosts the QDB that stores the data streams you want to chart. To specify a SQL Server instance, use the format *servername\instance*. 3. Select the QDB. 4. Select whether to use Windows or SQL Server authentication to connect to the QDB. If you select SQL Server authentication, also provide the account user name and password. 5. Click **Logon**.

To start the Chart Console from the...	Do this...
Operator Console	1. (Conditional) If you want to create a chart based on a specific computer, right-click the computer in the tree view and select **Create Chart by Selected Computer**.
	2. (Conditional) If you want to create a chart based on a specific job, right-click the job on the **Jobs** tab and select **Create Chart by Selected Job**.
	3. (Conditional) If you want to chart a specific data stream, right-click the data stream on the **Graph Data** tab and select **Create Chart**.
	4. (Conditional) If more than one data stream is available, select the data streams you want to chart.
	5. Specify a chart title, select a color scheme, and click **Finish**.

11.4.2 Viewing Charts

The Chart Console consists of a tree pane and a chart pane. The tree pane lists available charts and chart groups and allows you to create new charts and chart groups. For more information about creating charts, see Section 11.4.3, "Creating Charts," on page 172. For more information about organizing charts into groups, see Section 11.4.4, "Organizing Charts into Groups," on page 173.

When you select a chart group in the tree pane, a chart snapshot of each chart in the group displays in the chart pane. Chart snapshots provide a quick view of the most recent data points for the associated data streams. When you select a chart snapshot, a complete view of the data displays in the chart pane.

When you select an individual chart in the tree pane, a complete view of the data displays in the chart pane.

11.4.3 Creating Charts

Use the Add Data wizard to create new charts and add data streams to existing charts.

To create a chart:

1 In the tree pane, select the group in which you want to create the chart.

To share the chart with other console users, select the **Public** group. If the **Public** group is not available, check with your AppManager administrator to ensure you have permission to view and edit charts in the **Public** group. For more information about chart groups, see Section 11.4.4, "Organizing Charts into Groups," on page 173. For more information about setting permissions for the Chart Console, see the *Administrator Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).

To create a chart without adding it to a group, select the **Charts** folder.

2 On the **Chart** menu, select **Create Chart**.

3 Select the QDB, view, and filter for which you want to view available data streams, and then click **Next**.

4 Select the data streams you want to chart and click **Next**.

5 Specify a chart title and select a color scheme.

6 (Optional) If you want to display dates and times, select the option **Display dates and times based on the local time of the managed computer**.

7 Click **Finish**.

A snapshot of the new chart appears in the chart pane.

To add a data stream to a chart:

1 Select the chart to which you want to add data.

2 On the **Chart** menu, select **Add to Chart**.

3 Select **Existing Chart** and click **OK**.

4 Select the QDB, view, and filter that contains the data stream you want to add and click **Next**.

5 Select the data streams you want to add and click **Next**.

6 Specify a chart title, select a color scheme, and click **Finish**.

The updated chart appears in the chart pane.

To delete a data stream from a chart:

1 Select the chart from which you want to remove data.

2 In the **Legend** pane, right-click the data stream and select **Delete**.

11.4.4 Organizing Charts into Groups

You can use the Chart Console to organize charts into logical groups. For example, you might want to create a group for all charts related to West Coast computers or create separate groups for charts related to the Engineering and Finance departments in your organization. You can also create groups within groups.

To improve startup performance of the Chart Console, organize charts into groups so that the top-most level of the tree pane does not include any charts.

To create a group:

1 In the tree pane, select the **Charts** folder.

2 On the **Chart** menu, select **Add Group**.

3 Specify the group name and click **OK**.

To delete a group:

1 In the tree pane, select the group.

2 On the **Chart** menu, select **Delete Group**.

3 Select the charts or chart group you want to delete and click **OK**.

11.4.5 Customizing Charts

A standard toolbar provides quick access to frequently-used Chart Console commands. To show or hide the toolbar, select **View** > **Toolbar**. To view a description of a toolbar button, hover the mouse over the button.

Several toolbars for customizing charts are available in the chart pane. The following table describes the chart pane toolbars.

This toolbar...	Does this...
Time frame	Allows you to select the range of data to chart.
	By default, AppManager stores the data points that Knowledge Script jobs collect for a given day in a data table for immediate display in charts and graphs for 30 days, and each day removes any data tables that are more than 30 days old.
	To set QDB preferences, use the AppManager Operator Console. For more information about using the Operator Console to set preferences, see the *Operator Console User Guide for AppManager*, available on the AppManager Documentation page (http://www.netiq.com/documentation/appmanager-9-2).For information about changing the default, see Section 3.2.8, "Global Preferences Options," on page 44.
Legend	Lists the data streams in the active chart and allows you to perform the following functions:
	◆ Delete a data stream from the chart. To delete a data stream, right-click the data stream and select **Delete**.
	◆ Move a data stream to the foreground or background by dragging it to the desired location in the list.
	◆ Change the properties of a data stream. To change the properties, right-click the data stream and select **Properties**.
	◆ Change the style of a data stream. To change the style, right-click the data stream and select **Style**.
	◆ Make a data stream transparent. To make a data stream transparent, right-click the data stream and select **Transparent**.

To show or hide the toolbars, right-click the chart and select **Show time frame** or **Show legend**.

The Chart Console provides some customization options that are not available from the chart pane toolbars. The following table describes additional customization options.

To...	Do this...
Display charts horizontally	In the chart pane, right-click the chart and select **Horizontal chart**.
Make all data streams in a chart transparent	In the chart pane, right-click the chart and select **Transparent**.

To...	Do this...
Display only a specific portion of a chart	1. In the chart pane, ensure the time frame is set to **All**.
	2. Right-click the chart and select **Properties**.
	3. Click the **Range** tab.
	4. Select **Specify start date and time** and specify the date and time for which you want to start displaying data.
	5. Select **Specify end date and time** and specify the date and time for which you want to stop displaying data.
	6. Click **OK**.
	The chart shows only data collected between the date and times you specified.

11.4.6 Setting Thresholds

You can set thresholds to identify when a data stream crosses a critical value. For example, if you are using the NT_MemUtil Knowledge Script to monitor physical and virtual memory use, you can set thresholds to identify a warning level and a critical level.

You can apply thresholds to individual data streams or entire charts.

To set thresholds:

1 In the chart pane, right-click the chart and select **Properties**.

2 Click the **Data Streams** tab.

3 (Optional) If you want to set a threshold for a specific data stream, select the data stream from the **Stream** drop-down menu.

4 Click **Add**.

5 Complete the following actions and click **OK**:

- Select whether to apply the threshold to the specific data stream or the entire chart.

- Specify the threshold value.

- Select a color for the threshold indicator.

- Select whether to activate the threshold indicator.

6 Repeat Step 4 on page 175 and Step 5 on page 175 to set additional thresholds.

7 On the Chart Properties window, click **OK** to add the threshold indicators to the chart.

11.4.7 Obtaining Detailed Information

You can view a list of the data points for each data stream included in a chart, or you can view detailed information about a specific data point.

To view a list of the data points for each data stream included in a chart, right-click the chart and select **Data viewer**. For each data stream, the Data Viewer displays each time and value pair collected.

To view detailed information about a specific data point, double-click the data point in the chart. The Graph Data Details window provides the following information about the data point:

- ◆ Computer name
- ◆ Type of data
- ◆ Value of the data point
- ◆ Time at which the data point was collected
- ◆ Additional details that vary with the type of data

11.4.8 Changing Chart Properties

Use the Chart Properties window to configure properties such as frame rate, graphical style and color scheme, threshold values, and dates and times for which data is displayed. To change the properties of a chart, right-click the chart and select **Properties**.

The following sections describe the properties you can set on each tab.

General Tab

The following table describes the properties you can set on the **General** tab.

Property	Description
Title	Specifies the chart title.
Y-Axis Label	Specifies a label for the Y-axis.
View	Provides the following chart rendering options: ◆ **Perspective** - Renders the chart so that the foremost data stream appears larger than the last data stream. ◆ **Parallel** - Renders the chart so that all data streams have the same size relative to value.
WARNING: Frame Rate: (CPU Usage)	Determines the amount of CPU devoted to interactive rendering (rotating, zooming, panning, and scrolling). A low frame rate is less fluid, but uses less CPU. A high frame rate offers the best visual performance, but uses 100 percent of the CPU.
Vertical Maximum	Provides the following options for adjusting the Y-axis: ◆ **Auto** - Automatically sets the maximum value of the Y-axis to a value greater than the highest point rendered. ◆ **Manual** - Allows you to specify a maximum value for the Y-axis.
Wireframe when rotating, zooming, and panning	Renders the chart in wireframe when rotating, zooming, or panning. This option conserves CPU resources.
Screentips for data points	Displays a screen tip when you hover the mouse over a data point.
Data stream settings	Provides the following options for adjusting data appearance: ◆ **Depth** - Sets the depth of each data stream along the Z-axis. ◆ **Point Density** - Sets the density of data points along the X-axis.

Colors Tab

The following table describes the properties you can set on the **Colors** tab.

Property	Description
Scheme	Defines the chart color scheme. To save the currently-defined color scheme under a unique name, click **Save As**.
Colors	Defines colors for the following chart items: ◆ Title ◆ Grid lines ◆ Grid panels (the sides and bottom of the grid) ◆ Time scale (the numbers in the time line along the X-axis) ◆ Left scale (the numbers along the left Y-axis) ◆ Right scale (the numbers along the right Y-axis) ◆ Background (the background color for the chart pane) ◆ Gradient (creates a gradient for the background of the chart pane and defines the color for the bottom of the gradient; the color you selected for the background serves as the top color in the gradient) ◆ Data gap (missing points in a data stream rendered with the **Area** option) ◆ Data stream colors
Solid Grid	Renders the grid as a solid object.
Show Left Values	Displays values along the Y-axis on the left side of the grid.
Show Right Values	Displays values along the Y-axis on the right side of the grid.

Data Streams Tab

The following table describes the properties you can set on the **Data Streams** tab.

Property	Description
Stream	Data stream for which you want to set properties.
Style	Visual style of the data stream. The following styles are available: ◆ Area ◆ Bar ◆ Cylinder ◆ Line ◆ Ribbon

Property	Description
Scale	Scale for the data stream.
	The Y-axis is scaled relative to its original scale of 1.0. The grid is not affected.
High Thresholds	Thresholds to display in the chart.
	For more information about setting thresholds, see Section 11.4.6, "Setting Thresholds," on page 175.
Other information	SQL query used to retrieve the data stream.
	You cannot modify this property.

Range Tab

The following table describes the properties you can set on the **Range** tab.

Property	Description
Specify start date	Enables fields that allow you to specify the date and time for which you want to start displaying data in the chart.
Start Date	Date for which you want to start displaying data.
Start Time	Time for which you want to start displaying data.
Specify end date	Enables fields that allow you to specify the date and time for which you want to stop displaying data in the chart.
End Date	Date for which you want to stop displaying data.
End Time	Time for which you want to stop displaying data.

If the **Range** tab is not available, click **Cancel** and set **Time frame** to **All** in the chart pane.

11.4.9 Exporting Chart Information

You can export charts and chart data to a variety of formats. When exporting data, you can export all data or only a range of data. For more information about selecting a range of data, see "Selecting a Range of Data Points for Export" on page 179.

The following table describes the options for exporting chart information.

To export...	Do this...
Chart to an XML-formatted report	1. In the chart pane, right-click the chart and select **Export** > **Report to XML**.
	2. Select a location for the XML file, specify a file name, and click **Save**.

To export...	Do this...
Chart to an HTML-formatted report	1. In the chart pane, right-click the chart and select **Export** > **Report to HTML**.
	2. Select the type of report to generate and the data streams to include and click **Next**.
	3. Specify a report title, select a color scheme, and click **Finish**.
	4. Select a location for the HTML file, specify a file name, and click **Save**.
	The file is saved in a sub-folder named according to the report title. For example, if the report name is **IIS_CpuHigh (BALQEROW06A12).htm**, the report is saved in a subfolder named **IIS_CpuHigh (BALQEROW06A12)**.
Data to the Windows clipboard	1. In the chart pane, right-click the chart and select **Export** > **Data to Clipboard**.
	2. Select the appropriate export options and click **OK**.
	3. Paste the data into a text editor.
Data to a text file	1. In the chart pane, right-click the chart and select **Export** > **Data to File**.
	2. Specify the export path and file name, select the appropriate export options, and click **OK**.
Chart to the Windows clipboard as a `.png` image	1. In the chart pane, right-click the chart and select **Export** > **Image to Clipboard**.
	2. Paste the image into any application that accepts a `.png` image.
Chart to a `.png` file	1. In the chart pane, right-click the chart and select **Export** > **Image to File**.
	2. Select the export location, specify a file name, and click **Save**.

Selecting a Range of Data Points for Export

When exporting chart data, you can export all data or only the range of data you select.

To select a range of data points:

1 In the chart pane, right-click the chart and select **Select Data**.

A vertical bar appears in the chart pane.

2 Position the vertical bar at the point where you want to begin your selection.

3 Click and drag to highlight the desired data points.

4 Export the data.

For more information about exporting data, see Section 11.4.9, "Exporting Chart Information," on page 178.

11.4.10   Setting Chart Console Preferences

Preferences control how the Chart Console operates.

Changes to the Chart Console preferences apply only to the current user and QDB.

Preference	Description
Refresh Interval	Frequency with which the Chart Console updates data points for each chart in the currently selected group.
	The default interval is once per minute. At each interval, the Chart Console updates the data streams in the chart snapshots and the current chart (if you are viewing one) to include data points collected since the previous interval.
	To improve overall performance of the Chart Console, reduce the number of data points that must be updated during a refresh interval by organizing charts into several groups. For more information about creating groups, see Section 11.4.4, "Organizing Charts into Groups," on page 173.
Chart Snapshots: Fetch no more than n points	Maximum number of data points returned for chart snapshots.
	The default is 50 data points. If you specify 0, all points are returned.
	As a rule, you can improve overall performance of the Chart Console by reducing the number of data points returned for chart snapshots.
Image Width	Width, in pixels, of chart snapshots.
	The default is 280 pixels.
Image Height	Height, in pixels, of chart snapshots.
	The default is 190 pixels.
Charts: Fetch no more than n points	Maximum number of data points returned for charts.
	The default is 2,000 data points. If you specify 0, all points are returned.

12 Customizing Menu Extensions

This chapter provides an overview of how to extend the AppManager Utilities menu. You can update the AppManager Utilities menu to add extensible menus for:

- **Jobs** views
- **Servers** views
- **Events** views
- **Knowledge Scripts** views
- **Service Map** views

12.1 Extending the Context Menus for Views

To extend a context menu, create an XML file that:

- Specifies applications to be launched and parameters to be passed to it.
- Utilizes a replaceable parameter mechanism.
- Supports two types of replaceable parameters:
  - Data-centric parameters that map to the selected data in the view. Any column in the view for the current selected set of data can be passed to the command-line utility.

    If the column is **not** displayed in the view, the parameter cannot access the corresponding data.
  - System parameters, such as an installation path, in the format `%%PARM%%`.
- Supports cascading menus in the context menu, up to two levels.
- Supports multiple selections by use of a separator value.
- Is dropped in `\bin` directory.
- Is loaded each time the menu is displayed – no need to restart the Control Center console because of changes.
- Supports multiple XML files - must follow the naming convention `<filename>.Extensions.xml`.

12.2   Example XML Format

```xml
<?xml version="1.0" encoding="utf-8" ?>
<MenuExtensions>
  <View TypeID="NQCCEVENTS">
    <Item Name="AppManager Utilities">
       <Item Name="Launch Distributed event Console"
                                 VisibleOnSingleSelectOnly="true" >
        <Program>%%_AMPATH%%\bin\NetIQDEC.exe</Program>
        <Parameters></Parameters>
        <Directory></Directory>
      </Item>
      <Item Name="Launch Operator Console">
        <Program>%%_AMPATH%%\bin\netiq.exe</Program>
        <Parameters Separator="|">/SERVER=%%_COMPUTERNAME%% /
                            LOGONNAME=%%_LOGONNAME%%</Parameters>
      </Item>
      <Item Name="Chart Console">
        <Program>%%_AMPATH%%\bin\AMChartCon.exe</Program>
        <Parameters></Parameters>
      </Item>
    </Item>
    <Item Name="Test" VisibleOnSingleSelectOnly="true">
      <Program>%%_AMPATH%%\bin\AMChartCon.exe</Program>
      <Parameters>/Path=%%_DCPATH%% /SERVER=%%_COMPUTERNAME%%
                                    %%_AMOCPATH%%</Parameters>
    </Item>
     </View>
</MenuExtensions>
```

12.3   Menu Extension XML Keywords

- MenuExtensions
    - View - An element that contains items (menu items)

- TypeID - This attribute specifies the View ID the contained items will be associated with. The type IDs are as follows:
    - NQCCJOBS
    - NQCCServers
    - NQCCEVENTS
    - NQCCKS
    - NQCCServiceMap

- Item - Represents an item that will appear in the menu
    - Name - an attribute that specifies the name of the item - which will appear as the menu text
    - Program - specifies the program to be launched
    - Parameters - specifies the parameters to be passed to the program
    - Separator - specifies the separator character that will appear between multiple selections
    - Directory - specifies the working directory the program will run in (optional)

NOTE: Only the parameters support handling multiple selected data.

12.4   Cascading Menus within Menu Extensions

To create a cascading menu, embed one item tag within another:

```
<Item Name="Menu Name">
  <Item Name="Item 1">
     ....
  </Item>
  <Item Name="Item 2">
     ....
  </Item>
</Item>
```

12.5   Event Field Parameters (Data-centric Parameters)

- DataSourceName
- Computer
- Severity1
- Severity2
- Status
- ChildComment
- KPName
- LastOccurTime
- Occurrence
- eventMsg
- eventID
- ParenteventID
- jobID
- ModificationTime
- Severity
- CheckStatus

12.6   Job Field Parameters (Data-centric Parameters)

- DataSourceName
- Computer
- KSName
- SubmitUserName
- jobID
- ParentjobID
- SubmitTime
- LastRunTime
- StopTime
- EnterprisejobID

- VirtualjobIDStatus
- ModificationTime
- DropKPObjID
- VersID
- LangID
- SyncTime
- IconStatus
- IconPendingjobStatus
- IconGroupOrPolicy
- IconHasAction,StatusString

12.7 Knowledge Scripts View Parameters (Data-centric Parameters)

- CreateTime
- Name
- Comment
- IconID
- ParentVKSID
- Platform
- Status
- VirtualKSID
- ModificationTime
- VersID
- LangID
- CreateUserID
- Category
- PlatformText

12.8 Servers View Parameters (Data-centric Parameters)

- DataSourceName
- Computer
- eventSeverity
- ObjID
- TypeName
- TypeID
- Status
- ModificationTime

- jobCount
- Server
- Platform
- SyncTime
- VersIDLangID
- RootMachineObjID
- AgentStatus
- NT_MachineFolder:Product Type
- NT_MachineFolder:OS Version
- NT_MachineFolder:CSD
- NT_MachineFolder:Build Number
- NT_MachineFolder:Install Time
- NT_MachineFolder:Current Type
- NT_MachineFolder:System Directory
- NT_MachineFolder:NT Domain (role)
- SQLT_Server:ServerName
- SQLT_Server:Version
- SQLT_Server:RootDir
- SQLT_Server:MasterDevPath
- SQLT_Server:ErrorLog
- SQLT_Server:LoginSecurity
- SQLT_Server:NTeventLogging
- SQLT_Server:InCluster

NOTE: There can be many more parameters than are listed here. To discover their correct names, use the _LISTSCHEMA parameter and then go to the debug window.

12.9 Control Center Standard Parameters (Built-in Parameters)

- _AMPATH - the path where AM is currently installed.
- _DCPATH - the path where Diagnostic Console is currently installed.
- _MACHINENAME - environment variable.
- _OSVERSION - environment variable.
- _USERDOMAINNAME - environment variable.
- _USERNAME- returns the current logged in NT user - this is not always the same as the user who logs on to the Control Center console.
- _CURRENTDIRECTORY - environment variable.
- _SYSTEMDIR - environment variable.
- _PERSONALDIR - environment variable.
- _PROGRAMSDIR - environment variable.

- ◆ _COMMANDPROGRAMDIR - environment variable.
- ◆ _REPOSITORYSERVER - parses the repository column in the current list View and returns the name of the server (e.g. beltaftb01). If multiple selections are specified this function will only be based on the first selected row.
- ◆ _REPOSITORYNAME - parses the repository column in the current list View and returns the name of the repository (e.g. QDB). If multiple selections are specified this function will only be based on the first selected row.
- ◆ _REGLOOKUP(hive, subkey, value) - looks up any registry entry and returns its value as a string. For example: `%%_REGLOOKUP("HKEY_LOCAL_MACHINE", "SOFTWARE\NetIQ\AppManager\4.0","InstallPath")%%`
- ◆ _ENVLOOKUP(envVarName) - looks up any environment variable on the console machine. For example: `%%_ENVLOOKUP("Path")%%`
- ◆ _LISTSCHEMA - Lists all column names for the current View type that may be used as a data-centric parameter. This method is useful for determining the real names of the columns.
- ◆ _LOGONNAME -The name of the user currently logged into the CC console.
- ◆ _LOGONMODE - The authentication mode of the current Control Center console user - NT or SQL (0=WinNT, 1=SQL).
- ◆ _LOGONSERVER - The server where the Control Center repository database exists.

12.10 Debugging Command-line Parameters

The final command-line to be executed can be examined in the Appmanager Control Center 6.5 debug window (accessible via CTRL+F8). To enable this functionality you will need to enable a higher level of detail in the diagnostic window.

The `NQConsole.EXE.Config` file enables the diagnostic window to display more detailed diagnostic messages. Following is an example of this file - it should be placed in the `bin` directory.

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <configSections>
  </configSections>
  <appSettings>
    <add key="TRACELEVEL" value="INFO" />
  </appSettings>
</configuration>
```

NOTE: Do not leave this file in place in production environments because it could negatively affect performance.

12.11   More Examples

```xml
<?xml version="8.0" encoding="utf-8" ?>
<MenuExtensions>
  <View TypeID="NQCCServers">
    <Item Name="AppManager Utilities" VisibleOnSingleSelectOnly="true">
      <Item Name="Chart Console" VisibleOnSingleSelectOnly="true" >
        <Program>%%_AMPATH%%\bin\amchartcon.exe</Program>
        <Parameters> /SERVER=%%_REPOSITORYSERVER%% /
                     R=%%_REPOSITORYNAME%% %%_AMOCPATH%%</Parameters>
      </Item>
      <Item Name="Diagnostic Console"
                                  VisibleOnSingleSelectOnly="true" >
        <Program>%%_DCPATH%%\bin\rtdc.exe</Program>
        <Parameters> SERVER=%%_REPOSITORYSERVER%%
                          dbname=%%_REPOSITORYNAME%%</Parameters>
      </Item>
      <Item Name="List jobs on Agent via NetIQCtrl">
        <Program>passthrough.cmd</Program>
        <Parameters>"%%_AMPATH%%\bin\netiqctrl.exe" job
                               %%_COMPUTERNAME%% netiqmc</Parameters>
      </Item>
      <Item Name="Operator Console"
                                  VisibleOnSingleSelectOnly="true" >
        <Program>%%_AMPATH%%\bin\netiq.exe</Program>
        <Parameters> /S=%%_REPOSITORYSERVER%% /
                          DB=%%_REPOSITORYNAME%% </Parameters>
      </Item>
    </Item>
    <Item Name="Windows Utilities"
                                  VisibleOnSingleSelectOnly="true" >
      <Item Name="Computer Management"
                             VisibleOnSingleSelectOnly="true" >
        <Program>compmgmt.msc</Program>
        <Parameters> /computer=%%_COMPUTERNAME%%</Parameters>
      </Item>
      <Item Name="Event Viewer" VisibleOnSingleSelectOnly=
                                                        "true" >
        <Program>eventvwr.msc</Program>
        <Parameters>/computer=%%_COMPUTERNAME%%</Parameters>
      </Item>
      <Item Name="Explore" VisibleOnSingleSelectOnly="true" >
        <Program>explorer.exe</Program>
        <Parameters> \\%%_COMPUTERNAME%%</Parameters>
      </Item>
      <Item Name="Explore C$" VisibleOnSingleSelectOnly="true" >
        <Program>explorer.exe</Program>
        <Parameters> \\%%_COMPUTERNAME%%\C$</Parameters>
      </Item>
      <Item Name="Ping Machine" VisibleOnSingleSelectOnly=
                                                        "true" >
        <Program>passthrough.cmd</Program>
        <Parameters>ping %%_COMPUTERNAME%%</Parameters>
      </Item>
      <Item Name="Service Manager"
                             VisibleOnSingleSelectOnly="true" >
        <Program>services.msc</Program>
        <Parameters> /computer=%%_COMPUTERNAME%%</Parameters>
```

```
      </Item>
      <Item Name="Terminal Services"
                           VisibleOnSingleSelectOnly="true" >
        <Program>mstsc</Program>
        <Parameters>/v:%%_COMPUTERNAME%%</Parameters>
      </Item>
      <Item Name="Trace Route" VisibleOnSingleSelectOnly=
                                            "true" >
        <Program>passthrough.cmd</Program>
        <Parameters>tracert %%_COMPUTERNAME%%</Parameters>
      </Item>
    </Item>
  </View>
</MenuExtensions>
```

12.12 A Useful Batch File When Running a Command Program

If you want to see the output of a command program, use this snippet to create a batch file that pauses the program automatically:

```
@echo off
%1 %2 %3 %4 %5 %6 %7 %8 %9
Pause
```