

# **PC-Duo Diagnostics User Manual**

Copyright © MetaQuest Software Inc. and Vector Networks Limited.

The information in this document is subject to change without notice and should not be construed as a commitment by MetaQuest Software Inc. or Vector Networks Limited.

MetaQuest Software Inc. and Vector Networks Limited assume no responsibility for errors in this document.

The software described in this document is supplied under a license and may be used or copied only in accordance with the terms of such license.

MetaQuest is a trademark of MetaQuest Software Inc. PC-Duo, and its logos, are trademarks of Vector Networks Limited. All other trademarks are the property of their respective owners.

# Contents

<b>Chapter 1</b>	
<b>Getting Started</b> .....	1
Protecting Applications .....	1
Performing Change Analysis .....	1
Diagnostics .....	1
Using the Diagnostics Console .....	2
Setting Up Diagnostics .....	5
QuickStart .....	7
<b>Chapter 2</b>	
<b>Protecting and Restoring Applications</b> .....	9
Profiling Applications .....	9
Protecting Applications .....	13
Restoring Applications to Working Order .....	14
<b>Chapter 3</b>	
<b>Performing Change Analysis</b> .....	15
Manually Building a Profile .....	15
Auditing PCs .....	16
Viewing Audit Reports .....	17
Comparing Audit Reports .....	18
Printing Audit and Diagnostic Reports .....	20
<b>Chapter 4</b>	
<b>Collecting Information</b> .....	21
Defining Variables .....	21
Using Regular Expressions .....	26
System Resources .....	27
Auditing Files .....	27
Auditing ActiveX Controls .....	29
Auditing Registry Keys and Entries .....	29
Auditing Shortcuts .....	30
Copying Files .....	30
Auditing with Windows Management Instrumentation .....	31
Auditing Database Information .....	33
Collecting Diagnostics for IIS .....	35

Collecting Security Information .....	35
<b>Chapter 5</b>	
<b>Customizing Application Protection .....</b>	<b>37</b>
About Repair Rules .....	37
Customizing Repair Rules .....	37
Generating Repair Rules .....	37
Editing Repair Rules .....	38
Defining Conditions .....	38
Defining Actions .....	38
Setting Attribute Values .....	40
Locking Customized Repair Rules .....	41
Self-Repair Preferences .....	41
<b>Chapter 6</b>	
<b>Scheduling Jobs .....</b>	<b>43</b>
Defining Jobs .....	43
Running Jobs .....	44
Checking the Status of Jobs .....	44
Scheduling Jobs .....	44
<b>Chapter 7</b>	
<b>Requests .....</b>	<b>45</b>
Working with Requests .....	45
Troubleshooting Pending Requests .....	45
<b>Chapter 8</b>	
<b>Configuring .....</b>	<b>47</b>
Moving the Support Site .....	47
The Support Site User Account .....	47
Event Logging .....	48
Maintenance .....	48
Licensing .....	48
<b>Technical Support .....</b>	<b>51</b>
<b>Index .....</b>	<b>53</b>

# Chapter 1: Getting Started

Diagnostics allows you to protect and restore applications by taking snapshots of the applications on your networked PCs. From small utilities to business-critical applications, you can protect any number of applications across your entire network.

Diagnostics also provides change analysis capabilities to help determine root causes. By comparing application and PC settings against a baseline or at different points in time, you can quickly identify and correct the configuration changes that cause problems.

## Protecting Applications

To protect an application, you first build an application profile that describes a working configuration of the application: files, registry entries, ActiveX controls, self-registered files (DLLs), shortcuts, and environment variables.

After you have a profile, you can then protect the application on any computer in your network. When you protect an application, Diagnostics takes a snapshot of the application configuration on the machine. The profile drives this process, because it specifies what items make up the application configuration.

The snapshot contains everything needed to restore the application to working order, including repair rules for detecting and fixing problems, and an archive of application files.

## Performing Change Analysis

Change analysis is a basic technique for troubleshooting system and application problems. It is the process of tracking down configuration changes on a computer.

With Diagnostics, you can build profiles to collect application diagnostics and system configuration information such as services and printers, audit computers, and then analyze the collected diagnostic data. Diagnostics automatically compares application or system settings against a baseline, at different points in time, or on different computers. This allows you to quickly identify and correct the changes that caused the problem.

## Diagnostics

Diagnostics consists of three components: a central, administrative console, agents that run on remote computers, and a shared data folder called the Support Site.

### Diagnostics Console

You use the Diagnostics Console to profile, protect, and audit applications, and to diagnose and fix problems.

The Diagnostics Console is a Microsoft Management Console (MMC) snap-in that you can run as a standalone application.

MMC is a feature of the Windows 2000, NT, and XP operating systems, but can also run on the Windows 95, 98, and Me operating systems.

You can add the Diagnostics snap-in to other MMC consoles.

## Diagnostics Agents

Diagnostics Agents are installed on each computer on the network, and are responsible for auditing and protecting the computers.

## Support Site

Support Site is a shared folder that has the following functionality:

- It enables peer-to-peer communication between the consoles and agents.
- It stores the public profiles, all audit reports, and the licensing information.
- It includes the setup programs for consoles and agents. After the first Diagnostics Console is installed and configured, all other copies of the console are installed from Support Site.

## Using the Diagnostics Console

The Diagnostics Console consists of a window divided into two panes. The left pane contains the console tree, which shows the items available in the console.

The right pane contains the Details view. The Details view shows information about the item selected in the console tree. For example, when you click a profile in the console tree, the Details view allows you to view and edit the details of the profile.

## Action Menu

Most tasks in Diagnostics Console, such as protecting applications and running audits, can be accomplished from the Action menu.

The available commands on the Action menu depend on what type of item you select in the console tree. Right-clicking an item in the console tree opens a shortcut menu with the same commands.

---

*If the Action menu contains only the Help command, click in the console tree and open the Action menu again.*

---

## Console Tree

From the console tree, you can access any computer on your network to protect applications and run audits (collect diagnostics). You can also create and edit profiles, diagnose problems, analyze configuration changes, and schedule jobs.

**Profiles** For application protection, a profile specifies the application items to protect, such as files, registry entries, ActiveX controls, and shortcuts.

For change analysis, a profile specifies what configuration information to collect. In addition to files, registry entries, ActiveX controls, shortcuts, and environment variables, a profile can include lists of files to retrieve and system resource information (such as services, startup applications, and printers).

**Public versus Private** Public items are stored on a central server (in the SupportSite shared folder) and shared by all console users. For example, if you create a new profile you can share it with all

other users by saving it in the SupportSite folder. Private items are stored outside of the SupportSite folder, for example on your local hard disk.

---

*To audit and protect computers, a profile must be public. Private profiles can be used only on My Computer.*

---

**Audits** An audit is the configuration information and diagnostic data collected from a computer. An audit report is created whenever you audit or protect an application. Audit reports are stored in the Support Site.

In the console tree, audits are found under a computer node. The Audit Reports node is a general storage area for audit reports that you want to keep around for later change analyses.

**Snapshots** A snapshot is an archive of the application files listed in the profile. A snapshot is created when you protect an application, and stored on the local computer.

**Requests** Requests are audit and protect commands. Pending requests are waiting to be picked up by Diagnostics Agents. In Progress requests are being processed by the Diagnostics Agents.

In Progress requests are audit and protect jobs. A job request must finish before any another request is processed, while requests from a Diagnostics Console are processed independently in separate threads. While a job request is In Progress, all requests from Diagnostics Consoles are Pending.

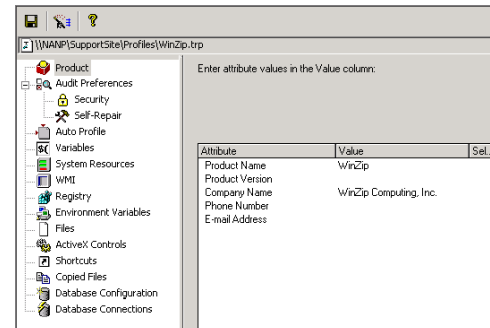
**Jobs** Jobs are audit and protect commands that are scheduled to run at specific times.

## Details View

The Details view displays the details of an item selected in the console tree. For example, you can view the details of a profile, an audit report, or of the problems found during an audit.

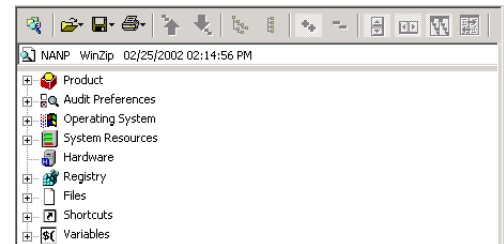
**Profile View** Allows you to create and edit profiles. The left pane contains the profile tree, which lists the different sections of a profile.

### Profile View



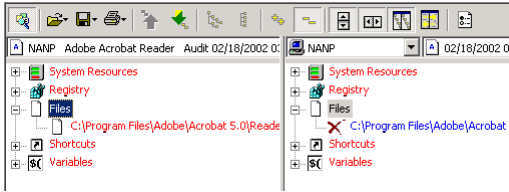
**Audit Report View** Allows you to review the contents of an audit report.

### Audit Report View



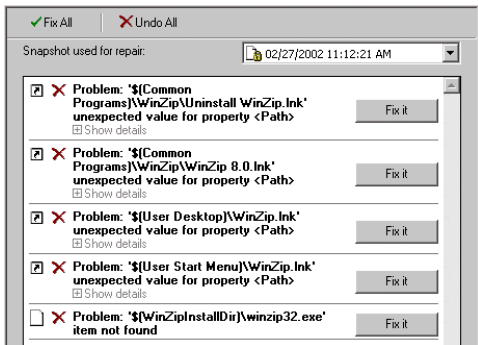
**Change Analysis View** Allows you to compare two audit reports. Differences between the two reports are visually highlighted, so you can quickly view problems such as missing files, wrong file versions, invalid registry entries, and invalid OS settings.

### Change Analysis View



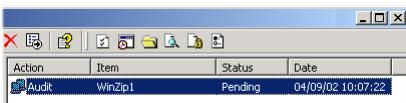
**Problem Diagnosis View** Allows you to review and fix the problems detected for a protected application.

### Problem Diagnosis View



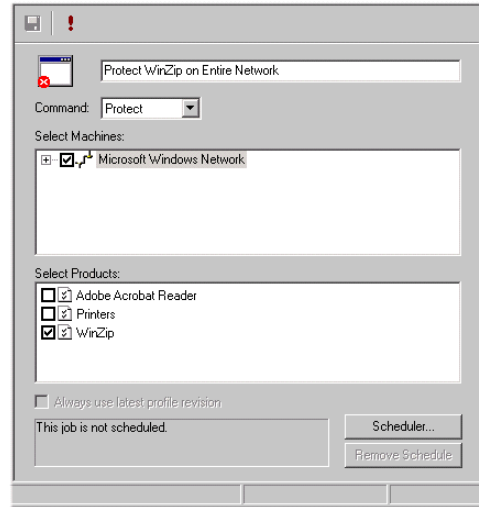
**Requests View** Allows you to check the status of audit and protect requests, and to delete requests if necessary.

### Requests View



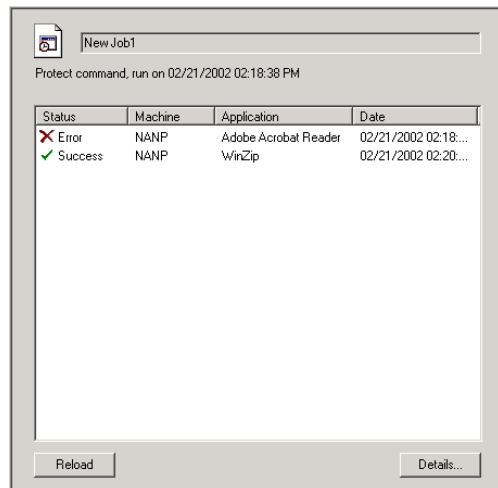
**Job View** Allows you to define and schedule audits and application protection. You can also define batch jobs to protect (or audit) an application on all computers in a domain or network.

### Job View



**Job Status View** Allows you to check the status of jobs.

### Job Status View



## Setting Up Diagnostics

Setting up Diagnostics involves installing at least one copy of Diagnostics Console, and a copy of Diagnostics Agent on every computer where you want to protect applications or collect diagnostics.

### To set up Diagnostics on your network:

- 1 Install a copy of Diagnostics from the CD. In addition to installing a copy of Diagnostics Console and Diagnostics Agent, the Setup program allows you to:
  - Set up the Support Site, the shared folder used by all agents and consoles.
  - Register your Diagnostics products.
  - Set the event logging options.
- 2 Install Diagnostics Agent on all computers.
- 3 Install additional copies of Diagnostics Console as required.

## Installing Diagnostics

You can install a copy of Diagnostics from the CD. After you install Diagnostics, the Setup program allows you to set options that will be shared by all agents and consoles (such as the Support Site location and event logging options).

By default, the Setup program installs all features in the default location. To select the features to install, choose the Custom setup type.

**Select Features** The Custom setup type allows you to select the features to install. You must install both Diagnostics Agent and Diagnostics. If you don't want to diagnose ODBC database problems, you don't need to install Diagnostics/db.

**Setting Up the Support Site** The Support Site is a shared folder on a network server. All agents and consoles must be able to access the Support Site. See "Support Site" on page 2 for more information on the Support Site.

**Setting the Support Site User Account** The Support Site user account is used by all agents and consoles to:

- Audit and protect computers.
- Access the Support Site shared folder.
- Run jobs.
- Run the Diagnostics service (named MQ Message Broker).

The Support Site user account must have the appropriate privileges on each local computer to perform tasks such as auditing and protecting a computer. Ideally, the Support Site user should be a Domain Administrator that has local Administrative privileges on each computer.

**Setting the Event Logging Options** By default, agents and consoles log events on the local computer. On Windows NT, 2000, and XP, events are logged to the Event Log. On Windows 95, 98, and Me, events are logged to a text file.

On Windows NT, 2000, and XP, you can log all events to the Event Log on a central server. See "Event Logging" on page 48 for more information.

**Registering Diagnostics** Each Diagnostics product (Diagnostics Console, Diagnostics Agent, Diagnostics/db) must be registered with its serial number. See "Registering Products" on page 48 for more information.

## Installing Diagnostics Agents

Diagnostics Agent must be installed on every PC where you want to protect or audit an application.

### To install the Diagnostics Agent manually:

- 1 Connect to the Support Site shared folder.
- 2 Run the setup program in Setup\Agent.
- 3 If you are going to use Diagnostics to diagnose ODBC database problems, Diagnostics/db must be installed on each computer. The Setup program automatically gets the required licenses.

To configure Diagnostics Agent, right-click the Agent icon in the status area of the taskbar (the area to the right of the taskbar buttons) and click Options.

## Installing Additional Consoles

You can install any number of additional consoles (each copy requires a separate license).

### To install additional Diagnostics Consoles:

- 1 Connect to the Support Site shared folder.
- 2 Run the setup program in Setup\Console.
- 3 If you are going to use Diagnostics to diagnose ODBC database problems, install the Diagnostics/db product. The Setup program automatically gets the required licenses.

## Creating a Silent Install for Diagnostics Agent

A normal (non-silent) installation requires input from the user as it displays a sequence of dialog boxes. A silent installation requires no input from the user. Instead, the Setup program gets its input from a InstallShield Silent response file (.ISS file).

In addition to the Agent setup program, Diagnostics includes two response files in the \\server\Support Site\Setup\Agent folder.

- setup.iss  
The default response file used when you run Setup.exe with the /s flag. Does not reboot the computer.
- setupr.iss  
Reboots the computer.

Both .ISS files perform a Custom setup that installs Diagnostics Agent and Diagnostics/db. You can edit the response files to change the install folder.

**Running Silent Installs** To run a silent install, run Setup.exe with the /s argument. By default, Setup.exe uses setup.iss, which is located in the same folder as Setup.exe. The /f1 argument can be used to specify a different response file.

For example:

```
set ss="\\server\SupportSite
\Setup\Agent\setup.exe"

rem Run a silent install with no reboot
rem (using the default setup.iss file in
rem \\server\SupportSite\Setup\Agent\
%ss% /s /verbose"c:\is.log"

rem Run a silent install with a reboot
%ss% /z"reboot" /f1"c:\setupr.iss" /s
/verbose"c:\is.log"
```

**Recording Response Files** To record a response file, run Setup.exe with the /r command-line argument. For example, you can record a response file to perform a Typical install.

```
set ss="\server\SupportSite
\Setup\Agent\setup.exe"
```

rem record a script with "no reboot" option (default)

```
%ss% /r /f1"c:\setup.iss" /f2"c:\setup.log"
/verbose"c:\is.log"
```

rem record a script with reboot

```
%ss% /z'reboot" /r /f1"c:\setupr.iss"
/f2"c:\setup.log" /verbose"c:\is.log"
```

When Setup finishes, it creates the response file in the specified folder.

All status information for a silent installation is recorded (by default) in a file called Setup.log, created in the same directory as the response file. To specify a different name and location for the log file, use the /f2 argument to Setup.exe.

## QuickStart

This section walks you through the processes of profiling and protecting an application, and then fixing it when something goes wrong.

### Profiling

**To profile an application:**

- 1 In the Action menu, click New and then click Profile.
- 2 In the profile tree, click Auto Profile.
- 3 Click Application.

Diagnostics displays a list of applications found on your computer.

- 4 Click an application and click OK.

Diagnostics starts the application, audits your computer, and generates the profile.

- 5 Save the profile.

### Save Profile



## Protecting

**To protect an application:**

- 1 In the console tree, right-click My Computer and click Protect.
- 2 In the Console dialog, double-click the application profile you just created.

After the application is protected, an audit and a snapshot are added under the computer entry in the console tree.

## Diagnosing and Repairing

You can now use the diagnosis and repair facility. To test this, open the application installation folder, and rename one of the files (for example, the main executable).

**Diagnose the problem and repair the application:**

- 1 In the console tree, right-click My Computer and then click Audit.
- 2 In the Console dialog, double-click the protected application to start the audit.
- 3 When the audit is finished, expand the audit and click Problems were detected.

The Details View displays a list of the problems detected during the audit. Note that if you renamed an executable, the shortcuts may also be broken.

- 4 Click Fix All to fix the problems and restore the application to working order. Diagnostics restores the file you renamed to break the application.

# Chapter 2: Protecting and Restoring Applications

## Profiling Applications

An application profile is used to protect an application. The profile lists the files, registry entries, ActiveX controls (.OCX), self-registered files (.OCX or .DLL), shortcuts, and environment variables that make up a working configuration of the application. Using this information, Diagnostics can take a snapshot of the application on a specific machine, and later use this snapshot to restore the application to working order.

You can automatically generate a profile from a Windows Installer package (.MSI) file, an InstallShield or Wise Installer project, or an existing installation. You can also use Diagnostics Console to manually edit the details of a profile.

## Profiling and the Windows Operating System

When you profile an application, it is important that you test the profile on the two main families of the Windows operating systems:

- Windows 95, 98, and Me.
- Windows NT, 2000, and XP.

Depending on the version of Windows, some setup programs install different files and create different registry entries. Therefore, you may need two profiles, one for each of the main families of Windows.

You may also need separate profiles within a family (for example, separate profiles for Windows XP and 2000).

## Building Profiles

Diagnostics provides several methods for building profiles:

### **Import a Windows Installer package (\*.MSI)**

**file** This is the recommended way to build a profile.

**Import the project file for a install package** If you use InstallShield (5.x, 6.x), InstallShield Express, or Wise Installer to develop install programs for your applications, you can import the project files. You can also import Visual Basic projects.

**Generate the profile from an application installation** when an application doesn't use Windows Installer and you don't have the source files for the setup, you can use an existing installation of the application.

**Build the profile manually** This method is ideal for building profiles for collecting configuration information for performing system change analysis. For example, to troubleshoot problems with hardware components such as printers and video cards that have associated software.

For applications, building profiles manually requires considerable, detailed knowledge of the application.

## Auto-profiling Applications

**Specifying What to Include** The Files, Registry Entries, Self-registered files (for example, OCXs), and Shortcuts check boxes control what items are included in the generated profile.

**Filtering Out Files and Registry Keys** As a general guideline, a profile should not exceed 1 megabyte in size. To control the size of a profile, use:

- File extensions to ignore to specify which files you do not want to include in the profile.
- Registry keys to ignore to specify which registry keys (for example, HKEY\_CLASSES\_ROOT) you do not want to include in the profile.

Filters are specified as a comma-separated list. You can include one or more spaces between commas to make the list more readable.

**Copying Files** When you generate a profile automatically, you can generate a list of files (ASCII or binary) to retrieve from the remote PC. For example, you can retrieve .INI and other configuration files from a user's PC.

The File extensions to process as Copy Files box is a comma-separated list of file extensions. When the profile is generated, all files with these extensions are added to the list of files to copy.

While copied files are not used to protect an application, they can be useful for performing change analysis.

---

*If the install path is found in the registry, Auto Profile creates a variable for the application install directory.*

---

## Importing Windows Installer (MSI) Packages

Microsoft Windows Installer is a component of the Windows operating system that manages the installation and removal of applications. A package (.MSI) file stores information regarding the application setup and installations and is distributed to end users.

Generating a profile from an MSI file is more reliable than reverse engineering an existing installation of the application. Whenever an application uses Windows Installer, you should use its MSI file to create its profile.


### To import an MSI package:

- 1 In the console tree, right-click Profiles, click New, and then click Profile.  
  
This creates a public profile (a profile that is available to all users running a Diagnostics Console). To create a private profile, expand Profiles, right-click Private, click New, and then click Profile.
- 2 In the profile tree, click Auto Profile.
- 3 Click MSI. Diagnostics displays a list of the MSI packages found on your system (in the Installer subfolder of your Windows System folder, for example, C:\WinNT\Installer).
- 4 If you don't see the package you want, click Browse to locate it.

---

*MSI files typically have unfriendly names such as 4499fdf.MSI. To find the MSI file you want, point to the file until the tooltip appears, or add the Title column to the Details view of the dialog (right click a column header, click More, and select the Title check box).*

---

- 5 Click a package and click OK.
- 6 Select the features you want to import into the profile and click OK.
- 7 If necessary, set the advanced MSI import options:
  - To import ActiveX controls from the MSI package, Diagnostics needs to scan HKEY\_CLASSES\_ROOT.
  - To ignore files, OCXs, registry entries, and shortcuts that are listed in the MSI package but not found on the local computer, select the Import items only if found on this computer check box.
  - If necessary, type the correct installation folder or click  to locate it.
  - To ignore components based on the install conditions specified in the .MSI file, type the install conditions in the Install conditions to ignore box. Use a semi-colon to separate each install condition.

---

*Avoid building large profiles, which slow down auditing, protecting, and change analysis.*

*Selecting features allows you to build smaller profiles. For example, the top-level features of Microsoft Office are Word, Excel, Power Point, and so on. By selecting features, you can create separate profiles for each Office program instead of one large profile for all of Microsoft Office.*

*For each top-level feature, you may also want to create profiles with and without optional features that some users may not install. For example, you may want a profile for an installation of Microsoft Word without the spell checker, so that all the profile items related to the spell checker won't be protected and identified as problems.*

*For complicated MSI packages, you may need to select shared components as well as the application. For example, to build a profile for Microsoft Outlook, you may need to select shared Office Tools components such as the Spell Checker. Otherwise you won't be able to diagnose spell checker-related problems with Outlook.*

*When you import an MSI file, the self-registered DLLs are not listed in the ActiveX Controls section of the profile. Instead, the Registry section includes all the registry entries required by the DLLs.*

---

## MSI Files and Self-Registered DLLs

When you import an MSI file, the self-registered DLLs are not listed in the ActiveX Controls section of the profile. Instead, the Registry section includes all the registry entries required by the DLLs.

## Importing Install Packages

If you have the source for an install package, you can use it to build a profile. Diagnostics can automatically import items from the following common install packages:

Install Package	What you can import
InstallShield 5.x, 6.x InstallShield 5.x, 6.x Log File Wise Installer	Files Self-registered files Registry keys Shortcuts
InstallShield Express	Files Self-registered files Registry keys
Visual Basic Project (vbp)	Files Self-registered files

**To import an install package:**

- 1 In the console tree, right-click Profiles, click New, and then click Profile.

This creates a public profile (a profile that is available to all users running a Diagnostics Console). To create a private profile, expand Profiles, right-click Private, click New, and then click Profile.

- 2 In the profile tree, click Auto Profile.
- 3 Click Package.
- 4 In the Files of type box, select the type of install package you want to import.
- 5 Click an install package and click OK.

**To import an InstallShield 5.x log file:**

- 1 Load the log (ISU) file in the InstallShield 5.x log file viewer.
- 2 Save it as a text file. Diagnostics Console can read only the text version of the log file.

## Installed Applications

Diagnostics can generate a profile from an existing installation of an application. After you select an installed application, Diagnostics scans your system for information about the application (such as files, registry entries, and shortcuts), starts the application to determine what ActiveX controls it uses, and then generates the profile.

**To auto-profile an installed application:**

- 1 In the console tree, right-click Profiles, click New, and then click Profile.

This creates a public profile (a profile that is available to all users running a Diagnostics Console). To create a private profile, expand Profiles, right-click Private, click New, and then click Profile.

- 2 In the profile tree, click Auto Profile.
- 3 Click Application. Diagnostics displays a list of applications found on the local PC.

If you do not see the application you want to profile in the Installed Applications dialog, click Browse and locate the application executable on your computer.

- 4 Click an application and click OK.

**ActiveX Controls** Diagnostics can determine only the ActiveX controls loaded at startup. ActiveX controls loaded on demand by the application are not included in the generated profile. If you are familiar with the application, you can manually add the missing ActiveX controls.

**Too Many Files?** If the generated list of files is too large, add some file extensions to the File extensions to ignore filter and generate a new profile.

**Files with No Path** If a file is listed with no path, it was probably found somewhere on your hard disk outside of the application installation directory and the standard Windows directories (for example, c:\temp). Generally, you can remove such files from the profile.

## After You Auto Profile

- Because not all applications follow standard rules for installations, profiles for installed applications may not be complete. Visually inspect the profile and verify that the files, registry entries, shortcuts, and so on make sense.
- If the profile includes keys or values under HKEY\_CURRENT\_USER, the user must be logged on when you audit, protect, or repair the user's computer. Otherwise, if no one is logged on, the current user will be the default user and the audited values will not reflect the user's environment.
- Make sure all paths to files and shortcuts use variables.
- If the application depends on environment variables, you must add them by hand.
- Check the Product attributes.

## Collecting Information for Change Analysis

If you cannot restore a protected application to working order using automated repairs, you may need to perform a change analysis. Diagnostics can quickly identify changes in system and application configuration that may be the cause of the problem.

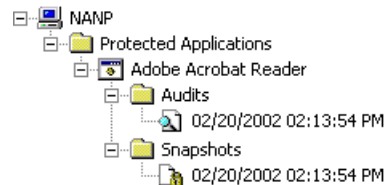
For example, you can easily collect configuration information on the operating system, system resources such as services, running applications, and memory, and hardware components. For details, see Chapter 4, "Collecting Information" on page 21.

## Protecting Applications

Protecting an application on a computer creates an audit and a snapshot. The audit represents the configuration of the application on a specific computer at a specific point in time. For example, the audit specifies the location and version of each file listed in the profile, the values of the registry entries, along with information on any ActiveX controls and shortcuts.

The snapshot is an archive of the files and ActiveX controls at that point in time, and is used to restore the application to a working configuration when a problem occurs.

### An Audit and a Snapshot



### To protect an application on a single PC:

- 1 In the console tree, expand the Entire Network and locate the computer.
- 2 Right-click the computer and click Protect.
- 3 In the Console dialog, double-click an application.

After the application is protected, an audit and a snapshot are added under the computer in the console tree. You can now view the audit details.




---

*While Diagnostics protects the computer, you can perform other tasks in the Diagnostics Console. For example, you can protect the application on other computers.*

If the protect request seems to be taking a long time to finish, check the Requests. If the request is listed there, it means that the Diagnostics Agent running on the target computer never picked up the request.

You can schedule jobs to protect applications at a more convenient time. See Chapter 6, “Scheduling Jobs” on page 43 for more information.

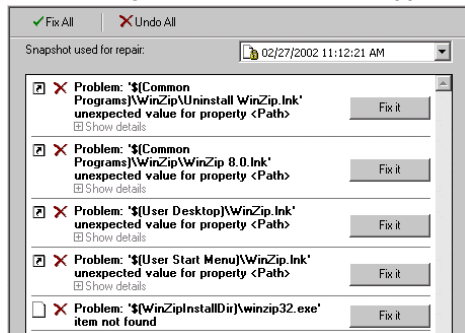
### To batch-protect applications on the network:

- 1 On the Action menu, click New and then click Job.
- 2 In the box beside the task icon , type a name for the job.
- 3 In the Command list, click Protect.
- 4 In the Select Machines box, select the Microsoft Windows Network check box. Or select the check box for one or more domains.
- 5 In the Select Products box, select the check boxes for the products you want to protect.
- 6 In the Job view toolbar, click  to save the job.
- 7 In the Job view toolbar, click  to run the job.

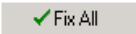

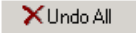
## Restoring Applications to Working Order

When a user reports a problem with a protected application, you can automatically diagnose the problem and fix the application. All you have to do is use the application profile to audit the user's computer. If any problems are detected during the audit, you can fix them by clicking a button.

### Problem Diagnostics for a Protected Application



### To diagnose and fix a problem:

- 1 In the console tree, right-click a computer and then click Audit.
- 2 In the Console, double-click the protected application to start the audit.
- 3 When the audit is finished, click Problems were detected. The Details view displays a list of the problems detected during the audit.
- 4 In the Snapshot used for repair list, click the snapshot you want to use to repair the application. This allows you to restore the application to its configuration at a specific point in time.
- 5 Review the problems and fix them:
  - To fix all problems, click 
  - To fix a specific problem, click 
  - To undo all fixes, click 

If the audit request seems to be taking a long time to finish, check the Requests. If the request is listed there, it means that the Diagnostics Agent running on the target computer never picked up the request.

# Chapter 3: Performing Change Analysis

Change analysis is a basic technique for troubleshooting system and application problems. It's the process of tracking down configuration changes on a computer.

With Diagnostics, you can build profiles to collect application and system configuration information, audit computers, and then analyze the collected diagnostic data. Diagnostics automatically compares application or system settings against a baseline, at different points in time, or on different computers. This allows you to quickly identify and correct the changes that caused the problem.

## Manually Building a Profile

To manually build a profile, you have to decide what information you want to collect. For example:

- Do you want to collect information on files? Which files? DLLs, ActiveX controls, shortcuts, or other types of files? Do you want to retrieve copies of files?
- Do you want to check the registry for specific keys and values?
- What kind of system configuration information do you want to collect? Installed applications? Running services? Loaded modules? Memory usage? Hardware components?

## Adding Items

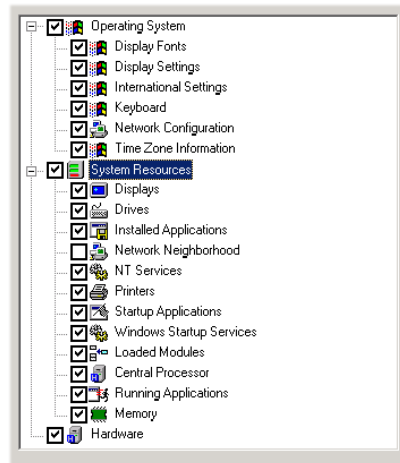
Diagnostics Console includes tools for building lists of items you want to audit, such as files, ActiveX controls, registry keys and values, shortcuts, and environment variables. To simplify the process, you can use regular expressions to select groups of files based on their names (for example, all the DLLs in a folder). You can also define variables to represent computer-specific values such as paths.

See Chapter 4, "Collecting Information" on page 21 for more information on adding items to a profile.

## Collecting System Resource Information

Setting up a profile to collect system resource information is straightforward. Just check off the items you want to collect.

### System Resources



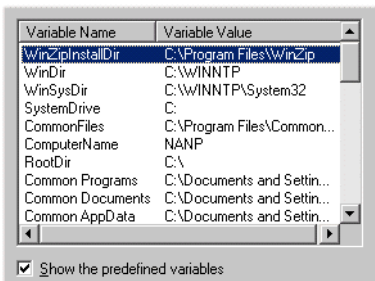
System resource information can include:

- Operating system information.
- System configuration information such as the amount of free disk space, what DLLs are loaded into memory, and what applications are running.
- Hardware component and configuration information.

## Defining Variables

You use variables to represent paths that can vary from computer to computer, such as the location of the Windows system folder or the installation folder of an application.

### Variable Definitions

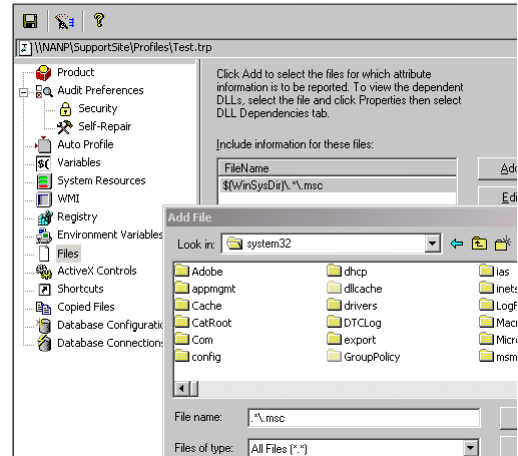


If you want to collect information on files and shortcuts, or retrieve copies of file, you can use variables to locate the files on each computer.

For example, you can use variables to represent the location of the Windows system folder, the installation folder of an application, or the location of the shortcuts on the Start menu.

For application files, you can define a variable that extracts the application install path from the registry, or use a predefined variable such as \$(Common Files), which stores the location of the Program Files\Common Files folder.

### Adding Files



## Auditing PCs

Auditing is the process of collecting diagnostic and configuration information from a computer. For basic change analysis, you can simply audit a computer to see if anything listed in the profile (such as a file) is missing.

For more detailed change analysis, you need at least one baseline audit of a working configuration on a computer. Then when a problem occurs, you can audit the non-working configuration and compare it against the baseline audit.




You can keep just a baseline audit, or you can periodically audit a computer to track configuration changes over time (for example: original configuration, configuration after a operating system upgrade, and so on).

Audits are saved on the Support Site server, so after you audit you do not have to connect to the computer again to diagnose the problem. All the collected diagnostics and configuration information is available from the Support Site server.

#### To audit a PC:

- 1 In the console tree, right-click a computer and then click Audit.
- 2 In the Console dialog box, double-click a profile to start the audit.

#### To batch audit PCs on a network:

- 1 On the Action menu, click New and then click Job.
- 2 In the box beside the task icon , type a name for the job.
- 3 In the Command list, click Audit.
- 4 In the Select Machines box, select the Microsoft Windows Network check box. Or select the check box for one or more domains, or for one or more computers.
- 5 In the Select Products box, select the check boxes for the products you want to audit.
- 6 In the Job view toolbar, click  to save the job.
- 7 In the Job view toolbar, click  to run the job.

---

*You can limit the maximum number of audit reports saved for each application on a computer. When the limit is exceeded, the oldest audit is deleted. To set the limit: on the Action menu click Options, and then click the Maintenance tab.*

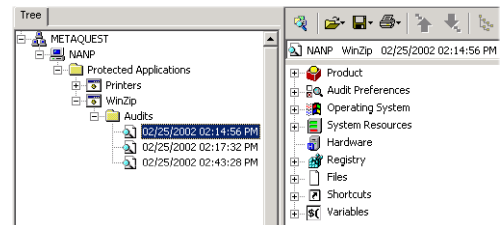
*If the audit request seems to be taking a long time to finish, check the Requests.*

*If the audit request is listed there, it means that the Diagnostics Agent running on the target computer never picked up the request.*

---

## Viewing Audit Reports

### Audit Report



### To view an audit report:

- 1 In the console tree, expand a computer, then expand Protected Applications and expand an application.



- 2 Expand Audits and click an audit report.

- 3 In the Details view, expand the sections of the audit report you want to view.

---

*If a section name is highlighted in a different color, that means an item is missing in the audit report (for example, a file was not found).*

---

*In an audit report, the Variables section contains the values of the variables on the audited computer.*

---

## Opening and Editing Copied Files

By default, ASCII and binary files are always attached to the audit report, and opened or edited with their associated applications. However, ASCII files can be included in the body of the audit report, and viewed directly in Diagnostics Console (if the Attach Copied Files preference is set to False). Including copied files in audit reports also allows you to compare the contents and highlight differences.

### To view attached files:

- 1 In the Details view, expand Copied Files.
- 2 Under Copied Files, right-click the file you want to view.
- 3 Click Open, Open With, or Edit.

The command you choose depends on the type of file and what actions are associated with that file type. For example, on some systems, Open will execute a javascript (.JS) file, while Edit will simply load the file into a text editor.

If you are not sure, click Open With and click the program you want to use to open the file.

### To view included files:

- 1 In the Audit view, expand Copied Files.
- 2 Under Copied Files, expand the file you want to view.
- 3 Expand Contents.

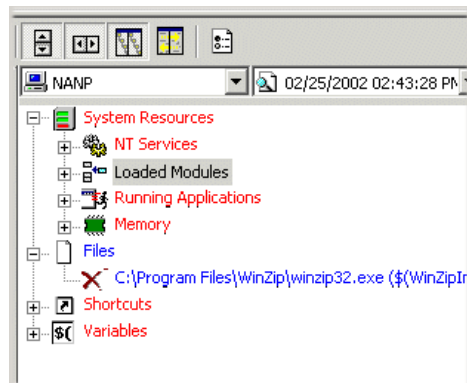
### To copy content from included files:

- 1 Expand Contents.
- 2 Right-click the line you want to copy and click Properties.
- 3 Highlight the text you want to copy.
- 4 Right-click the highlighted text and click Copy.

## Comparing Audit Reports

When you compare two audit reports, Diagnostics automatically highlights any differences between the two reports. This allows you to review configuration changes and quickly spot bad or missing files, wrong file versions, missing registry entries, invalid OS settings, and more.

### Changes Visually Highlighted



You can compare a computer's configuration:


- Against a baseline.
- At two different points in time.
- Against the configuration of another computer.

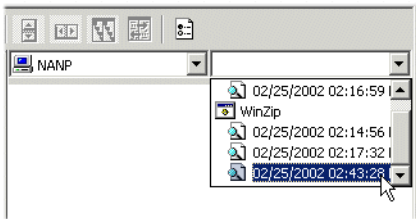
---

*The reference report is in the left pane, and the audit report is in the right pane.*




---

### To compare audit reports:

- 1 In the console tree, click the audit report you want to use as a baseline for the comparison.
- 2 In the Details view toolbar, click Compare Audit Reports .
- 3 In the right pane, click a computer in the list (this allows you to compare the configuration of one computer against another), then click an audit report in the list of available audits.







- 4 Review the differences:

- Click  to display only the differences.
- Click  or  to display the next or previous difference.

By default, the display of the two audit reports is synchronized, so that both reports scroll up and down together, and expand and collapse together.

This makes it easier to perform a side-by-side comparison of the reports. Turn this feature off if you want to view each report independently.


To turn off	Click
Synchronized vertical scrolling	
Synchronized horizontal scrolling	
Synchronized expanding and collapsing of report sections	

Click Synchronize Item  to display the same item in both reports when display synchronization is turned off.

## Filtering Audit Reports

Filtering allows you to filter out irrelevant differences when comparing audit reports. Use filters to reduce the number of differences displayed when you view differences only.

### To filter out differences:

- 1 In the Details view toolbar, click Options .
- 2 In the Filters tab, clear the check boxes for the audit items you want to filter out.
  - Select when to apply the filter: When viewing differences only or all items, click Always.
  - When viewing differences only, but not when viewing all items, click when viewing Differences Only.

To disable filtering, click Never on the Filters tab.

---

*Diagnostics saves the filter settings, so each time you compare two audit reports the same items are filtered out.*


*Filtered items are never highlighted when they are different. For example, if you choose to always apply a filter, the filtered items are never highlighted as different, even if they are.*

*Filters are ignored if you load a single report.*

---

## Customizing the Difference Highlighting

**To customize difference highlighting:**


- 1 In the Details view toolbar, click Options  and then click the General tab.
- 2 Change the colors.

To change the color of	Do this
Items that are different in each report.	In the Color of different items list, click a color.
Items missing in the audit report displayed in the left pane	In the Color of items missing in reference report list, click a color.
Items missing in the audit report displayed in the right pane	In the Color of items missing in audit report list, click a color.

## Hiding Files from Non-Active Operating Systems

When more than one operating system is installed on a computer, an audit report contains information for each operating system. You can filter out the non-active operating system when viewing the audit report.


**To filter out files from the non-active OS:**

- 1 In the Details view toolbar, click Options  then click the General tab.
- 2 Click Ignore files in the non-active operating system.

## Printing Audit and Diagnostic Reports


Diagnostics can print audit reports and diagnostic reports. A diagnostic report summarizes the differences between two audit reports. You can also save diagnostic reports (in a .TRD file).

**To print an audit report:**


- 1 View an audit report.
- 2 In the Details toolbar, click  and then click Print Reference.

If you are comparing audit reports, Print Reference prints the audit in the left pane, and Print Audit prints the audit in the right pane.

**To print a diagnostic report:**

- 1 Compare two audit reports.
- 2 In the Details toolbar, click  and then click Print Diagnostic.

**To save a diagnostic report:**

- 1 Compare two audit reports.
- 2 In the Details toolbar, click  and then click Print Diagnostic.

# Chapter 4: Collecting Information

In addition to collecting information on files, registry entries, ActiveX controls, self-registered files, shortcuts, and environment variables, a profile can also collect:

- System, operating system, and hardware information.
- Copies of text and binary files. For example, you can get copies of text files such as .INI, .SYS, and .BAT files.
- Database configuration and connection information.
- Advanced diagnostics from Microsoft Windows systems through Windows Management Instrumentation (WMI).
- Diagnostic information about Microsoft Internet Information Server (IIS).

## Defining Variables

Diagnostics uses variables to specify the paths to files and shortcuts. A variable can represent a file path that can vary from machine to machine. For example, the location of the Windows folder can vary from machine to machine, and different users can install an application in different directories.

If an application stores paths in the registry, in an INI file, or relies on environment variables, Diagnostics can use variables to look for files and shortcuts only in those locations. Otherwise, Diagnostics searches the entire computer.

Similarly, if you know that a file should be in the Windows folder, you can use a variable to search only the Windows folder.

Variables can be combined together to form a single expression. Variables can also be combined with regular expressions.

Diagnostics provides the following variable types:

- Registry variables that are expanded based on a value stored in the registry.
- INI variables that are expanded based on a value stored in an INI file.
- Predefined variables that are automatically expanded by Diagnostics.
- Environment variables that are expanded based on the value of an environment variable.
- User-defined variables, which act like constants in a profile.

## Using Variables

To reference a variable, you type an expression of the form \$(Variable Name), where Variable Name is the name you gave to the variable when you defined it.

To reference an environment variable, enclose it in “%(“ and “)”. For example, “%(TEMP)”.

You can use variables with the following items:

- File names of files, shortcuts, ActiveX controls, and files to copy (to specify computer-specific paths)

- Definitions of Variables

You can use INI, Registry, Pre-defined, and Environment variables in the definitions of INI and Registry variables.

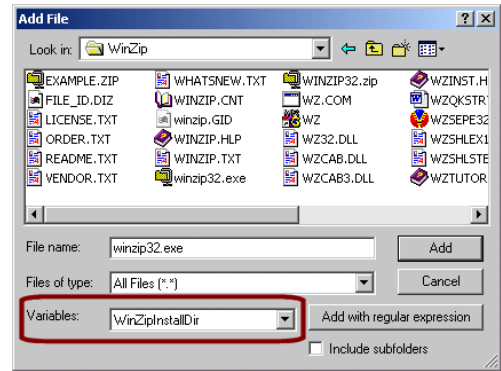
- Values of the Pre-audit Application and Post-audit Application audit preferences.
- Self-repair rules for files, ActiveX controls, shortcuts.
- Repair rule conditions.
- Database Information: database connection names, SQL statements, SQL server attributes, and SQL connection strings.

## Registry Variables


A registry variable represents a value stored under a registry key (either the default value or a named value). You use these variables to specify file paths when you add files, ActiveX controls, files to copy, or shortcuts to a profile. When you use a variable to specify the path to a file or shortcut, Diagnostics looks only in that location for the file. Otherwise, Diagnostics searches all drives for the file.

For example, suppose an application stores its installation directory in the registry as the default value of a key named InstallPath. If you want Diagnostics to look for files in this installation directory, you can define a variable that extracts the default value of the InstallPath key. Then you can use this variable to specify the location of the file.

## Using a Variable to Specify the Location of a File



## To define a registry variable:

- 1 In the profile tree, click Variables.
- 2 Click Add.
- 3 Click Registry to define a registry variable.
- 4 In the Registry Key row, click  to open the Registry dialog, and select a registry value.

If you select a registry key, the variable is given the default value of the key (if the default value is set).

- 5 In the Variable Name row, click in the Value column and enter a name for the variable.

## INI Variables

An INI variable represents a value stored in an INI file. You can use these variables to specify file paths when you add files, files to copy, or shortcuts. For example, suppose an application stores its installation directory in an INI file as follows:

```
[Paths]
```

```
InstallPath=C:\Program Files\Company\App
```

You can then define an INI variable that extracts the value of the InstallPath entry in the PATHS section of the INI file. This INI variable can then be used to specify the location of a file.

When you use a variable to specify the path to a file, Diagnostics looks only in that location for the file. Otherwise, Diagnostics searches all drives for the file and audits every instance it finds.

#### To define a registry variable:

- 1 In the profile tree, click Variables.
- 2 Click Add.
- 3 Click INI to define an INI variable.
- 4 Enter the name of the INI file, the name of the INI section, and the name of INI entry.
- 5 In the Variable Name row, click in the Value column and enter a name for the variable.

## Other Variable Attributes for INI and Registry Variables

The Variable Value attribute is set when you click OK or Apply. This value is used while building the profile (for example, to find the files you add to the profile). During an audit or protect, the variable value is determined by the settings of the user's computer.

The Default Value attribute is used when the value cannot be extracted from the INI file. For example, when an application is protected, the Default Value attribute is assigned the value of the variable. Then when the application needs to be repaired, the value will be available even if it cannot be found in the registry.

The Extract As and Variable Data Type attributes are used to extract folder paths from file names. See "Extracting Folders from File Names" on page 25.

## User-Defined Variables

A user-defined variable is a variable that stores a value specified in the profile. If you want to use the same value (for example, a string) in a number of places, you can define a variable to hold this value.

#### To define a user-defined variable:

- 1 In the profile tree, click Variables.
- 2 Click Add.
- 3 Click User Defined.
- 4 In the Variable Name row, click in the Value column and type a name for the variable.
- 5 In the Default Value row, click in the Value column and type a value.

## Predefined Variables

Predefined variables are variables whose values are supplied by Diagnostics when you audit or protect a computer. Most of the predefined variables provide computer-specific values, such as the location of the Windows folder and the name of the computer.

## Predefined System Variables

**WinDir** Windows folder (for example, "c:\WinNT").

**WinSysDir** Windows system folder (for example, "c:\WinNT\system32").

**SystemDrive** Drive where the operating system is installed (for example, "c:\").

**CommonFiles** Windows common files folder (for example, "c:\Program Files\Common Files").

**ComputerName** Name of the computer (for example, "KIMA").

**RootDir** Boot drive (for example, "c:\").

## Predefined User-profile Variables

**Common Desktop** Location of the shared Desktop folder. For example, %SystemRoot%\Profiles\All Users\Desktop.

**Common Documents** Location of the shared Documents folder. For example: C:\Documents and Settings\All Users\WINNTP\Documents.

**Common Administrative Tools** Location of the shared Application Data folder. For example, C:\Documents and Settings\All Users\Administrative Tools\.

**Common AppData** Location of the shared Application Data folder. For example, C:\Documents and Settings\All Users\Application Data\.

**Common Programs** Location of the shared Programs folder. For example, %SystemRoot%\Profiles\All Users\Start Menu\Programs.

**Common Start Menu** Location of the shared Start Menu folder. For example, %SystemRoot%\Profiles\All Users\Start Menu.

**Common Startup** Location of the shared Startup folder. For example, %SystemRoot%\Profiles\All Users\Start Menu\Programs\Startup.

**Common Templates** Location of the shared Templates folder. For example, C:\Documents and Settings\All Users\Templates\.

**Personal** Location of the current user's My Documents folder. For example, C:\Documents and Settings\stephen\My Documents\.

**AppData** Location of the current user's Application Data folder. For example, C:\Documents and Settings\stephen\Application Data\.

**Cookies** Location of the current user's Cookies folder. For example, C:\Documents and Settings\stephen\Cookies\.

**Desktop** Location of the current user's Desktop folder. For example, C:\Documents and Settings\stephen\Desktop\.

**Favorites** Location of the current user's Favorites folder. For example, C:\Documents and Settings\kima\Favorites\.

**NetHood** Location of the current user's NetHood folder. For example, C:\Documents and Settings\kima\NetHood\.

**My Pictures** Location of the current user's My Pictures folder. For example, C:\Documents and Settings\kima\My Documents\My Pictures\.

**PrintHood** Location of the current user's PrintHood folder. For example, C:\Documents and Settings\kima\PrintHood\.

**Recent** Location of the current user's Recent folder. For example, C:\Documents and Settings\kima\Recent\.

**SendTo** Location of the current user's SendTo folder. For example, C:\Documents and Settings\kima\SendTo\.

**Start Menu** Location of the current user's Start Menu folder. For example, %SystemRoot%\Profiles%\UserName%\Start Menu.

**SendTo** Location of the current user's SendTo folder. For example, C:\Documents and Settings\kima\SendTo\.

**Templates** Location of the current user's Templates folder. For example, C:\Documents and Settings\kima\Templates\.

**Startup** Location of the current user's Startup folder. For example:

%SystemRoot%\Profiles\%UserName%\Start Menu\Programs\Startup.

**Local Settings** Location of the current user's Local Settings folder. For example, C:\Documents and Settings\kima\Local Settings\.

**Local AppData** Location of the current user's local Application Data folder. For example, C:\Documents and Settings\kima\Local Settings\Application Data\.

**Cache** Location of the current user's Temporary Internet files folder.

**History** Location of the current user's History folder.

**Fonts** Location of the system fonts folder. For example, C:\WinNT\Fonts.

**Administrative Tools** Location of the current user's Application Data folder. For example, C:\Documents and Settings\kima\Administrative Tools\.

## Environment Variables

You can control where Diagnostics locates files by prefixing a filename with an environment variable. For example, to locate a file in the TEMP directory, you can specify %(TEMP)\myfile.txt.

Typical environment variables that could be useful as variables:

- %(COMPUTERNAME) returns the name of the computer where Diagnostics Agent is running.
- %(SYSTEMDRIVE) returns the drive on which the active operating system is installed.
- %(TEMP) returns the path of the temporary folder.

## Extracting Folders from File Names

Sometimes, an application does not store its installation path in the registry, but it does store the full path names of some files in its installation folder. You can define a variable that gets the file name from the registry, and then extracts only the path part.

For example, if a registry value is C:\Program Files\MyApp\myapp.exe, you can define a variable that extracts just the C:\Program Files\MyApp part.

### To extract the folder from a file name:

- 1 Create a new registry variable.
- 2 Set the Extract As attribute to Folder. This specifies how to extract the variable value when replacing a variable reference in the profile.
- 3 Set the Variable Data Type attribute to File. Variable Data Type specifies what kind of value is stored in the registry key.

For example, if you auto profile the WinZip application, the following variable is defined:

<b>Variable Name</b>	WinZipInstallDir
<b>Extract As</b>	Folder
<b>Registry Key</b>	HKCU\software\nico mak computing\winzip\programs\zip2exe
<b>Variable Value</b>	G:\Program Files\WinZip\WZSEPE32.EXE
<b>Variable Data Type</b>	File

Given this variable definition, \$(WinZipInstallDir) evaluates to G:\Program Files\WinZip.

## Using Regular Expressions

Use regular expressions to select groups of files based on their names. For example, to select all MFC DLLs in the Windows system directory, you would use the regular expression "^mfc.\*\dll".

Diagnostics audits any file whose name contains a substring that matches the regular expression. So, for example, the regular expression "mfc" matches any file containing the string "mfc"—not just the DLLs, but also files like "mfcuix.hlp" and "MFC Tracer" (a shortcut).

.

The period (.) matches any character. For example, "ie." matches both "ie5" and "ie6". To match an ordinary period, you use the backslash. For example, "\.ini" matches ".ini".

\*

The asterisk (\*) matches zero or more occurrences of the preceding character. For example, ".\*" matches any string of characters, and ".\*\dll" matches all DLLs.

^

The caret (^) matches the beginning of a string. For example, "^reg" matches any string that begins with "reg".

\$

The dollar sign (\$) matches the end of a string. For example, "ini\$" matches any string that ends with "ini". And while "\.ini" matches both "runlog.ini" and "foo.init", "\.ini\$" matches only files with a ".ini" extension.

[ ]

Matches a range of characters. For example, "[A-Za-z0-9]" matches any alphanumeric character. "[0-9]\*" matches zero or more digits. If the first character is the caret (^), the expression matches any character not in the range. For example [^AB^] matches any character except A, B and the caret itself.

\

Used to escape special characters. For example, "\" matches a period (.) and "\\\$" matches a dollar sign (\$).

## Examples

**To look for all files that have a .DLL extension:**

- 1 In the File Name box, type the regular expression ".\*\DLL".
- 2 Click Add with regular expression.

**To look for all files in a specific folder:**

- 1 In the File Name box, type the regular expression ".\*\.\*".
- 2 Click Add with regular expression.

**To look for all files that have a .DLL extension in the Windows system directory:**

- 1 In the Variables list, click WinSysDir.
- 2 In the File Name box, type the regular expression ".\*\DLL".

- 3 Click Add with regular expression.

**To look for all files that have a .DLL extension in the Windows system directory and its subfolders:**

- 1 In the Variables list, click the WinSysDir variable.
- 2 Click the Include subfolders check box
- 3 In the File Name box, type the regular expression “.\*\DLL”.
- 4 Click Add with regular expression.

**To look for all files that have a .DLL extension in a subfolder of the Windows system directory:**

- 1 In the Variables list, click the WinSysDir variable.
- 2 In the File Name box, type the regular expression “aSubFolderName\.\*\DLL”.
- 3 Click Add with regular expression.

## System Resources

Diagnostics can collect a wide variety of information about the configuration of a computer:

- System resource information, including displays, drives, installed applications, NT services, printers, startup applications, loaded modules, central processor, running applications, memory, and RAM.
- Operating system information, such as international settings, keyboard, time zone information, and Windows system information.
- Hardware information about components such as CD-ROM drives, disks, displays, hard drive controllers, monitors, ports, and system boards.

**To collect system resource information:**

- 1 In the profile tree, click System Resources.
- 2 Select the check boxes for the information you want to collect. Clear the check boxes for information you don't want to collect.

---

*To select just one or two check boxes under Operating System or System Resources, clear the top-level check box. This clears all check boxes so you can then select the check boxes you want.*

*By default, the Network Neighborhood check box (under System Resources) is cleared. Do not select this check box for large networks because auditing can take a substantial time.*

*The system resource information collected by Diagnostics depends on the version of Windows installed. For example, Display Fonts information is collected on Windows 95 and 98, but not on Windows NT or 2000. If Diagnostics does not collect the system resource information you need, use Windows Management Instrumentation (WMI) to collect the required information. See “Auditing with Windows Management Instrumentation” on page 31.*

---

## Auditing Files

A profile includes a list of application files that you want to audit. To include files in a profile, you can:

- Select files from the folders on your computer or on any other computer in the network neighborhood.
- Add all DLLs that one of your application DLLs depends on.

For an EXE file, Diagnostics automatically collects information about the DLLs that the EXE loads (so you don't have to add the DLLs yourself in Diagnostics Console).

But if you want to collect information for all instances of a DLL on a system, you must add the DLL to the profile.

When you audit a file, Diagnostics collects information for all instances of the file found on the computer. Use variables in the file name to collect information for only one specific instance of the file.

#### To add files:

- 1 In the profile tree, click Files.
- 2 Click Add.
- 3 Locate the folder containing the files you want to add.
- 4 Add the files you want to audit:

To add specific files, select the files.

To add all files whose names match a regular expression, type the regular expression in the File Name box.

- 5 If you have defined a variable to represent the location of the files, then in the Variables list, click the variable that represents the location of the files.
- 6 If you selected the files, click Add. If you typed a regular expression in the File Name box, click Add with regular expression.

#### To use a variable to represent the location of the files:

In the Variables list, click the variable.

Note that if you use a variable and regular expressions, you do not have to locate the actual folder containing the files.

#### To search subfolders for the files:

Click the Include Subfolders check box.

#### To include files in a profile even if they do not exist on your computer:

Type the file names in the File Name box.

#### To search network drives and CDROMs:

By default, Diagnostics searches for files on the local hard drives of a user's machine. If you want Diagnostics to also search network or CD-ROM drives by default, set Include Network Drives and Include CDROMS to True in the Audit Preferences.

#### To add DLL dependencies:

- 1 Add a DLL to the profile, click it, and then click Properties.
- 2 Click the DLL Dependencies tab to browse the hierarchy of DLLs that your application DLL depends on.
- 3 Click Add All to add all the required DLLs to the list at the bottom of the dialog, or click Add Selected Item to add just the selected DLL.
- 4 Click OK to add the DLLs to the profile.

## Collecting File Version Information

The File Version Information audit preference determines how much file version information is collected during an audit. Setting this attribute to Minimal or Normal reduces the amount of memory and time required to audit files. It also reduces the size of the audit reports, so they load and compare faster.

**Minimal** extracts FileVersionProp, FileDescriptionProp, and LegalCopyrightProp.

**Normal** extracts the Minimal information plus: CompanyName, InternalName, OriginalFileName, Productname, and ProductVersion.

**Full** extracts Normal and Minimal information plus: Comments, FileVersion (not the same as the one above), ProductVersion (not the same as the one above), TradeMarks, PrivateBuild, Special-Build, fileFlagsMask, FileFlags, Os, Type, SubType, Translations, and TranslationsCharset.

## Auditing ActiveX Controls

A profile can include a list of ActiveX controls (.OCX) and self-registered files (.OCX or .DLL) to audit. For example, you can set up a profile to check that a DLL is registered correctly.

For each ActiveX control listed in the profile, an audit report includes the CLSID and TypeLib information found in the registry, as well as general and file version information.

To add ActiveX controls to a profile:

- 1 In the profile tree, click ActiveX Controls and then click Add.
- 2 In the Add ActiveX Controls dialog, select the files you want to add and click Add. You can also type the name of a file in the File Name box.

## Auditing Registry Keys and Entries

A profile can include a list of registry keys and values to collect during an audit.

### Adding Keys and Values

If you add a registry key, Diagnostics adds all values and subkeys under that key, and selects the key. If you add a registry value, Diagnostics adds just the value and selects it.

### Selecting Keys and Values

During an audit, Diagnostics gets the selected keys and values. To select a key or value, click the check box for the key or value.

For each selected key, Diagnostics gets all values entered in the registry for the key. If the Recursive Registry Scan audit preference is True, Diagnostics gets all subkeys and values under that key.

---

*Only selected keys and values can be repaired.*

---

## Synchronizing

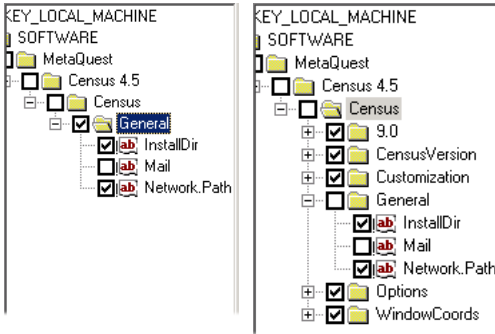
Synchronizing allows you to add missing subkeys and values. For example, after manually adding a single subkey, you may decide you want all the subkeys at the same level. To do this, click the parent key and then click Synchronize.

---

*After you synchronize, you must select the keys and values you want to audit.*

---

### Before and After Synchronizing a Key



## Restricting Keys

To prevent users from selecting keys such as HKEY\_LOCAL\_MACHINE\Software and all their subkeys and values, you can build a list of restricted keys. Restricted keys cannot be added or selected.

The list of restricted keys is stored in the file ProfViewer.ini, which you can find in the Diagnostics installation folder.

## Auditing Shortcuts

A profile can include a list of shortcuts (.LNK files) to check. For example, you can set up a profile to check that a shortcut exists and that it points to the correct target.

For each shortcut listed in the profile, an audit report includes shortcut properties such as the shortcut's target, arguments, and working directory.

### To add shortcuts to a profile:

- 1 In the profile tree, click Shortcuts and then click Add.
- 2 In the Add Shortcuts dialog, select the shortcut files you want to add and click Add.

You can use variables such as Common Start Menu to represent the location of the shortcut. In the Variables list, click a variable. Click Add to add the shortcut files.

## Copying Files

A profile can include a list of files to retrieve during an audit. These files can be text files or binary files.

Unless you use a variable to specify the exact location of the file to copy, Diagnostics copies all occurrences of the file it finds on the computer. Therefore, it is strongly recommended to use variables when specifying files to copy.

## Attaching Copied Files

Binary files are always attached to audit reports. And by default, ASCII files are also attached to audit reports (so the audit report contains only a reference to the copied files, which are stored externally in the file system).

Attaching the copied files reduces the size of the audit report and reduces the amount of time required to load the report into Diagnostics Console. It also allows you to use the application associated with the file type to open or edit the file.

You can include copied ASCII files in the audit report file by setting the Attach Copied Files attribute to False. Including copied files in an audit allows you to automatically compare them when you compare audit reports.

However, including copied files increases the size of the audit report and the time required to load the report into Diagnostics Console. It also means you cannot open the file in another application (such as Notepad).

---

*If you are retrieving copies of large files, attach them to the audit report.*

---

## Auditing with Windows Management Instrumentation

Windows Management Instrumentation (WMI) is the Microsoft implementation of Web-Based Enterprise Management (WBEM), which is an industry initiative to develop a standard technology for accessing management information. Such management information includes information on

the state of system memory, networks, devices, and other information on client status. WMI offers a powerful set of base services that include query-based information retrieval and event notification.

WMI is supported on Windows 2000, XP, and Me, and available as an optional install for Windows 95 OSR 2, 98, and NT4 SP5.

## WMI Components

An application profile can specify a list of WMI components and their properties to audit. To customize the WMI information audited, you can:

- View the properties and change their values.
- Reload the factory default settings for a category or an object.

To customize the WMI Components list, you can:

- Remove a component or a category from the list.
- Add additional WMI Components for selection.

## Editing WMI Category Properties

**Display name** Caption of the WMI category (referred to as a namespace).

**Namespace** Specifies the server path of the namespace.

## Editing WMI Component Properties

**Display name** Caption of the WMI component.

**Query Associators** If True, Diagnostics audits all associated WMI objects.

**WMI SQL** SQL statement that specifies what information to retrieve. You can change the name and the value of the WMI SQL property. You can also add new SQL statements for the same object.

For example, to query the NT event log for errors only and separate the result of each query under three different categories: Application Errors, Security Errors and System Errors:

- 1 Rename the default WMI SQL to "Application Errors" and modify the SQL statement to:

```
SELECT * FROM Win32_NTLogEvent WHERE  
LogFile = "Application" AND Type = "Error"
```

- 2 Add a WMI SQL property and rename it to "Security Errors". Set the WMI SQL statement to:

```
SELECT * FROM Win32_NTLogEvent WHERE  
LogFile = "Security" AND Type = "Error"
```

- 3 Add a WMI SQL property and rename it to "System Errors". Set the WMI SQL statement to:

```
SELECT * FROM Win32_NTLogEvent WHERE  
LogFile = "System" AND Type = "Error"
```

For Diagnostics to audit a WMI Component and return information about the component, you must provide at least one WMI SQL property for the component.

## Customizing the WMI Components List

To audit a WMI object not listed in the factory default list, you use the Customize feature to first add it to the list.

### To add a new component to the list:

- 1 Click Customize.
- 2 In the Customize dialog, click the check box for component you want to add.

- 3 Click Add.

---

*You can change the display name of the object to a more user-friendly name by entering the new name in the Display Name column. (Objects prefixed with a '\*', are objects containing a modified Display Name.)*

---

To add several objects at once, hold down the CTRL key and then click each object you want to select. Hold down the SHIFT key to select a range of files. Click Add to add the selected objects.

To add a new category you must edit the UserWMI.INI file and add it under the [Namespaces] section.

### To remove a component or category from the list:

Click a WMI component or category and then click Remove. You cannot remove any of the factory default WMI Objects from the list.

## WMI INI File Format

The list of WMI categories and components displayed in Diagnostics is defined by the MqWMI.INI and UserWMI.INI files.

- MqWMI.INI provides the list of default WMI categories and their components. Settings in the MqWMI.INI are referred to as factory settings and cannot be removed using Diagnostics.
- UserWMI.INI contains the categories and components added using Diagnostics.

If you edit the INI files manually, you must follow to the file format described below so that Diagnostics can load these files. Categories (namespaces) must be added manually to the UserWMI.INI file

following the format outlined below. To add a category, you must add an entry under the [Namespaces] section:

Category (Namespace) entry:

```
[Namespaces]namespace=type:
display name:namespace server path
```

where type can have two possible values:

- 0 (Default)
- 1 (Custom)

For example:

```
[Namespaces]CIMV2=0:Win32 Environment:
\\.\root\cimv2
```

To add a category's component list, you must add object (class) entries under its corresponding namespace section:

Component (Class) entry:

```
[namespace]class name=type:
displayname:SQLstatement
```

For example:

```
[CIMV2]Win32_DMAChannel=0:DMA
Channel:SELECT * FROM
Win32_DMAChannelWin32_IRQResource=0:IRQ
```

```
Resources:SELECT * FROM
Win32_IRQResourceStoppedManualServices=
1:Stopped Manual
```

```
Services:SELECT * FROM Win32_Service WHERE
StartMode = "Manual" AND State = "Stopped"
```

StoppedManualServices is an example of a custom class that you can add that adheres to the format guidelines.

## Auditing Database Information

The Diagnostics/db install option extends the auditing capabilities of Diagnostics to include database configuration information and database content. Diagnostics/db can collect information for any ODBC-compliant database such as Oracle, Microsoft SQL Server, and Microsoft Access.

Due to the nature of how ODBC is implemented, (multiple layers of programs and drivers communicating with each other), troubleshooting can be a challenge.

With Diagnostics/db, when an ODBC call fails, you no longer need to spend hours trying to determine whether it is a problem with client libraries, or a net protocol mismatch, or even a database engine not running, Diagnostics/db can collect all the information required to perform a proper diagnosis in minutes.

## ODBC Database Configuration

Diagnostics steps you through the process of specifying what to collect about a user's ODBC installation. The ODBC configuration information is grouped into categories:

**System DSNs** Data Source Name, registry security, description, system database, ODBC driver, User, DSN configuration settings, and more.

**User DSNs** Data Source Name, registry security, description, system database, ODBC driver, User, DSN configuration settings, and more.

**File DSNs** Data Source Name, and file information (location, size, attributes).

**ODBC Drivers** File version information (file name, location, file version, attributes, and more), API level, ODBC driver version, SQL level, and more.

## Database Connection Information

Diagnostics/db can retrieve data from any database table a user has access to read. The Database Connection Editor provides three ways to retrieve data from a database:

**By selecting tables** Diagnostics returns the content of the selected tables.

**By selecting stored procedures** Diagnostics returns the result of running the stored procedure.

**By specifying an SQL statement** Diagnostics returns the result of running the SQL statement.

With Database Connection Information, you can add new connections and edit or remove existing connections.

### To add a database connection:

- 1 In the Database Connections dialog, click Add.

The Database Connection Editor opens to allow you to create a new Database Connection.

- 2 In the Type list, click the type of connection.
- 3 For an ODBC connection, click Browse and then click the type of DSN.

**User DSN** Click a user DSN and click OK.

**System DSN** Click a system DSN (if any) and click OK.

**File DSN** In the Look in Drive list click a drive, then click a file DSN and click OK.

**SQL Server** Enter the names of the SQL server and the database, a user ID and password, and then click OK. You can use variables in any of the fields.

**No DSN** Enter a connection string that will open the database. For example:

```
DRIVER=SQL Server;SERVER=YourServer;  
UID=YourLogonName;PWD=YourPassword;  
APP=Microsoft@Access;WSID=YOURMACHINE;  
DATABASE=YOURDATABASE)
```

You can use variables in the connection string.

## Selecting Data to Collect

You can select the data to be collected from the connection as Tables, Procedures and SQL Statements.

### To select tables:

- To select all the tables in the DSN, click the check box beside ODBC Tables in the list of tables.
- To select only certain tables in the DSN, click the check boxes beside the tables you want to include.

### To select procedures:

- To select all the procedures in the DSN, click the check box beside ODBC Procedures in the list of procedures.
- To select only certain procedures in the DSN, click the check boxes beside the procedures you want to include.

**To enter SQL statements:**

- 1 Under SQL Statements, click Add to add a query to your connection.
- 2 Under Edit SQL Statement, type a name and SQL statement (for example: Select \* from tblAttachments).

---

*You can use variables in the SQL statement.*

---

- 3 Click Test to view the results of your query in your default Web browser.
- 4 When you are satisfied with the query, click Apply.

You can add more SQL Statements to your connection, and edit or remove existing ones.

## Collecting Diagnostics for IIS

You can collect information about the Web sites, virtual directories, FTP sites, and SMTP servers on an IIS Web server.

**To collect IIS diagnostics:**

- 1 In the profile tree (Details view), click Audit Preferences.
- 2 In the Value list of the Internet Information Server attribute, click True.

## Collecting Security Information

You can collect files, shares, and registry security information.

**To collect security information:**

- 1 In the profile tree (Details view), under Audit Preferences, click Security.
- 2 Set the Include File Security, Include Registry Security, or Include Share Security attribute to True.



# Chapter 5: Customizing Application Protection

When you protect an application, Diagnostics generates repair rules that specify how to detect and fix problems. For example, the repair rule for a file looks like this:

```
If
    Audit Status = Found AND
        Size (bytes) = 987,136
    Do Nothing
Else
    Fix it
```

So when you audit an application, if the file is not found or its size does not match the size found when it was protected, a problem is detected.

Repair rules are saved in a copy of the profile. This copy is created when you protect the application, and is stored on the local computer.

## About Repair Rules

The general form of a repair rule looks like this:

```
if ( condition )
    action1
else
    action2
```

condition is a logical expression that tests the values in an audit report.

actions are predefined actions such as Fix it, Display Message, and Do Nothing. Fix it depends on the type of object.

## Customizing Repair Rules

When you protect an application, Diagnostics generates default repair rules. You can replace the default repair rules with customized repair rules.

**To customize repair rules:**

- 1 In Diagnostics Console, generate repair rules in the original profile.
- 2 Edit the repair rules.
- 3 If you customized any of the repair conditions, lock the repair rules so the conditions are not overwritten when you protect an application.

## Generating Repair Rules

You can generate repair rules for files, ActiveX controls, shortcuts, environment variables, and registry values in the profile.

**To generate repair rules for specific items:**

- 1 In the profile tree, click Files, Registry, ActiveX Controls, Environment Variables, or Shortcuts.
- 2 Select one or more items.
 

Use the Shift and Ctrl keys to select multiple objects, or drag the pointer over the objects you want to select. To select by dragging, point to a blank area (for example, the whitespace after an item name) and then drag the bounding outline.
- 3 Click Self Repair and then click Auto Build.
 

Diagnostics Console generates default repair rules for the selected objects.

## Editing Repair Rules

**To edit a repair rule:**

- 1 In the profile tree, click Files, Registry, ActiveX Controls, Environment Variables, or Shortcuts.
- 2 Click an item (a file, ActiveX control, shortcut, environment variable, or registry entry).
- 3 Click Self Repair and click Build Condition.

## Defining Conditions

A condition is one or more expressions joined by And or Or. Each expression tests the value of an object property. For example:


Audit Status = Found AND  
Size (bytes) = 987,136

**To define a condition:**

- 1 Click Add.
- 2 Click in the Property box and select an object property. The Property box lists the object properties that can be used to build a condition.  
  
Use the Audit Status property to test whether the object was found during the audit.
- 3 Click in the Test box and select a logical test.
- 4 Click in the Value box.

The value you enter here is compared against the value in an audit report.

---

Click Get  to get the current value of a property.

---

**To test environment variables like PATH** Use the Contains test operator instead of the = operator. When Diagnostics gets the current value of the PATH environment variable, it gets the value

for the current process (Diagnostics Console). So the path to the Diagnostics installation directory is added to the start of the PATH variable.

## Defining Actions

### Action Arguments

The Display Message and Go to URL actions each have an argument. For Display Message, the argument is the text to display. For Go to URL, the argument is the URL.

### Action Captions

The caption is the text displayed on the button beside a problem in the Details view. The default caption is Fix it.

The width of the button is controlled by the Action column width preference. See “Self-Repair Preferences” on page 41.

### Delete it

Deletes registry entries.

**Delete it for Registry Keys** Diagnostics can delete a registry key and all of its descendants.

You do not need to add any condition to operate on keys. If the key exists, Diagnostics considers that the condition is met. If the key does not exist, the condition is not met.

**Delete it for Registry Values** If you want to delete the registry value regardless of its current value, do not specify any condition. If the value exists, Diagnostics will delete it.

## Display Message

Displays the message specified by the Argument field.

```
if ( condition )
    Display Message
    Argument = "Condition met!"
else
    Display Message
    Argument = "Condition failed!"
```

In the Problem Diagnosis view, the Fix All button does not execute Display Message actions. Only the Fix it button for a specific problem executes a Display Message action.

## Do Nothing

No action.

## Fix it

**Fix it for Files** Fix it for files extracts the file from the snapshot and puts it in the required location.

**Fix it for ActiveX Controls and Self-Registered Files** If the file is not registered, is the wrong version, or is missing, Diagnostics gets the file from the snapshot and registers it. If the file is already present on the computer but is just not registered, Diagnostics registers it.

**Fix it for Shortcuts** If a shortcut is broken, Diagnostics tries to fix it based on the path specified in the condition. But if the path does not

point to an existing file, Diagnostics scans the system for the first occurrence of a file with the same name and fixes the shortcut to point to that file.

**Fix it for Registry values** If a registry value does not meet the specified condition, Diagnostics updates the registry entry according to the criteria specified in the condition. Diagnostics can repair individual registry values only, not complete hierarchies.

**Fix it for Environment Variables** Fix it updates the value of the environment variables to match the value found when the application was protected.

## Go to URL

Starts the default browser and loads the URL specified in the Argument field.

In the Problem Diagnosis view, the Fix All button does not execute Go to URL actions. Only the Fix it button for a specific problem executes the Go to URL action.

## Rename it

Renames a file.

## Unregister it

Unregisters an ActiveX control.

## Setting Attribute Values

To edit the attributes of a repair rule:

- 1 In the profile tree, click Files, Registry, ActiveX Controls, Environment Variables, or Shortcuts.
- 2 Click an item (a file, ActiveX control, shortcut, environment variable, or registry entry).
- 3 Click Self Repair and click Build Condition.
- 4 In the Attributes list, click in the Value field to edit the attribute value.

## Auto Execute Action

If True, Diagnostics automatically executes the specified repair action.

## Description

Text displayed between the problem title and the Show Details section in the Problem Diagnosis view.

### Problem Description



## Enable Self-Repair

If True, Diagnostics applies the repair rule. If False, the rule is disabled.

## Locked

If True, the repair condition is not updated when you protect the application. By default, all repair rule conditions are regenerated when you protect the application.

## Problem Priority

By default, Diagnostics sorts problems by priority, with the highest priority problems appearing at the top of the list. Lower numbers indicate higher priority.

## Self-Repair Package


Specifies the snapshot file used to repair the problem. This attribute is automatically set when you protect an application.

## Target Directory

Specifies where to put a file on the user's machine when the problem is fixed. This attribute is automatically set when you protect an application.

TargetDirectory is typically set to the value of a variable such as `$(WinDir)` or `$(AppInstallDir)`.

## Title

Text displayed after the Problem: label for a problem. To type or edit a multi-line title, click .

## Locking Customized Repair Rules

If you customize any repair rule conditions, you must lock the repair rules so the conditions are not overwritten with the defaults generated when you protect an application.

The condition is the logical expression tested by the if...then statement.

## Self-Repair Preferences

**Action column width** Width (in pixels) of the column that holds the Fix it button.

**Diagnosis - Show Details** If True, Diagnostics displays problem details

**Enable Self-Repair** Set to True when you protect an application (repair is always enabled for protected applications).



# Chapter 6: Scheduling Jobs

Jobs allow you to schedule application protection and audits at the most convenient times for you (or for your users). Jobs also allow you to batch protect and audit.


Diagnostics uses the Windows Task Scheduler to schedule jobs. Task Scheduler starts each time you start Windows, and runs in the background. Task Scheduler is part of Windows 98, Me, 2000, and XP. On Windows 95 and NT 4.0 SP3+, Task Scheduler is an Internet Explorer component that you can install by using the Add/Remove Programs tool in Control Panel.

Jobs run as the Support Site user.


## Defining Jobs

You can define jobs to protect and audit applications on any computer with a licensed version of Diagnostics Agent. Jobs can run on a single computer, some computers, all the computers in a domain, or all computers in the network. For example, you can use a job to protect an application on every agent-licensed computer.


### To define a job:

- 1 On the Action menu, click New and then click Jobs.
- 2 In the box beside the task icon , type a name for the job.
- 3 In the Command list, click Protect or Audit.
- 4 In the Select Machines box, expand the Microsoft Windows Network and select the check boxes for the computers you want to protect.

To protect all PCs on all domains of the network, select the Microsoft Windows Network check box. To protect all computers on a given domain, select the domain check box.

- 5 In the Select Products box, select the check boxes for the products you want to protect.
- 6 In the Job view toolbar, click  to save the job.

After you save the job, you can either run it immediately or schedule it:

- To run the job, click  in the Job view toolbar.
- To schedule the job, click Scheduler (see “Scheduling Jobs” on page 44).

---

*The Select Machines box lists only the computers with a licensed version of Diagnostics Agent.*

*If all PCs in a domain are selected, the job runs on all agent-licensed PCs in the domain, which may not match the PCs selected when the job was defined. Just before it runs, the job dynamically finds all agent-licensed computers in the domain. Similarly, if all domains in a network are selected, the job runs on all agent-licensed PCs in the domain, and the job dynamically finds the agent-licensed PCs.*


*If you select a specific set of PCs, the job only runs on those PCs with an agent license.*

---

## Running Jobs

You can run jobs manually without scheduling them. This is handy for doing batch protects of applications on many machines when you don't want to repeat the protection at regular intervals. Scheduled tasks can be run manually as well.

### To run a job:

- In the Job view toolbar, click .

## Checking the Status of Jobs

### To check the status of a job:

- 1 In the console tree, expand Jobs and then expand the job definition. The starting date and time of each job is listed under the job definition.

#### Job Definition in the Console Tree



- 2 Click a job to display the Job Status view.

The Job Status view displays the status of the commands executed on each machine.

### To see more details for a specific command:

For example, you may want to know why a Protect command failed on a certain computer.

Click the command and then click Details.

### To refresh the display:

Click Reload.

## Scheduling Jobs




You can schedule a job to run daily, weekly, or monthly, and change the schedule for a task.

### To schedule a job:

- 1 If the job you want to schedule is not already open, expand Jobs in the console tree and click the job you want to schedule.
- 2 In the Job view, click Scheduler.

### To remove the schedule for a job:

- 1 If the job you want to schedule is not already open, expand Jobs in the console tree and click the job you want to schedule.
- 2 In the Job view, click Remove Schedule.

Unscheduled tasks  are represented by , and scheduled tasks by .

A protected computer stores a local copy of the profile. If the version of the profile stored on the Support Site is more recent, then Always use latest profile revision determines which profile an Audit command uses.

To	Do this
Use the most recent version of the profile on the Support Site	Select the Always use latest profile revision check box.
Use the version of the profile stored on the protected computer.	Click to clear the Always use latest profile revision check box.

# Chapter 7: Requests

## Working with Requests

A pending request is an audit or protect request that has not been picked up by the Diagnostics Agent on the target computer. If an audit or a protect request seems to be taking a long time to finish, check the Requests view.


In Progress requests are being processed by the Diagnostics Agents. In Progress requests are requests issued by scheduled jobs. A job request must be finished before any another request can be processed, while requests from a Diagnostics Console are processed independently in separate threads.

For example, while a job request is in progress, all requests from Diagnostics Consoles are pending.

### To view the list of requests for a computer:

- 1 In the console tree, expand the Entire Network and locate the computer.
- 2 Expand the computer and click Requests.

### To delete a pending request:

Click the request and then click .

## Troubleshooting Pending Requests

When there are no In Progress requests and one or more requests are pending, there a number of things you can check before you delete the pending requests:


- Is the target computer on and connected to the network?
- Is the Diagnostics Agent is running on the target computer? Does the target computer have an agent license or was it revoked?
- Has the server run out of licenses? Try disconnecting all other users from the Support Site share and try again.
- Is the MQ Message Broker service is running? Is it running with the correct credentials?
- Is TriMon.exe running with the correct credentials? (Use the dcomcnfg.exe utility to check the Distributed COM Configuration Properties.)
- Is the Support Site is still available over the network? Does the Support Site User have enough privileges to access the Support Site?



# Chapter 8: Configuring

Diagnostics provides options for configuring Support Site, setting event logging options, registering products, and getting and revoking licences. These option settings are shared by all agents and consoles.

## To set options:

On the Action menu, click Options, or click  on the console toolbar.

When you change any of the options, the changes apply to all agents and consoles. Agents are sent a notification message of the changes, and consoles pick up the changes the next time they start up.

## Moving the Support Site

When you change the location of the Support Site, all agents and consoles are automatically notified. If the notifications fail and the agents and consoles cannot automatically update their Support Site settings, they can do it manually through the Options dialog.

You can also move the data in your Support Site to another Support Site.

### To move Support Site data to another Support Site:

- 1 In the Support Site Path box, enter the path to the other Support Site.
- 2 In the dialog that appears, click the Move all data from your current Support Site to the new location check box.

- 3 If you want agents and consoles installed from the current Support Site to switch to the other Support Site, click the Notify all clients of the change in Support Site location check box.

### To switch to a different Support Site:

- 1 In the Support Site Path box, enter the path to the other Support Site.
- 2 In the dialog that appears, click the Notify all clients of the change in Support Site location check box.

When you don't select the Move all data from your current Support Site to the new location check box, you switch to using the other Support Site and its data.

## The Support Site User Account

The Support Site user account is used to:

- Audit and protect computers.
- Access the Support Site shared folder.
- Run jobs.
- Run the MQ Message Broker service.

The Support Site user account must have the appropriate privileges on each local computer to perform tasks such as auditing and protecting a computer. The Support Site user should be a Domain Administrator that has local Administrative privileges on each computer.

To verify that the Support Site user account has access to the SupportSite from a computer, log on to Windows with that user account and try to copy a file to and from the Support Site shared folder.

## Event Logging

By default, agents and consoles log events on the local computer. On Windows NT, 2000, and XP, events are logged to the Event Log. On Windows 95, 98, and Me, events are logged to a text file.

On Windows NT, 2000, and XP, you can log all events to the Event Log on a central server.

**To log all events to the Event Log on a central server:**

- 1 Click the Log events to a central server check box.
- 2 Type the computer name of the central server.

By default, settings changes are applied only to new installations of Diagnostics Agent and Diagnostics Console.

**To apply new event logging settings to all installed agents and Consoles:**

Before you click OK, click the Apply new settings to all Clients check box. Agents receive a notification message of the changes, and consoles pick up the changes at their next startup.

## Maintenance

Diagnostics can automatically clean up old audits and snapshots in the Support Site.

To do this, set the maximum number of snapshots and audits that you want to keep for each protected application. When the number of snapshots or audits exceeds the maximum, the oldest snapshot or audit is deleted.

## Licensing

Each Diagnostics product (Diagnostics Console, Diagnostics Agent, Diagnostics/db) requires a serial number and a unique license key. The serial number is used to register a product. After you register a product, you can get license keys for the product. Typically, this is done during setup.

## Registering Products

Sometimes you need to register a product after it has been installed. For example, you may installed a product but only later decided to purchase licenses.

**To register a product:**

- 1 Click the License Information tab.
- 2 In the Serial Number box, enter the serial number and click Register.

If you have an evaluation version of a product, you may need to unregister it when the evaluation period ends.

**To unregister a product:**

- 1 Click the License Information tab.
- 2 In the list of products, click the product.
- 3 Click Unregister.

**To get a License Key for a product:**

- 1 Click the License Information tab.
- 2 In the list of products, click the product.
- 3 Click Get License.

## Revoking Licenses

If someone is on an extended vacation or leave of absence, you can revoke the license so someone else can use Diagnostics.

### **To revoke a user's license for a product:**

- 1 Click the License Usage tab.
- 2 Expand the product and click the user's computer.
- 3 Click Revoke.



# Appendix A: Technical Support

For Technical Support enquiries, contact your reseller. Alternatively, you can contact the Vector Networks Technical Support departments using the following details:

## US or Canada

**Phone:** 770 622 2850

**Toll Free:** 800 330 5035

**Fax:** 770 495 6214

**Email:** [support@vector-networks.com](mailto:support@vector-networks.com)

## Rest of the World

**Phone:** +44 (0) 1827 67333

**Fax:** +44 (0) 1827 67068

**Email:** [support@vector-networks.co.uk](mailto:support@vector-networks.co.uk)



# Index

## A

- Action column width 41
- Action Menu 2
- ActiveX Controls
  - Auditing 29
  - Fixing 39
  - Imported from MSI Packages 11
  - Loaded on Demand vs. Startup 12
  - Unregistering 39
- Applications
  - Auto-profiling 12
  - Diagnosing Problems 14
  - Install Packages 11–12
  - Protecting 1, 13–14
  - Restoring 14
  - Settings *See* Change Analysis
  - Windows Installer Packages 10–11
- Audit Status Property 38
- Auditing 16–17
- Audits
  - ActiveX Controls 29
  - Auditing PCs 16–17
  - Batch Auditing 17
  - Comparing 18–20
  - Console Tree 3
  - Copying Files 30–31
  - Database, ODBC 33–35
  - Details View 3
  - Files 27–29
  - Filtering Differences 19–20
  - Hardware Information 27
  - Highlighting Differences 20
  - Internet Information Services 35
  - Limiting Number of 48
  - Operating System Information 27
  - Printing 20

- Printing Change Analysis Reports 20
- Registry Keys and Values 29–30
- Requests 3
- Scheduling 43–44
- Searching CDROMs 28
- Searching Network Drives 28
- Security Information 35
- Selecting Data 34
- Self-Registered Files 11
- Shortcuts 30
- System Resource Information 27
- Variables 18
- Viewing 17–18
- Viewing Copied Files 18 with Windows Management Instrumentation 31–33
- Auto Execute Action 40
- Auto-profiling 10, 12

## B

- Batch Jobs
  - Auditing PCs 17
  - Protecting Applications 14

## C

- Cancelling Jobs 44
- CDROM Drives 28
- Change Analysis
  - Described 13, 15
  - Details View 3
  - Overview 1
  - Reviewing Configuration Changes 18–19
- Computer Settings *See* Change Analysis
- Conditions, Repair Rule 38
- Configuration Changes *See* Change Analysis

- Console Tree 2–3

## D

- Databases
  - Configuration Information 33
  - Database System Name (DSN) 34
  - Diagnostics/db 5
  - Getting Data from 34–35
- dcomcnfg.exe utility 45
- Delete it Action 38
- Description Attribute 40
- Diagnosis - Show Details 41
- Diagnostics Agent
  - Creating Silent Installs 6
  - Event Logging 5, 48
  - Installing 6
  - Overview 2
  - Registering 48
  - Setting Options 6
- Diagnostics Console
  - Action Menu 2
  - Console Tree 2–3
  - Elements of 2–4
  - Event Logging 5, 48
  - Installing 6
  - Overview 1–2
  - Problems 40
  - Registering 48
- Diagnostics/db 5
- DLL Dependencies 28
- DLLs *See* ActiveX Controls
- Drivers, ODBC 34
- DSN 34

## E

- Enable Self-Repair 41
- Environment Variables and Auto-profiling 13
- Fixing 39

Referencing in Profiles 21, 25  
Event Logging 5, 48

## F

File DSNs 33  
Files  
  Auditing 27–29  
  Copying into Audit Report 10, 30–31  
  Fixing 39  
  Renaming 39  
  Security Information 35  
  Selecting with Regular Expressions 26–27  
  Version Information 29  
  Viewing, in Audit Reports 18  
Fixing  
  Application Problems 14  
  Environment Variables 39  
  Files 39  
  Registry Entries 39  
  Self-Registered Files 39  
  Shortcuts 39

## H

Hardware Information 27  
HKEY\_CLASSES\_ROOT 11  
HKEY\_CURRENT\_USER 13

## I

INI Variables 22–23  
Install Packages 10–12  
InstallShield 11  
Internet Information Services (IIS) 35

## J

Jobs  
  Console Tree 3  
  Details View 4  
  Scheduling 43–44  
  Status View 4

## L

Licensing 48–49  
Locked Attribute 40

## M

Maintenance 48  
Messages 39  
MQ Message Broker Service 45  
MSI Packages 10–11

## N

Network Drives 28

## O

OCXs *See* ActiveX Controls,  
  Self-Registered Files  
ODBC  
  Configuration 33  
  Drivers 34  
  *See Also* Databases  
Operating System Information 27

## P

Private 2, 10  
Problem Priority Attribute 40  
Problems  
  Description 40  
  Details View 4  
  Diagnosing 14  
  Priorities 40  
  Showing Details 41  
  Title text 40  
Profiles  
  ActiveX Controls 29  
  Auto-profiling Options 10  
  Building Manually 15–16  
  Console Tree 2  
  Copying Files 10, 30–31  
  Database, ODBC 33–35  
  Details View 3  
  DLL Dependencies 28

Environment Variables 13  
Files

  Adding 27–29  
  Selecting with Regular Expressions 26–27  
Filtering Files 10  
Filtering Registry Keys 10  
Generating from Installed Applications 12  
Hardware Information 27  
Importing Install Packages 11–12  
Importing Windows Installer Packages 10–11  
Installed Applications 12  
Internet Information Services 35  
  Methods for Building 9  
  Operating System Information 27  
  Private 2, 10  
  Public 2  
  Reducing Size of 10  
  Registry Keys and Values 29–30  
  Restricting Registry Keys 30  
  Security Information 35  
  Self-Registered DLLs 11  
  Shortcuts 30  
  System Resource Information 27  
  Variables  
    Defining 21–26  
    Using to Locate Files 28  
  Windows Management Instrumentation 31–33  
  Windows Platforms 9  
Protecting  
  All PCs 43  
  Applications 13–14  
  Overview 1  
  Requests 3  
  Scheduling 43–44  
Public 2

## R

Registering Products 48

- Registry
  - Auditing 29–30
  - Deleting Keys 38
  - Deleting Values 39
  - Fixing Entries 39
  - Security Information 35
- Registry Variables 22
- Regular Expressions 26–27
- Repair Rules
  - Action Arguments 38
  - Action Captions 38
  - Audit Status Property 38
  - Auto Execute 40
  - Customizing 37
  - Defining Conditions 38
  - Deleting Registry Keys 38
  - Deleting Registry Values 39
  - Displaying Messages 39
  - Editing 38
  - Enabling 40
  - Fixing
    - ActiveX Controls 39
    - Environment Variables 39
    - Files 39
    - Registry Entries 39
    - Self-Registered Files 39
    - Shortcuts 39
  - Generating 37
  - Locking 40, 41
  - Overview 37
  - Problem Priorities 40
  - Renaming Files 39
  - Unregistering ActiveX Controls 39
  - URLs, Go to 39
- Requests 3

## S

- Scheduling Jobs 43–44
- Security Information 35
- Self-Registered Files
  - Defined 9
  - Fixing 39
  - in MSI Packages 11
- Self-Repair Package Attribute 40

- Self-Repair *See* Repair Rules
- Serial Numbers 48
- Shares, Security Information 35
- Shortcuts
  - Adding to Profile 30
  - Fixing 39
- Silent Installs 6
- Snapshots
  - Console Tree 3
  - Defined 13
  - Limiting Number of 48
  - Repairing Problems 14
- SQL Statements 35
- Stored Procedures 34
- Support Site
  - Audit Reports 17
  - Maintenance 48
  - Moving 47
  - Overview 2
  - User Account 5, 43, 47
- System Configuration Changes
  - See* Change Analysis
- System DSNs 33
- System Resource Information 27

## T

- Tables, Database 34
- Target Directory Attribute 40
- Title Attribute 40
- TriMon.exe 45

## U

- Unregistering ActiveX Controls 39
- URLs 39
- User Account, Support Site 5, 47
- User DSNs 33
- User-Defined Variables 23

## V

- Variables

- Defining 21–26
- Extracting Path Names 25
- Predefined 23–25
- Using to Locate Files 28
- Visual Basic Projects 11

## W

- Windows Installer Packages 10–11
- Windows Management
  - Instrumentation 27, 31–33
- Windows Operating Systems 9
- Wise Installer 11

