Appendix C

Xtremesoft AppMetrics and NetIQ AppManager

Overview

AppMetrics provides a facility that extends your NetIQ AppManager network. This facility uses custom-managed objects that retrieve data from AppMetrics Application Monitors. As an optional part of your AppMetrics setup, you can install an AppManager extension library, Knowledge Scripts, and reports that enable easy access to the data provided by your monitors.

Introduction

Successful installation and configuration of an AppMetrics – AppManager network requires planning. Users should become familiar with the requirements of an AppManager network prior to installing the AppManager components of AppMetrics. Refer to the NetIQ AppManager documentation for information on AppManager network configurations.

Within this Appendix, AppManager-specific terms are used, a few of which are defined below. For further information about each item, refer to the NetIQ AppManager documentation.

Glossary

AppManager Console – Used by system operators and administrators to manage the AppManager aspects of the environment. Allows users to view servers and workstations on the AppManager network, information about the availability and performance of server and application resources.

AppManager Management Server – The service that allows the AppManager Agents on managed clients to communicate with the AppManager Repository.

AppManager Repository – Microsoft SQL Server database that is used as the data repository for AppManager. Stored data contains information about discovered configurations, Knowledge Scripts, events, jobs, and graphs.

AppManager Agent – A service that runs on a managed client.

AppManager Reports – A set of pre-defined AppManager Reports that is available on machines where the Reports components have been installed.

Client Machine – Also called Managed Client. In the AppManager context, a machine managed by AppManager.

Knowledge Script – A business-management or system-management rule that handles one or more tasks.

Configuration

The installation instructions contained in this Appendix assume a typical AppManager site has been configured. This configuration contains the following components:

- AppManager Repository.

- AppManager Management Server.

- One or more Operator Consoles.

- One or more Managed Clients running the AppManager Agent. This includes the machines running the MTS Packages or COM+ Applications.

- (Optional) One or more Web Management Servers.

The AppManager repository and AppManager management server may or may not reside on the same server machine. Network connectivity must be available between the various machines where the AppManager and AppMetrics components are installed.

Installation

In the AppMetrics install, two install components for NetIQ AppManager are available:

- **NetIQ AppManager Repository Files** – Integration components that must be installed on the machine(s) with the AppManager Repository.

- **NetIQ AppManager Console Reports** – AppMetrics report files that you can view within the AppManager 3.x Report Manager. These files are required only on AppManager 3.x Console machines where the reports are to be viewed. They are neither needed nor available for AppManager 4.0.

For any given AppManager network, several AppMetrics install tasks are required, some of which involve one or both of the preceding install components.

Notes

- Please read through the entire Installation section of this Appendix prior to starting an install with the NetIQ AppManager setup type.

- This section assumes that an AppManager Agent has been installed on the AppManager Management Server.

The following table lists the required AppMetrics components on machines performing specific AppManager tasks. If a machine performs multiple tasks, you can combine the components listed for each task:

If the target machine will perform the following AppManager task:	Refer to the following section for instructions on how to install AppMetrics on the machine:	During the install, select the following and component(s):
Managed Client running as an AppMetrics Manager machine	"Installing AppMetrics Using the NetIQ AppManager Setup Type" on Page C-5.	• SQL Server Files • Program Files • Help Files (optional) • Historical Reporting Excel Files (optional) • Manager Monitor Template Files
Managed Client running as an AppMetrics Agent machine	Use the regular AppMetrics Install procedure in Chapter 2. Install AppMetrics using the Agent setup type.	• N/A
AppManager Repository	No standalone AppMetrics install is available for the repository files. Instead, you can install the files within the context of an AppMetrics install on a Manager machine. See "Installing AppMetrics Using the NetIQ AppManager Setup Type" on Page C-5.	• NetIQ AppManager Repository Files.
AppManager 3.x Console	• If installing to an AppManager console machine that will also serve as the AppMetrics Manager machine, use "Installing AppMetrics Using the NetIQ AppManager Setup Type" on Page C-5. • If installing to an AppManager console machine that runs no other AppMetrics component, use "Installing NetIQ AppManager 3.x Console Reports for AppMetrics" on Page C-11.	• NetIQ AppManager Console Reports.

Table C-1   AppManager Tasks and the Required AppMetrics Install Components

NetIQ AppManager Repository Files for AppMetrics

The Repository component for AppMetrics must be installed to each Repository connected to an AppManager Management Server where AppMetrics components and data are stored.
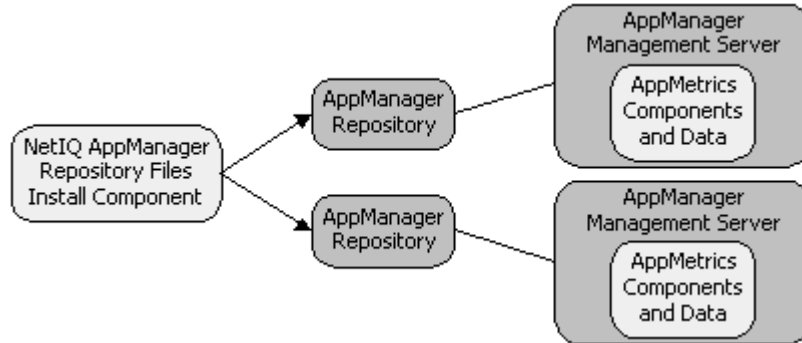
Additionally, the Repository components may be installed to any number of Repository machines that can be reached from the machine where the install is performed. The install enables you to simultaneously target multiple repositories during a single install session via the NetIQ AppManager Installation Information screen (see Figure C-5 on Page C-9). However, uninstalls must be performed from the same machine where the installs were performed.

If multiple Repositories are to be installed from the same location, they must all be installed during the same install session. You cannot split up the installation for the different repositories over different install sessions. And when uninstalling, the components must be uninstalled from all those same Repositories at the same time. Therefore, you should either install the Repository components to one Repository from one Console machine or perform the install from the Repository's Management Server itself.

NetIQ AppManager Console Reports (AppManager 3.x only)

To view the AppManager 3.x Reports provided by AppMetrics, the NetIQ AppManager Console Reports component must be installed onto each Console machine where the reports are to be viewed. To make the reports available through the Web Console, they must be installed to the AppManager Web Management Server.

Installing AppMetrics Using the NetIQ AppManager Setup Type

The following procedure lets you install AppMetrics and its AppManager components onto a Manager machine. It also includes steps for installing the AppManager Repository files for AppMetrics to one or more Repository machines. You will be able to install these repository files to any network-connected machine where an AppManager Repository is available.

1. Log in to the machine with local Administrators privileges.

2. Place the AppMetrics CD-ROM in the CD-ROM drive. Use Windows Explorer to locate the folder corresponding to the operating system of the machine (NT 4.0 or Win2K).

3. Double-click **setup.exe**.

    After the initial Xtremesoft splash screen, the **Welcome** wizard screen appears.

4. Click **Next**.

    This opens the **License Agreement** wizard screen.

5. After reading the license agreement, click **Yes**.

    This opens the **Customer Information** screen (Figure C-1).

Figure C-1   Customer Information Screen

6. Enter your name, company name, and license key. Click **Next**.

    This opens the **Destination Folders** screen.

7. Either click **Browse** to select the folders in which to install the software and data files, or click **Next** to accept the default locations.

    Note: You may receive prompts if the specified folders do not exist on the machine. Click **Yes** to these prompts.
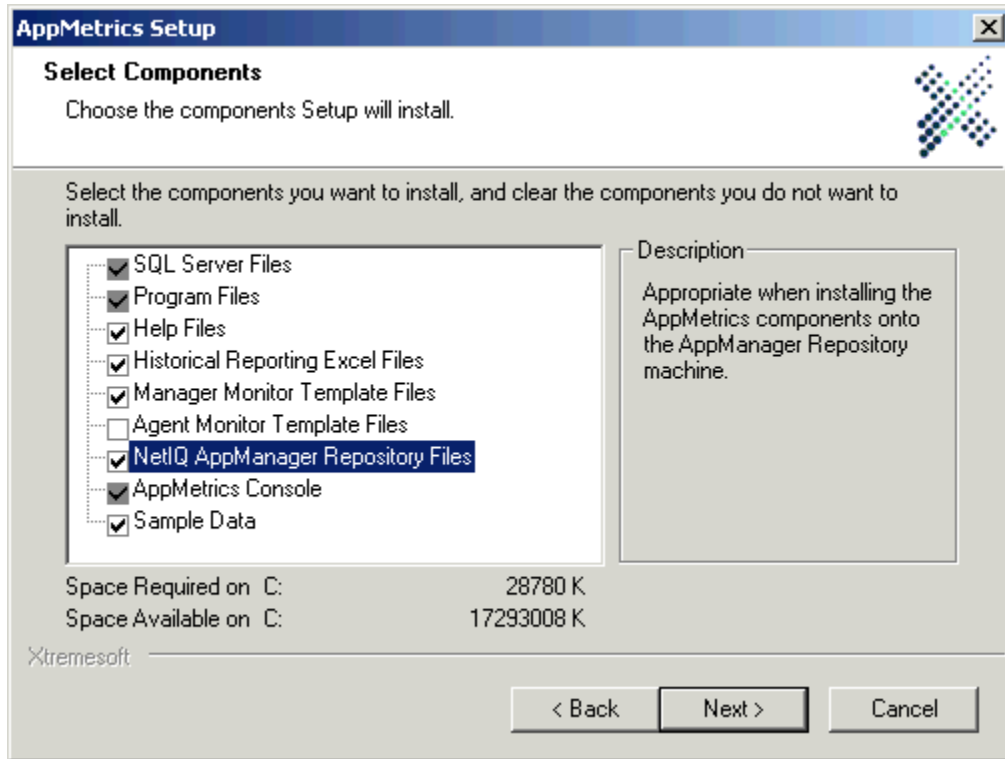
This opens the **AppMetrics Setup Type** screen (Figure C-2).
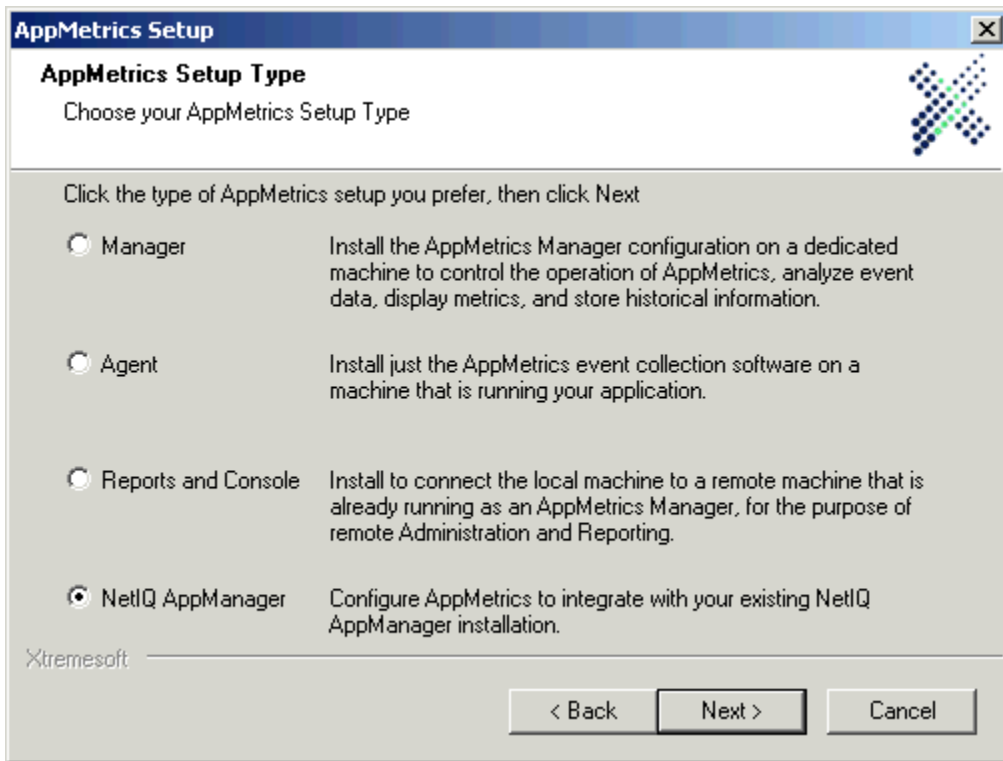
Figure C-2   AppMetrics Setup Type Screen

8.   Choose **NetIQ AppManager**. Click **Next**.

This opens the **Select Components** screen (Figure C-3).

Figure C-3   Select Components Screen

Note: The NetIQ AppManager Console Reports component is only available when installing on a machine with AppManager 3.x.

9. Select the appropriate install component(s) for the target machine based on Table C-1.

   Notes

   - If the machine performs more than one of the listed AppManager tasks, you can combine their lists of install components.

   - You can select the **NetIQ AppManager Repository Files** component, even if the repository is not on the local machine. In a later step, you will have the chance to install the repository files to one or more remote machines.

10. Click **Next**.

    On Windows 2000 machines, this opens the **Service Run As Account** screen.

    On Windows NT 4 machines, this opens the **Remote Access Account** screen, which requests the same information as the Service Run As Account screen (Figure C-4).
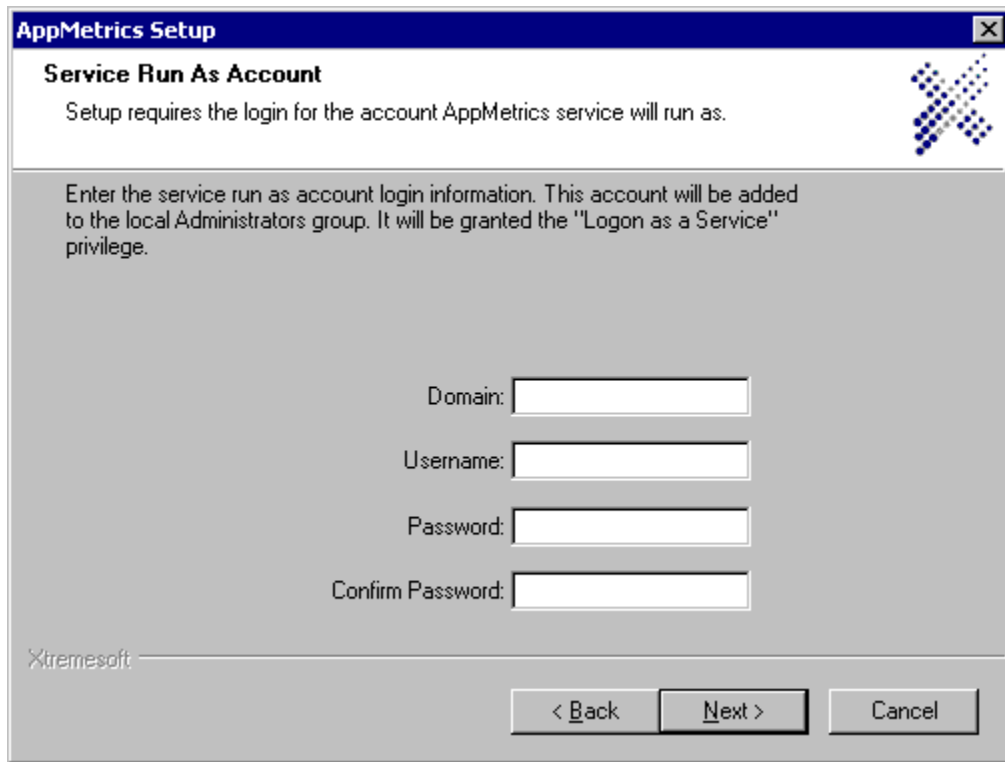
Figure C-4   Service Run As Account Screen

11. Specify a user account and its related information for the AppMetrics Service. Click **Next**.

    The **SQL Server Administrator access** screen appears.

12. Enter the username and password for a SQL Server administrator account or select Windows Authentication. Click **Next**.

    The **AppMetrics SQL Server Account** screen appears.

13. Accept the default **Username** or type a preferred name.

14. Type your preferred password in the **Password** and **Confirm Password** fields. Click **Next**.

- If you selected **NetIQ AppManager Repository Files** as one of the install components in Step 9, then the **NetIQ AppManager Installation Information** screen (Figure C-5) opens. Proceed with Step 15.

- If you did not select **NetIQ AppManager Repository Files**, then the **Select Program Folder** screen opens. Skip to Step 20.
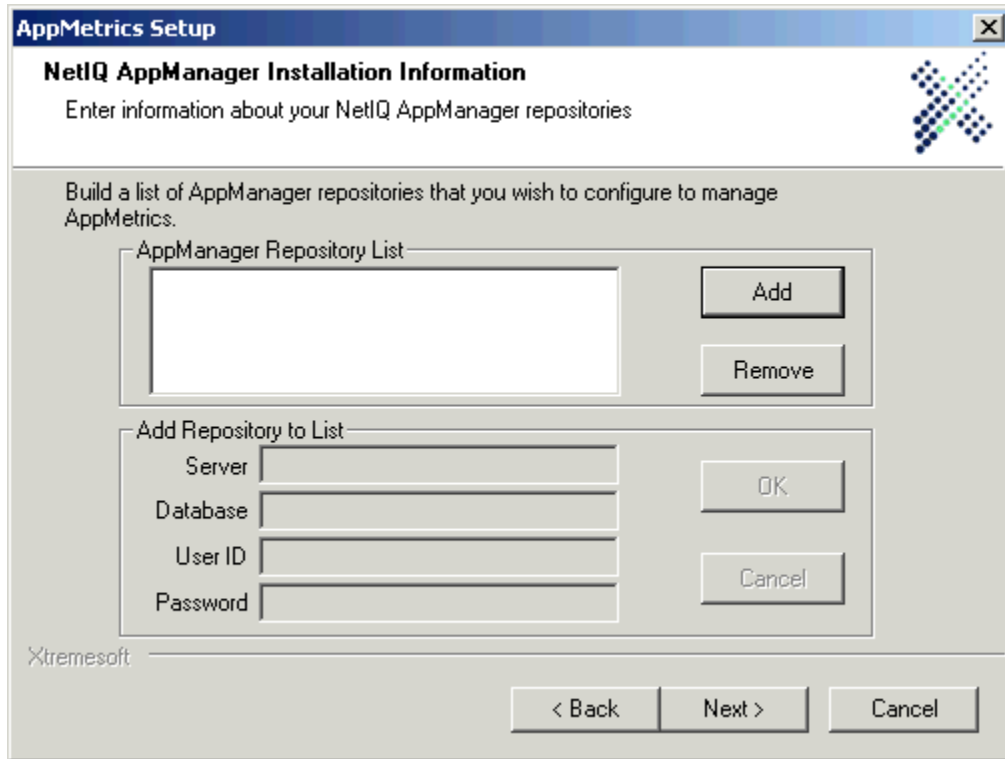
Figure C-5   NetIQ AppManager Installation Information Screen

15. In the **NetIQ AppManager Installation Information** screen, click **Add**.

16. Complete the information in the **Add Repository to List** area.

- **Server** is the SQL Server machine where the repository database resides. This may be the local machine or another accessible machine on your network.

- **Database** is the name of the AppManager repository.

- **User ID** and **Password** correspond to a database administrative account for the repository.

17. Click **OK**.

18. To add another repository to the list, repeat Steps 15 through 17.

19. After adding all the repositories, click **Next**.

   This opens the **Select Program Folder** screen.

20. In the **Select Program Folder** screen, enter the name of the folder in which to add the AppMetrics icons, or leave the default name. Click **Next**.

   The **Setup Status** screen appears. During this screen, the install program copies the files to the selected folder location.

Soon afterwards, the **Monitoring Templates** screen appears.

21. Click **Next**.

The **Installation - Setup is complete** screen appears.

22. Choose the preferred option for rebooting the machine, and then click **Next**.

     Notes

     - AppMetrics will not function unless you reboot the system.

     - If you choose **Reboot Now**, the reboot will occur immediately after you complete Step 24 below.

     The **Installation Complete** screen appears.

23. If you prefer to read the Readme file, select **View readme.txt**.

24. Click **Finish**.

     Note: If you chose **Reboot Now** in Step 22 above, and you chose to view the Readme file in Step 23, then the readme.txt file opens in your text editor. The reboot will occur as soon as you close the readme.txt window.

After the Install

After completing the install onto a Manager machine, assign the AppMetrics Administrators privilege to the run-as account of the NetIQ AppManager Client Resource Monitor service (netiqmc.exe).

Installing NetIQ AppManager 3.x Console Reports for AppMetrics

If you want to use the AppManager 3.x Report Manager on a machine to view AppMetrics data, you must install the AppManager reports for AppMetrics onto the machine. The following procedure is for installing only the AppManager reports for AppMetrics.

1. Log in to the machine with local Administrators privileges.

2. Place the AppMetrics CD-ROM in the CD-ROM drive. Use Windows Explorer to locate the folder corresponding to the operating system of the machine (NT 4.0 or Win2K).

3. Double-click **setup.exe**.

     After the initial Xtremesoft splash screen, the **Welcome** wizard screen appears.

4. Click **Next**.

     This opens the **License Agreement** wizard screen.

5. After reading the license agreement, click **Yes**.

     This opens the **Customer Information** screen (Figure C-1).

6. Enter your name, company name, and serial number. Click **Next**.

     This opens the **Destination Folders** screen.

7. Either click **Browse** to select the folders in which to install the software and data files, or click **Next** to accept the default locations.

     Note: You may receive prompts if the specified folders do not exist on the machine. Click **Yes** to these prompts.

     This opens the **AppMetrics Setup Type** screen (Figure C-2).

8. Choose **NetIQ AppManager**. Click **Next**.

     This opens the **Select Components** screen (Figure C-3).

     Note: The NetIQ AppManager Console Reports component is only available when installing on a machine with AppManager 3.x.

9. Select **NetIQ AppManager Console Reports** and deselect all other components. Click **Next**.

    The **Setup Status** screen appears. During this screen, the install program copies the files to the selected folder location.

    Soon afterwards, the **InstallShield Wizard Complete** screen appears.

10. Click **Finish**.

Scenario for Uninstalling AppMetrics and its AppManager 3.x Repository Files

You can uninstall the AppMetrics Repository integration from AppManager 3.x with the following tasks. However, these tasks do not enable you to remove the managed objects discovered for AppMetrics.

1. In the AppManager Tree View, stop and delete all jobs on AppMetrics objects.

2. On the machine where you ran the install for the repositories files, run the AppMetrics uninstall. See "Uninstall Procedure for AppMetrics and its AppManager Components" on Page C-13.

Scenario for Uninstalling AppMetrics and Its AppManager 3.x Repository Files, Including the Discovered Managed Objects

You cannot use the Tree View in the AppManager 3.x Console to delete discovered managed objects from any managed client. So if you want to completely uninstall the AppMetrics Repository integration from AppManager, including the managed objects discovered for AppMetrics, you could delete the managed client(s) containing the discovered objects, and then re-add the client(s) after the AppMetrics uninstall. This is a drastic measure because you will lose all objects on the deleted clients. As a result, it is not recommended. However, if you must remove the discovered managed objects for AppMetrics, here are the tasks:

1. In the AppManager Tree View, stop and delete all jobs on the managed client(s) where AppMetrics was discovered.

2. (Optional) Back up any data for managed objects that you want to use after you remove the managed client(s).

3. In the AppManager Tree View, remove the managed client(s) containing AppMetrics managed objects. This erases all data for the managed objects from the database.

4. On the machine where you ran the install for the repositories files, run the AppMetrics uninstall. See "Uninstall Procedure for AppMetrics and its AppManager Components" on Page C-13.

5. In AppManager, re-add the managed client(s) that you removed in Step 3.

6. Re-discover the managed objects on the managed client(s).

Scenario for Uninstalling AppMetrics and its AppManager 4 Repository Files

To uninstall the AppMetrics Repository integration from AppManager 4, perform the following tasks:

1. In the AppMetrics Tree View, stop and delete all jobs on AppMetrics objects.

2. Delete the AppMetrics managed objects from the AppManager 4 Console Tree View. As a result, AppManager will remove the AppMetrics view tab automatically.

3. On the machine where you ran the install for the repositories files, run the AppMetrics uninstall. See "Uninstall Procedure for AppMetrics and its AppManager Components" on Page C-13.

Uninstall Procedure for AppMetrics and its AppManager Components

This procedure should only be performed within the context of one of the previous three uninstall sections. Refer to those sections first, choose the section that best describes your preferred uninstall scenario, and then perform the tasks within it. Those tasks will eventually refer you to this procedure.

Note: Prior to uninstalling AppMetrics, you should stop all monitors in AppMetrics.

1. Log in to the machine with Administrators and AppMetrics Administrators privileges.

2. Click **Start → Settings → Control Panel**.

3. In the **Control Panel** window, double-click **Add/Remove Programs**.

4. In the **Modify, Upgrade or Repair, or Remove the program** screen, select **Remove**, and then click **Next**.

5. In the **Confirm uninstall** prompt, click **OK**.

    This opens the **NetIQ AppManager Installation Information** screen (Figure C-6):

Figure C-6   NetIQ AppManager Installation Information Screen

In this screen, you specify all the repositories containing AppMetrics data to be removed.

6. Click **Add**.

7. Complete the information in the **Add Repository to List** area.

    - **Server** is the SQL Server machine where the repository database resides. This may be the local machine or another accessible machine on your network.

    - **Database** is the name of the AppManager repository.

- **User ID** and **Password** correspond to a database administrative account for the repository.

8. Click **OK**.

9. To add another repository from which you want to remove AppMetrics data, repeat Steps 6 through 8.

10. After adding all the repositories to the list, click **Next**.
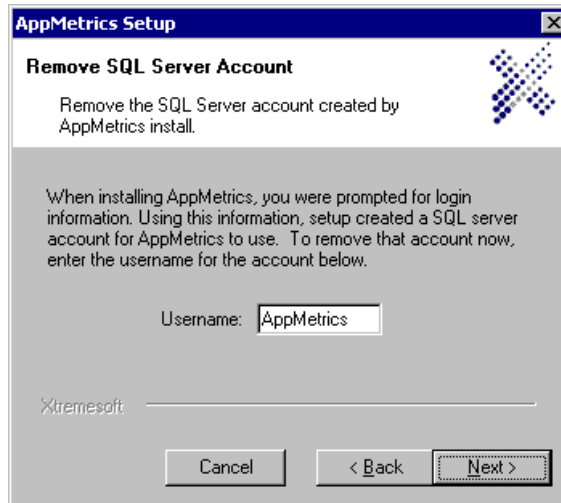
    This opens the **SQL Server Administrator access** screen (Figure C-7):
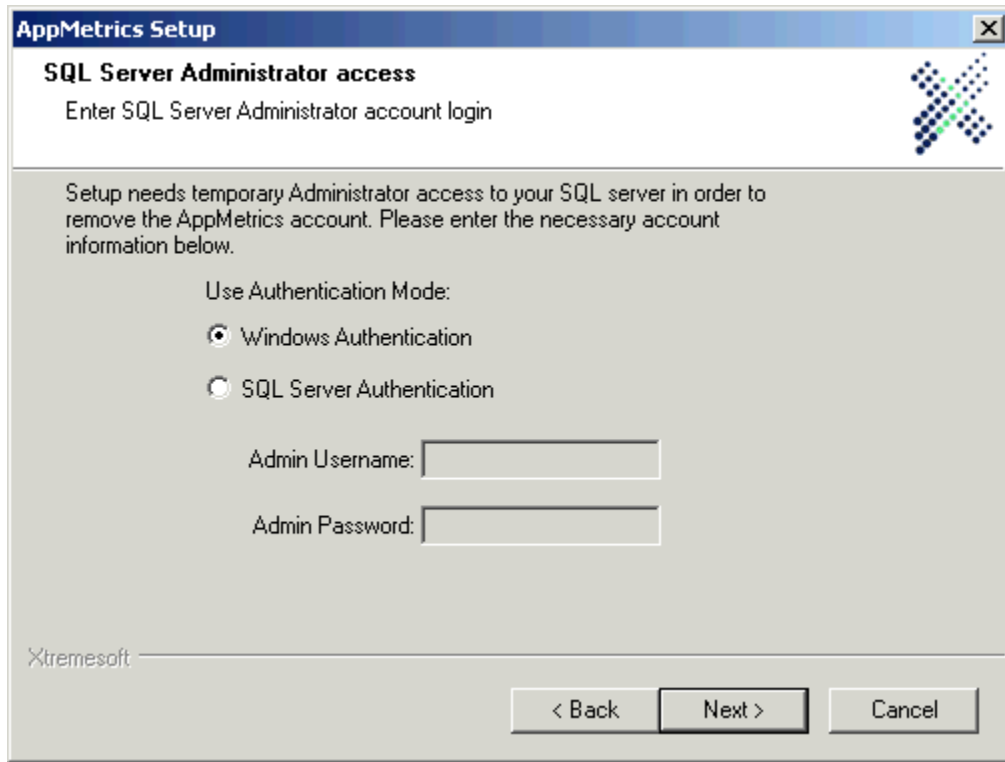
Figure C-7   SQL Server Administrator access Screen

11. Enter the username and password for a SQL Server administrator account or select Windows Authentication. Click **Next**.

    This opens the **Remove SQL Server Account** screen (Figure C-8):

Figure C-8   Remove SQL Server Account Screen

12. Make sure the SQL Server account that was created during the install is specified, and then click **Next**.

The **Setup Status** screen appears. During this screen, the install program removes the AppMetrics files from the machine and from the specified repositories.

It may also prompt you to remove the AppMetrics Administrators group and the SQL Server accounts. Click the preferred option in each prompt.

Soon afterwards, the **Setup Status** screen closes.

Using AppMetrics for Transactions and NetIQ AppManager

After you install AppMetrics on your machine(s), you should create and configure a monitor for your packages/applications. You can then run the monitor to generate metrics. Refer to Chapter 3 for more information.

Once your monitor has generated metrics, AppManager can begin retrieving the metrics from your monitor. This involves the following tasks:

1. Open an AppManager Console.

2. Log into a Repository for a Management Server that monitors the AppMetrics.

3. If this is a first-time connection to the Managed Client after an AppMetrics install, run an AppManager Discovery to locate the AppMetrics objects. (See the "Discovery_AppMetricsForTransactions" Knowledge Script on Page C-30.)

This creates an AppMetrics hierarchy in the Tree View of the AppManager Console. Additionally, if this is the first machine where AppMetrics objects have been discovered, AppManager will add a new AppMetrics View tab at the bottom of the Console.

4. Use one of the AppMetrics data-collection Knowledge Scripts to obtain the data from your monitor.

Refer to the "Knowledge Script Reference" section on Page C-16, which provides details about the AppMetrics monitoring scripts. The data-collection Knowledge Scripts will send data streams from AppMetrics to AppManager. This data can then be incorporated into graphs and reports. The "AppManager 3.x Report Reference" section on Page C-32 provides further details on using AppMetrics reports in Report Manager.

Levels of Monitoring and the Amount of Generated Data

Using the AppMetrics integration with AppManager requires careful planning and consideration of an AppManager network. Depending on the level of monitoring you select, the AppMetrics monitoring solution can provide a considerable amount of data. The data-collection Knowledge Scripts provided with the AppMetrics integration should be employed with an awareness of its effects on both system and network performance. If you only wish to set benchmarks and monitor threshold exceptions, this can be done without collecting any data streams.

Before starting an AppManager job with one of these scripts, be aware of the number of data streams that will be returned across every object that you target. Also note that these data streams will be collected during every interval, based on the timing you select.

AppMetrics allows you to specify which packages or applications you wish to monitor. For each package or application, you can also enable method-level detail. Furthermore, you can collect average metrics over intervals or every event that occurs in the system. For more information on how much detail you can specify for a monitor, see Chapter 3: Application Monitors.

Every collected data stream has to be communicated from the AppManager Agent to the Management Server/Repository, and back to the Console. This increases network traffic and SQL Server load. Consult the NetIQ Work Smarter Guide, *Tuning Tips For The AppManager Operator Console*, for information regarding both the data capacity of an AppManager network and the methods for managing the traffic, such as configuring Agent machines to send data at regular intervals or scheduling site uploads.

Knowledge Script Reference

AppMetrics Knowledge Scripts can be located by clicking the AM4T for Transactions tab in the Knowledge Script pane, except for the "Discovery_AppMetricsForTransactions" Knowledge Script, which is found on the *Discovery* tab.

AM4T_Production_AllTransactionsMetrics

This script retrieves the metrics for the aggregate of all transactions. Optionally, you may choose to provide these metrics as AppManager data streams for graphing and reporting. You can also set benchmark numbers for certain metrics and receive notification when these benchmarks are exceeded.

Note: The interval for this Knowledge Script should be set higher than the reporting interval in the AppMetrics monitor. This prevents AppManager from collecting duplicate values from within the same AppMetrics interval.

Target Managed Objects

AppMetrics Production AllTransactions Objects

Parameters

Parameter	Description
Event? (y/n)	Indicates whether or not to generate an event when an execution error or a benchmark exception occurs. **Note:** In AppManager 3.4, a console event notification (triggered by selecting AKP_NULL on the Action tab of the Knowledge Script's properties) will not occur if its underlying event name is longer than 128 characters, which is the maximum limit in AppManager 3.4.

Parameter	Description
	Each event name from this Knowledge Script is made up in part of the machine name and monitor name. So if these names contribute to an event name that is longer than 128 characters, then the resulting console event notification will not occur.
Collect Data? (y/n)	If selected, the data collected by the script will be delivered to the AppManager graphing and reporting tools.
Event Severity on Execution Error	(1-40, Severity Level) The event level for an error during the script's execution.
Event Severity on Exceeding Benchmark Warning Level	(1-40, Severity Level) The event level when a warning benchmark is exceeded.
Event Severity on Exceeding Benchmark Error Level	(1-40, Severity Level) The event level when an error benchmark is exceeded.
Notify threshold exceptions for Active count?	(Checkbox) Controls the monitoring of thresholds for the Active count.
Warning Benchmark	The warning threshold for the active count.
Error Benchmark	The error threshold for the active count.
Notify threshold exceptions for Maximum Active count?	(Checkbox) Controls the monitoring of thresholds for the Maximum Active count.
Warning Benchmark	The warning threshold for the Maximum Active count.
Error Benchmark	The error threshold for the Maximum Active count.
Notify threshold exceptions for Minimum Active count?	(Checkbox) Controls the monitoring of thresholds for the Minimum Active count.
Warning Benchmark	The warning threshold for the Minimum Active count.
Error Benchmark	The error threshold for the Minimum Active count.
Notify threshold exceptions for Interval Begin count?	(Checkbox) Controls the monitoring of thresholds for the Interval Begin count.
Warning Benchmark	The warning threshold for the Interval Begin count.
Error Benchmark	The error threshold for the Interval Begin count.
Notify threshold exceptions for Interval Maximum Active count?	(Checkbox) Controls the monitoring of thresholds for the Interval Maximum Active count.
Warning Benchmark	The warning threshold for the Interval Maximum Active count.
Error Benchmark	The error threshold for the Interval Maximum count.
Notify threshold exceptions for Interval Minimum Active count?	(Checkbox) Controls the monitoring of thresholds for the Interval Minimum Active count.
Warning Benchmark	The warning threshold for the Interval Minimum Active count.
Error Benchmark	The error threshold for the Interval Minimum count.

Parameter	Description
Notify threshold exceptions for Interval Started count?	(Checkbox) Controls the monitoring of thresholds for the Interval Started count.
Warning Benchmark	The warning threshold for the Interval Started count.
Error Benchmark	The error threshold for the Interval Started count.
Notify threshold exceptions for Interval Completed count?	(Checkbox) Controls the monitoring of thresholds for the Interval Completed count.
Warning Benchmark	The warning threshold for the Interval Completed count.
Error Benchmark	The error threshold for the Interval Completed count.
Notify threshold exceptions for Interval Aborted count?	(Checkbox) Controls the monitoring of thresholds for the Interval Aborted count.
Warning Benchmark	The warning threshold for the Interval Aborted count.
Error Benchmark	The error threshold for the Interval Aborted count.
Notify threshold exceptions for Interval Started Rate?	(Checkbox) Controls the monitoring of thresholds for the Interval Started Rate.
Warning Benchmark (Scale by 10,000) *	The warning threshold for the Interval Started Rate.
Error Benchmark (Scale by 10,000) *	The error threshold for the Interval Started Rate.
Notify threshold exceptions for Interval Completed Rate?	(Checkbox) Controls the monitoring of thresholds for the Interval Completed Rate.
Warning Benchmark (Scale by 10,000) *	The warning threshold for the Interval Completed Rate.
Error Benchmark (Scale by 10,000) *	The error threshold for the Interval Completed Rate.
Notify threshold exceptions for Interval Aborted Rate?	(Checkbox) Controls the monitoring of thresholds for the Interval Aborted Rate.
Warning Benchmark (Scale by 10,000) *	The warning threshold for the Interval Aborted Rate.
Error Benchmark (Scale by 10,000) *	The error threshold for the Interval Aborted Rate.

* The Knowledge Script divides your specified value by 10,000. So for example, if you want to set a threshold of 4.5012 transactions per second, enter a value of "45012" in the parameter.

Data Streams

The data streams contain a header with the format "Monitor Name: AllTransactions", followed by one of the following:

Note: The "interval" in the following metrics means the interval of the AppMetrics monitor, not the reporting interval chosen for the Knowledge Script in NetIQ.

":Active" - Currently active instances

":IntAborted" - Number of instances aborted in the last interval

":IntAbortedRate" - Number of instances aborted per second in the last interval

":IntBeginActive" - Number of instances active at the start of the last interval

":IntCompleted" - Number of instances completed in the last interval

":IntCompletedRate" - Number of instances completed per second in the last interval

":IntMaxActive" - Maximum number of concurrently active instances in the last interval

":IntMinActive" - Minimum number of concurrently active instances in the last interval

":IntStarted" - Number of instances started in the last interval

":IntStartedRate" - Number of instances started per second in the last interval

":MaxActive" - Maximum number of concurrently active instances monitored this session

":MinActive" - Minimum number of concurrently active instances monitored this session

AM4T_Production_ComponentMetrics

This script retrieves metrics for one or more Components. Optionally, you may choose to provide these metrics as AppManager data streams for graphing and reporting. You can also set benchmark numbers for certain metrics and opt to receive notification any time a monitored Component exceeds one of those benchmarks.

Note: The interval for this Knowledge Script should be set higher than the reporting interval in the AppMetrics monitor. This prevents AppManager from collecting duplicate values from within the same AppMetrics interval.

Target Managed Objects

AppMetrics Production Component Objects

Parameters

Parameter	Description
Event? (y/n)	Indicates whether or not to generate an event when an execution error or a benchmark exception occurs.
	Note: In AppManager 3.4, a console event notification (triggered by selecting AKP_NULL on the Action tab of the Knowledge Script's properties) will not occur if its underlying event name is longer than 128 characters, which is the maximum limit in AppManager 3.4.
	Each event name from this Knowledge Script consists in part of the machine name, monitor name, and component name. So if these names contribute to an event name that is longer than 128 characters, then the resulting console event notification will not occur.
Collect Data? (y/n)	If selected, the data collected by the script will be delivered to the AppManager graphing and reporting tools.
Event Severity on Execution Error	(1-40, Severity Level) The event level for an error during the script's execution.
Event Severity on Exceeding Benchmark Warning Level	(1-40, Severity Level) The event level when a warning benchmark is exceeded.
Event Severity on Exceeding Benchmark Error Level	(1-40, Severity Level) The event level when an error benchmark is exceeded.
Notify threshold exceptions for Active count?	(Checkbox) Controls the monitoring of thresholds for the Active count.
Warning Benchmark	The warning threshold for the active count.
Error Benchmark	The error threshold for the active count.
Notify threshold exceptions for Maximum Active count?	(Checkbox) Controls the monitoring of thresholds for the Maximum Active count.
Warning Benchmark	The warning threshold for the Maximum Active count.
Error Benchmark	The error threshold for the Maximum Active count.

Parameter	Description
Notify threshold exceptions for Minimum Active count?	(Checkbox) Controls the monitoring of thresholds for the Minimum Active count.
Warning Benchmark	The warning threshold for the Minimum Active count.
Error Benchmark	The error threshold for the Minimum Active count.
Notify threshold exceptions for Interval Begin count?	(Checkbox) Controls the monitoring of thresholds for the Interval Begin count.
Warning Benchmark	The warning threshold for the Interval Begin count.
Error Benchmark	The error threshold for the Interval Begin count.
Notify threshold exceptions for Interval Maximum Active count?	(Checkbox) Controls the monitoring of thresholds for the Interval Maximum Active count.
Warning Benchmark	The warning threshold for the Interval Maximum Active count.
Error Benchmark	The error threshold for the Interval Maximum count.
Notify threshold exceptions for Interval Minimum Active count?	(Checkbox) Controls the monitoring of thresholds for the Interval Minimum Active count.
Warning Benchmark	The warning threshold for the Interval Minimum Active count.
Error Benchmark	The error threshold for the Interval Minimum count.
Notify threshold exceptions for Interval Started count?	(Checkbox) Controls the monitoring of thresholds for the Interval Started count.
Warning Benchmark	The warning threshold for the Interval Started count.
Error Benchmark	The error threshold for the Interval Started count.
Notify threshold exceptions for Interval Completed count?	(Checkbox) Controls the monitoring of thresholds for the Interval Completed count.
Warning Benchmark	The warning threshold for the Interval Completed count.
Error Benchmark	The error threshold for the Interval Completed count.
Notify threshold exceptions for Interval Aborted count?	(Checkbox) Controls the monitoring of thresholds for the Interval Aborted count.
Warning Benchmark	The warning threshold for the Interval Aborted count.
Error Benchmark	The error threshold for the Interval Aborted count.
Notify threshold exceptions for Average Duration on the interval?	(Checkbox) Controls the monitoring of thresholds for the Average Duration on the interval.
Warning Benchmark	The warning threshold for the Average Duration on the Interval.
Error Benchmark	The error threshold for the Average Duration on the Interval.

Parameter	Description
Notify threshold exceptions for Minimum Duration on the interval?	(Checkbox) Controls the monitoring of thresholds for the Minimum Duration on the interval.
Warning Benchmark	The warning threshold for the Minimum Duration on the interval.
Error Benchmark	The error threshold for the Minimum Duration on the interval.
Notify threshold exceptions for Maximum Duration on the Interval?	(Checkbox) Controls the monitoring of thresholds for the Maximum Duration on the interval.
Warning Benchmark	The warning threshold for the Maximum Duration on the Interval.
Error Benchmark	The error threshold for the Maximum Duration on the interval.
Notify threshold exceptions for Interval Started Rate?	(Checkbox) Controls the monitoring of thresholds for the Interval Started Rate.
Warning Benchmark (Scale by 10,000) *	The warning threshold for the Interval Started Rate.
Error Benchmark (Scale by 10,000) *	The error threshold for the Interval Started Rate.
Notify threshold exceptions for Interval Completed Rate?	(Checkbox) Controls the monitoring of thresholds for the Interval Completed Rate.
Warning Benchmark (Scale by 10,000) *	The warning threshold for the Interval Completed Rate.
Error Benchmark (Scale by 10,000) *	The error threshold for the Interval Completed Rate.
Notify threshold exceptions for Interval Aborted Rate?	(Checkbox) Controls the monitoring of thresholds for the Interval Aborted Rate.
Warning Benchmark (Scale by 10,000) *	The warning threshold for the Interval Aborted Rate.
Error Benchmark (Scale by 10,000) *	The error threshold for the Interval Aborted Rate.

* The Knowledge Script divides your specified value by 10,000. So for example, if you want to set a threshold of 4.5012 components per second, enter a value of "45012" in the parameter.

Data Streams

The data streams contain a header with the format "Monitor Name: Package Name: Component Name", followed by one of the following:

Note: The "interval" in the following metrics means the interval of the AppMetrics monitor, not the reporting interval chosen for the Knowledge Script in NetIQ.

":Active" - Currently active instances

":IntAborted" - Number of instances aborted in the last interval

":IntAbortedRate" - Number of instances aborted per second in the last interval

":IntAvgDuration" - Average duration (milliseconds) of an instance in the last interval

":IntBeginActive" - Number of instances active at the start of the last interval

":IntCompleted" - Number of instances completed in the last interval

":IntCompletedRate" - Number of instances completed per second in the last interval

":IntMaxActive" - Maximum number of concurrently active instances in the last interval

":IntMaxDuration" - Maximum duration (milliseconds) of an instance in the last interval

":IntMinActive" - Minimum number of concurrently active instances in the last interval

":IntMinDuration" - Minimum duration (milliseconds) of an instance in the last interval

":IntStarted" - Number of instances started in the last interval

":IntStartedRate" - Number of instances started per second in the last interval

":MaxActive" - Maximum number of concurrently active instances monitored this session

":MinActive" - Minimum number of concurrently active instances monitored this session

AM4T_InstallAppMetrics

This script performs a silent InstallShield installation of AppMetrics to a managed client machine. A failed install will always generate an error notification.

Notes

- This script operates under the same restrictions as the AMAdmin_AgentInstall script. (See the NetIQ AppManager Knowledge Script Reference Guide) The AppManager Agent must be installed on the management server currently in use. Also, both Agent services (NetIQccm and NetIQmc) must be running as a user account with privileges and permission to access the target managed client. The managed client must be in the same domain as, or in a trusted domain to, the management server. If necessary, the setup of the Agent services on the management server may be temporarily changed to run under a trusted account, and then switched back when the installation is complete.

- For security purposes, delete all AM4T_InstallAppMetrics jobs after they finish running.

Target Managed Objects

Managed Client Machines

Parameters

Parameter	Description
Raise an event when installation completes successfully?	(y/n) If selected, and notification will be sent upon successful completion of the install.
Event severity level when installation is successful	(1-40, SevLevel) The event severity if notification is being sent on success.
User name for remote install	(Required) The user name for logging in to the managed client.
Password for remote install user	The password for logging in to the managed client.

Parameter	Description
Domain of remote install user	(Required) The domain of the above user account; can be machine name for local accounts.
Temporary directory on the remote computer	(Required) Directory for storing temporary install files on the managed client; use no spaces in path.
Path and filename of the AppMetrics silent install program	(Required) Full UNC path to the AppMetrics installation shell script (AppMetricsNetIQInstall.bat); located either on your CD-ROM, or a share on your network to which the setup files were copied.
Customer name for the installation	(Required) AppMetrics will be registered with this user name.
Company name for the installation	(Required) AppMetrics will be registered with this company name.
AppMetrics serial number	(Required) The serial number for your copy of AppMetrics.
Desired AppMetrics installation directory	The directory in which to install AppMetrics on the managed client.
Desired AppMetrics data files directory	The directory in which to store AppMetrics data files on the managed client.
AppMetrics setup type	(Dropdown list box, with selections "Agent" and "Manager") The role AppMetrics will perform on the machine.
Remote Access account (NT4), or Service RunAs account (W2K)	Windows user account needed for all setup types.
Remote/RunAs account password	The password for the Windows user account.
Remote/RunAs account domain	Windows account domain needed for all setup types.
Desired MSDE installation directory	Location for MSDE install if MSDE/MSSQL is not currently installed on the target machine.
Desired MSDE data files directory	Location for MSDE data files if MSDE/MSSQL is not currently installed on the target machine.
MSDE/MSSQL Administrative username	Current administrator username or desired new administrator username (if installing MSDE with AppMetrics). For all non-Agent installs.
MSDE/MSSQL Administrative password	Current password or desired new admin password (if installing MSDE with AppMetrics). For all non-Agent installs.
MSDE/MSSQL AppMetrics username	MSDE/MSSQL username for the AppMetrics account. For or all non-Agent installs.
MSDE/MSSQL AppMetrics password	The password for MSDE/MSSQL AppMetrics username. For all non-Agent installs.
Reboot when finished?	(y/n) Determines if the install will reboot the machine when the install process is complete.

AM4T_Production_ApplicationMetrics

This script retrieves metrics for one or more MTS packages/COM+ applications. Optionally, these metrics may be used to provide AppManager data streams for graphing and reporting.

Note: The interval for this Knowledge Script should be set higher than the reporting interval in the AppMetrics monitor. This prevents AppManager from collecting duplicate values from within the same AppMetrics interval.

Target Managed Objects

AppMetrics Production Application Objects

Parameters

Parameter	Description
Event? (y/n)	Indicates whether or not to generate an event when an execution error or a benchmark exception occurs.
	Note: In AppManager 3.4, a console event notification (triggered by selecting AKP_NULL on the Action tab of the Knowledge Script's properties) will not occur if its underlying event name is longer than 128 characters, which is the maximum limit in AppManager 3.4.
	Each event name from this Knowledge Script consists in part of the machine name, monitor name, and package name. So if these names contribute to an event name that is longer than 128 characters, then the resulting console event notification will not occur.
Collect Data? (y/n)	If selected, the data collected by the script will be delivered to the AppManager graphing and reporting tools.
Event Severity on Error	(1-40, Severity Level) The event level for an error during the script's execution.

Data Streams

The data streams contain a header with the format "Monitor Name: Package Name", followed by one of the following:

Note: The "interval" in the following metrics means the interval of the AppMetrics monitor, not the reporting interval chosen for the Knowledge Script in NetIQ.

":Activations" - Number of packages startups monitored this session

":Crashes" - Number of abnormal package shutdowns monitored this session

":IntLost" - Number of MTS events lost during the monitor's interval

":Lost" - Number of MTS events lost during this session

":MTSThreads" - Size of the STA thread pool

":PageFaultsPerSec" - Number of page faults per second

":PercentCPU" - Percent CPU usage by the server package process

":PrevIntLost" - Number of MTS events lost during the monitor's last interval

":ShutdownReason" - Package shutdown status (0 - normal, 1 - abnormal, 2 - manual)

":Shutdowns" - Number of normal packages shutdowns monitored this session

":State" - Package status (0 - stopped, 1 - started)

":WorkingSet"- Package process memory usage (Kb)

AM4T_Production_TransactionMetrics

This script retrieves metrics for one or more transaction. Optionally, you may choose to provide these metrics as AppManager data streams for graphing and reporting. You can also set benchmark numbers for certain metrics and opt to receive notification any time a monitored transaction exceeds one of those benchmarks.

Note: The interval for this Knowledge Script should be set higher than the reporting interval in the AppMetrics monitor. This prevents AppManager from collecting duplicate values from within the same AppMetrics interval.

Target Managed Objects

AppMetrics Production Transactions Objects

Parameters

Parameter	Description
Event? (y/n)	Indicates whether or not to generate an event when an execution error or a benchmark exception occurs.
	Note: In AppManager 3.4, a console event notification (triggered by selecting AKP_NULL on the Action tab of the Knowledge Script's properties) will not occur if its underlying event name is longer than 128 characters, which is the maximum limit in AppManager 3.4.
	Each event name from this Knowledge Script consists in part of the machine name, monitor name, and transaction name. So if these names contribute to an event name that is longer than 128 characters, then the resulting console event notification will not occur.
Collect Data? (y/n)	If selected, the data collected by the script will be delivered to the AppManager graphing and reporting tools.
Event Severity on Execution Error	(1-40, Severity Level) The event level for an error during the script's execution.
Event Severity on Exceeding Benchmark Warning Level	(1-40, Severity Level) The event level when a warning benchmark is exceeded.
Event Severity on Exceeding Benchmark Error Level	(1-40, Severity Level) The event level when an error benchmark is exceeded.
Notify threshold exceptions for Active count?	(Checkbox) Controls the monitoring of thresholds for the Active count.
Warning Benchmark	The warning threshold for the active count.
Error Benchmark	The error threshold for the active count.

Parameter	Description
Notify threshold exceptions for Maximum Active count?	(Checkbox) Controls the monitoring of thresholds for the Maximum Active count.
Warning Benchmark	The warning threshold for the Maximum Active count.
Error Benchmark	The error threshold for the Maximum Active count.
Notify threshold exceptions for Minimum Active count?	(Checkbox) Controls the monitoring of thresholds for the Minimum Active count.
Warning Benchmark	The warning threshold for the Minimum Active count.
Error Benchmark	The error threshold for the Minimum Active count.
Notify threshold exceptions for Interval Begin count?	(Checkbox) Controls the monitoring of thresholds for the Interval Begin count.
Warning Benchmark	The warning threshold for the Interval Begin count.
Error Benchmark	The error threshold for the Interval Begin count.
Notify threshold exceptions for Interval Maximum Active count?	(Checkbox) Controls the monitoring of thresholds for the Interval Maximum Active count.
Warning Benchmark	The warning threshold for the Interval Maximum Active count.
Error Benchmark	The error threshold for the Interval Maximum count.
Notify threshold exceptions for Interval Minimum Active count?	(Checkbox) Controls the monitoring of thresholds for the Interval Minimum Active count.
Warning Benchmark	The warning threshold for the Interval Minimum Active count.
Error Benchmark	The error threshold for the Interval Minimum count.
Notify threshold exceptions for Interval Started count?	(Checkbox) Controls the monitoring of thresholds for the Interval Started count.
Warning Benchmark	The warning threshold for the Interval Started count.
Error Benchmark	The error threshold for the Interval Started count.
Notify threshold exceptions for Interval Completed count?	(Checkbox) Controls the monitoring of thresholds for the Interval Completed count.
Warning Benchmark	The warning threshold for the Interval Completed count.
Error Benchmark	The error threshold for the Interval Completed count.
Notify threshold exceptions for Interval Aborted count?	(Checkbox) Controls the monitoring of thresholds for the Interval Aborted count.
Warning Benchmark	The warning threshold for the Interval Aborted count.
Error Benchmark	The error threshold for the Interval Aborted count.

Parameter	Description
Notify threshold exceptions for Average Duration on the interval?	(Checkbox) Controls the monitoring of thresholds for the Average Duration on the interval.
Warning Benchmark	The warning threshold for the Average Duration on the Interval.
Error Benchmark	The error threshold for the Average Duration on the Interval.
Notify threshold exceptions for Minimum Duration on the interval?	(Checkbox) Controls the monitoring of thresholds for the Minimum Duration on the interval.
Warning Benchmark	The warning threshold for the Minimum Duration on the interval.
Error Benchmark	The error threshold for the Minimum Duration on the interval.
Notify threshold exceptions for Maximum Duration on the Interval?	(Checkbox) Controls the monitoring of thresholds for the Maximum Duration on the interval.
Warning Benchmark	The warning threshold for the Maximum Duration on the Interval.
Error Benchmark	The error threshold for the Maximum Duration on the interval.
Notify threshold exceptions for Average Aggregate DTC Duration on the interval?	(Checkbox) Controls threshold monitoring for the average aggregate duration that an instance and its sub-objects spent using DTC transaction resources. The duration is cumulative for both the instance and its sub-objects. It represents durations for instances that completed during the most recently completed interval.
Warning Benchmark	The warning threshold for the Average Aggregate DTC Duration on the interval.
Error Benchmark	The error threshold for the Average Aggregate DTC Duration on the interval.
Notify threshold exceptions for Interval Started Rate?	(Checkbox) Controls the monitoring of thresholds for the Interval Started Rate.
Warning Benchmark (Scale by 10,000) *	The warning threshold for the Interval Started Rate.
Error Benchmark (Scale by 10,000) *	The error threshold for the Interval Started Rate.
Notify threshold exceptions for Interval Completed Rate?	(Checkbox) Controls the monitoring of thresholds for the Interval Completed Rate.
Warning Benchmark (Scale by 10,000) *	The warning threshold for the Interval Completed Rate.
Error Benchmark (Scale by 10,000) *	The error threshold for the Interval Completed Rate.

Parameter	Description
Notify threshold exceptions for Interval Aborted Rate?	(Checkbox) Controls the monitoring of thresholds for the Interval Aborted Rate.
Warning Benchmark (Scale by 10,000) *	The warning threshold for the Interval Aborted Rate.
Error Benchmark (Scale by 10,000) *	The error threshold for the Interval Aborted Rate.

* The Knowledge Script divides your specified value by 10,000. So for example, if you want to set a threshold of 4.5012 transactions per second, enter a value of "45012" in the parameter.

Data Streams

The data streams contain a header with the format "Monitor Name: Transaction Name", followed by one of the following:

Note: The "interval" in the following metrics means the interval of the AppMetrics monitor, not the reporting interval chosen for the Knowledge Script in NetIQ.

":Active" - Currently active instances

":IntAborted" - Number of instances aborted in the last interval

":IntAbortedRate" - Number of instances aborted per second in the last interval

":IntAvgDuration" - Average duration (milliseconds) of an instance in the last interval

":IntAvgDTCDuration" - Average aggregate duration that an instance and its sub-objects spent using DTC transaction resources in the last interval

":IntMinDTCDuration" - Duration that the shortest transaction instance and its sub-objects spent using DTC transaction resources in the last interval

":IntMaxDTCDuration" - Duration that the longest transaction instance and its sub-objects spent using DTC transaction resources in the last interval

":IntBeginActive" - Number of instances active at the start of the last interval

":IntCompleted" - Number of instances completed in the last interval

":IntCompletedRate" - Number of instances completed per second in the last interval

":IntMaxActive" - Maximum number of concurrently active instances in the last interval

":IntMaxDuration" - Maximum duration (milliseconds) of an instance in the last interval

":IntMinActive" - Minimum number of concurrently active instances in the last interval

":IntMinDuration" - Minimum duration (milliseconds) of an instance in the last interval

":IntStarted" - Number of instances started in the last interval

":IntStartedRate" - Number of instances started per second in the last interval

":MaxActive" - Maximum number of concurrently active instances monitored this session

":MinActive" - Minimum number of concurrently active instances monitored this session

Am4T_UninstallAppMetrics

This script performs a silent InstallShield uninstall of AppMetrics on a managed client machine. A failed uninstall will always generate an error notification.

Notes

- This script operates under the same restrictions as the AMAdmin_AgentInstall script. (See the NetIQ AppManager Knowledge Script Reference Guide) The AppManager Agent must be installed on the management server currently in use.

- Both Agent services (NetIQccm and NetIQmc) must be running as a user account with privileges and permission to access the target managed client.

- The managed client must be in the same domain as, or in a trusted domain to, the management server. If necessary, the setup of the Agent services on the management server may be temporarily changed to run under a trusted account, and then switched back when the uninstall is complete.

- For security purposes, delete all AM4T_UninstallAppMetrics jobs after they finish running.

Target Managed Objects

Managed Client Machines

Parameters

Parameter	Description
Raise an event when installation completes successfully?	(y/n) If selected, and notification will be sent upon successful completion of the install.
Event severity level when installation is successful	(1-40, SevLevel) The event severity if notification is being sent on success.
User name for remote install	The user name for logging in to the managed client.
Password for remote install user	The password for logging in to the managed client.
Domain of remote install user	The domain of the above user account.
Temporary directory on the remote computer	Directory for storing temporary install files on the managed client.
MSDE/MSSQL Administrative username	Current database administrator username.
MSDE/MSSQL Administrative password	Current password for the database administrator username.
MSDE/MSSQL AppMetrics username	MSDE/MSSQL username for the AppMetrics account.

Discovery_AppMetricsForTransactions

This script performs the AppManager discovery process for AppMetrics managed objects on a managed client machine. It creates the AppMetrics hierarchy in the console Tree view pane for a managed client. You should run this script before you run any of the other AppMetrics monitoring Knowledge Scripts.

When run on the managed client machine, the Discovery_AppMetrics script discovers AppMetrics monitors that derive from the MTS and COM+ Production Templates. The script adds these monitors to the Console Tree hierarchy, where they become managed objects under the managed client machine. Additional objects are the packages/applications, components, and transactions that have been monitored.

After the initial discovery, if any new packages/applications, components, and transactions are to be monitored, re-run the discovery script to add these objects to the AppMetrics hierarchy in AppManager.

Notes

- This Knowledge Script is located in the Discovery tab in the Knowledge Script pane.

- The Discovery will fail if any of the application, component, or transaction names from AppMetrics is longer than 128 characters. AppManager does not handle names longer than 128 characters. You can also use the parameters to constrain name length.

Target Managed Objects

Managed Client Machines

Parameters

Parameter	Description
Event for successful discovery? (y/n)	If y, an event will be generated when the discovery is successful; failure will always generate an event.
Event Level when Discovery is OK	(1-40 SevLevel) Event severity upon successful discovery.
Event Level when Discovery Failed	(1-40 SevLevel) Event severity upon failed discovery.
Event Level when Discovery is NOT Applicable	(1-40 SevLevel) Event severity if managed client is not an AppMetrics server or does not have AppMetrics extension DLL.

AppManager 3.x Report Reference

You can generate three types of reports from the AppMetrics NetIQ data: "Top Ten" reports, Minimum/Maximum/Average reports over various time periods, and Charts of Metrics By Time Interval reports.

To run a report, first run the Knowledge Script associated with it. In particular, you can run any of the following Knowledge Scripts to generate data for reports:

- AppMetrics_AllTransactionsMetrics

- AppMetrics_ComponentMetrics

- AppMetrics_PackageMetrics

- AppMetrics_TransactionMetrics

The Keyword column in the Report Manager indicates which script is associated with each report.

Once the Knowledge Script has collected some data, run the report from the Report Manager. To run a report using its default parameters, just double-click it. To change one or more parameters before running the report, right-click the report, select Properties, edit the parameter(s) you want to change, and click Preview.

Below is a description of the reports and how the parameters can be used to filter the report. To find out more about how to use the parameters, right-click a report and choose "Help on Selected Report."

"Top Ten" Reports

There are six "Top Ten" reports. The Component reports show the top ten Components by average starts, aborts, and duration. The Transaction reports show the top ten transaction types by average starts, aborts, and duration. When previewing the report, all components/transactions in the top ten show up in the tree control on the left.

Each of the six reports has four parameters:

KSName   Set to AppMetrics_ComponentMetrics or AppMetrics_TransactionMetrics. It shouldn't be altered.

LegendName   Filters the report by legend. By default, it is set to filter by metric, but you may change it to filter by Package, Component, or even Monitor.

MachineList The MachineList defaults to "ALL", however it may be changed to a list of machines of interest separated by commas.

ReportType   The ReportType defaults to "Monthly", meaning only the last 30 days of data is included in the report. The Report Type may be changed to "Annually", "Quarterly", "Weekly", "Daily", or "Hourly".

Minimum/Maximum/Average Reports

There are 11 Minimum/Maximum/Average reports. These reports show the minimum, maximum, and average of different metrics over various time periods. Each report contains five sub-reports, each containing data for a specific time period:

- All time periods

- The last 30 days

- The last week

- The last 24 hours

- The last hour

The All Transactions reports show the minimum, maximum, and average of starts and aborts for all the transactions over the five time periods. The Component reports show the minimum, maximum, and average of starts, aborts, and durations for Components over the five time periods. The Transaction reports show the minimum, maximum, and average of starts, aborts, and duration of Transactions over these five time periods. The Package reports show the minimum, maximum, and average of STA Threads, Page Faults, and Percent CPU for Packages (Applications in Windows 2000) over the five time periods. When previewing the report, you can double-click on a sub-report to create a view specific to that sub-report.

Each of the 11 reports has 20 parameters – 4 parameters for each sub-report. The four parameters for each sub-report are:

KSName　　　Set to AppMetrics_AllTransactionsMetrics, AppMetrics_ComponentMetrics, AppMetrics_PackageMetrics, or AppMetrics_TransactionMetrics. It should not be altered.

LegendName　Filters the report by legend. By default, it is set to filter by metric, but you may change it to filter by Package, Component, or even Monitor.

MachineList　Defaults to "ALL", however it may be changed to a list of machines of interest separated by commas.

ReportType　Set differently for each sub-report. Possible values are "Annually", "Quarterly", "Monthly", "Weekly", "Daily", and "Hourly." Typically the setting for each sub-report requires no change.

Chart of Metrics By Time Interval

There are 15 charts of metrics by time interval reports. These reports show the average value of a metric over intervals in the specified day. All Transaction reports show Starts and Aborts for all the transactions during each interval. Component reports show Starts, Aborts, Duration, and Number Active for components in each interval. Transaction reports show Starts, Aborts, Duration, and Number Active for transactions in each interval. Package Reports show the STA Threads, Page Faults, Percent CPU, and Memory Usage for Packages (Applications in Windows 2000) in each interval. When previewing the report, all Components/Activities/Packages/Transactions for which data was collected show up in the tree control on the left and assist in navigating the report. Double-click the average number below the chart to show the values that make up the chart.

Each of the 15 reports has five parameters:

KSName   Set to AppMetrics_AllTransactionsMetrics, AppMetrics_ComponentMetrics, AppMetrics_PackageMetrics, or AppMetrics_TransactionMetrics. It shouldn't be altered.

LegendName Filters the report by legend. By default, it is set to filter by metric, but you may change it to filter by Package, Component, or even Monitor.

MachineList Defaults to "Default", but can be changed to another machine.

Intervalmin Defaults to 15, which means that data is grouped into 15-minute intervals and averaged over those 15 minutes before being displayed on the chart. This may be changed to a smaller value if not enough detail is available in the chart or to a larger value if the chart looks crowded (this is common due to a limitation of the reporting package).

day   Defaults to "Default", which corresponds to the current day. This may be changed to another day (such as "Jan 15 2000") to look at historical data.

With this report, it is common to change the Machine, Intervalmin, and day parameters before running the report. When the report is run, the header information indicates the number of minutes in the interval, the data-collection date, and the machine on which the data was collected.