CHAPTER 10

Using DFM Adapters

Adapters integrate Device Fault Manager (DFM) with its environment and can be configured using either the GUI or the command line. The GUI allows you to do essential adapter configuration, while the command line lets you fine-tune the adapter. Both configuration methods modify the same underlying configuration files.

These topics explain how to configure the adapters:

Note Starting and stopping the adapters, with the exception of the HPOV-NetView Adapter, is controlled by CiscoWorks. The HPOV-NetView Adapter is started and stopped automatically by the OpenView or NetView network management system.

Adapter Overview and Uses

DFM adapters provide a means of communication between a domain manager and the networked system, as described in the "DFM Adapters" section on page 1-9. You can configure these adapters to suit your needs—for example, to send notifications to certain mailboxes, to forward and receive traps to and from specified network management systems, or to store alarms in files. Most DFM adapters can be configured using the DFM administration menus, although using the command line can sometimes provide fine-tuning.

To understand how adapters work with DFM, they are categorized as:

- Notification adapters, which receive event notifications from a domain manager and forward the event information to designated recipients.
- Event adapters, which monitor the status of managed elements and forward events to the domain manager for analysis.
- Special adapters, which perform specialized functions.

Notification Adapters and their Uses

You can configure notification adapters to forward event information to recipients on the network. DFM provides three types of notification adapters:

- File Notifier Adapter, which logs alarms detected by the DFM server and forwards them to a file. A file is the only valid recipient for this adapter. Use this adapter to create a historical file containing all alarms generated by DFM.
- Trap Notifier Adapter, which converts DFM alarms into SNMP trap messages and forwards the traps to recipients. You can specify the recipients, such as network management systems or other domain managers, using an IP address or a system name. Use this adapter to send DFM alarms to another application or NMS for additional processing or display. The format of the converted SNMP trap messages is provided in Appendix C, "SNMP Trap Notifier MIB."
- Mail Notifier Adapter, which uses SMTP to send mail notifications to recipients. Like the Trap Notifier Adapter, you can specify the recipients—in this case, an email address. Use this adapter to generate asynchronous email notifications when one or more alarm conditions occur. For example, you could use the Mail Notifier Adapter to send an epage to a specified list of recipients. For less serious faults, you might want to forward the notifications to a list of email addresses.

Event Adapters and their Uses

Event adapters include the HPOV-NetView Adapter, which forwards traps (sent from managed devices to the NMS) from an HP OpenView or NetView network management system to DFM. Use this adapter when you want DFM to monitor faults on devices managed by HP OpenView or NetView. This adapter can also be used with remote versions of HP OpenView or NetView. For more information, refer to *Installation and Setup Guide for Device Fault Manager*.

Special Adapters and their Uses

Special adapters are in a class of their own and operate somewhat differently from notification and event adapters, performing special functions or enabling services required by DFM. Special adapters include:

- SNMP Trap Adapter, which listens for traps (sent to DFM from managed devices) on a user-specified port and forwards the traps to specified destinations. It can receive SNMP V1 and V2 traps. Supported traps for forwarding are SNMP V1 (V2 traps received are forwarded as V1 traps). Sample uses for this adapter are:
  - Allowing DFM to coexist with another trap receiving application, such as an NMS, on the same server. Traps can be received by the SNMP Trap Adapter and forwarded to the other application on a different port (and vice versa).
  - Having an NMS forward traps to DFM for processing.
  - Having DFM listen for traps from devices and forward the traps to an NMS that does not support trap forwarding.
- RME Adapter, which synchronizes the list of managed devices in the Essentials inventory with the DFM inventory. Use this adapter when you want DFM to monitor faults on devices managed by Resource Manager Essentials. This adapter can also be used with remote versions of Essentials. For more information, refer to *Installation and Setup Guide for Device Fault Manager.* For more information on this adapter, including the supported versions of Resource Manager Essentials, refer to Chapter 6, "Working with the DFM Inventory."

Configuring and Starting Adapters

Table 10-1 summarizes which adapters you must configure, whether you can use the GUI and/or command line to configure the adapter, and whether you must manually start the adapter.

Note Whenever you configure any adapter using the command line, you must manually stop and restart the adapter. Adapters configured with the DFM administration menus do not need to be stopped and restarted.

Table 10-1 Configuring and Starting Adapters

Adapter Type and Name	Must Be Configured Before Use	Can Be Configured Using...		Automatically Starts with CiscoWorks[1]
		DFM GUI	CLI	
Notification Adapters				
File Notifier	No	Yes	Yes	No
Trap Notifier	No	Yes	Yes	No
Mail Notifier	Yes	Yes	Yes	No
Event Adapters				
HPOV-NetView	No	No	Yes	Yes, if HP OpenView or NetView are installed on the same machine as the adapter
Special Adapters				
SNMP Trap	Yes	Yes	Yes	Yes
RME Adapter	No	N/A, but can be *started* using the CiscoWorks GUI	N/A	Yes, if Essentials is installed on the same machine as the adapter

1. You can configure an adapter to automatically restart when CiscoWorks restarts. Refer to the "Registering and Unregistering the DFM Processes Using pdcmd" section on page 11-15.

Note For more information on the RME Adapter, refer to Chapter 6, "Working with the DFM Inventory."

When an adapter starts, its configuration information is loaded from a file. The configuration files have an extension of /conf and are located in directories under the directory *NMSROOT*/objects/smarts/conf.

Caution Do not rename the configuration files or move them. Adapters that cannot find the required configuration file will not run.

All of the adapters, except the HPOV-NetView Adapter, can be configured using either the DFM GUI or the command line. Both methods modify the same underlying configuration files.

Using the DFM GUI to Configure Adapters

You can configure the following adapters using the DFM GUI. After configuring an adapter using the GUI, you do not need to restart the CiscoWorks process associated with the adapter.

Adapter	DFM GUI Choice	For more information, refer to...
File Notifier Adapter	**Device Fault Manager> Administration> Fault Notification > File Notifier**	Using the DFM GUI to Configure the File Notifier Adapter, page 10-18
Trap Notifier Adapter	**Device Fault Manager>Administration> Fault Notification >Trap Notifier**	Using the DFM GUI to Configure the Trap Notifier Adapter, page 10-22
Mail Notifier Adapter	**Device Fault Manager > Administration> Fault Notification > Mail Notifier**	Using the DFM GUI to Configure the Mail Notifier Adapter, page 10-28
SNMP Trap Adapter	**Device Fault Manager > Administration > Trap Configuration > Trap Receiving** **Device Fault Manager > Administration > Trap Configuration > Trap Forwarding**	Using the DFM GUI to Configure the SNMP Trap Adapter, page 10-34

Using the Command Line to Configure Adapters

To change the configuration of an adapter using the command line:

Step 1 Locate the configuration file for the given adapter. The configuration files are located in the *NMSROOT*/objects/smarts/conf directory; the filenames are listed in the sections that provide detailed information about the adapters.

Note Save a copy of any configuration file you intend to change.

Step 2 Edit the configuration file, changing only the parameters listed in Table 10-2 through Table 10-5. When changing notification adapter subscriptions, see the "Changing Subscriptions for Notification Adapters" section on page 10-10.

Step 3 Save the edited configuration file(s).

Step 4 Stop and restart the appropriate CiscoWorks process (using **Server Configuration > Administration > Process Management**), as follows:

- DfmFileNotifier (for the File Notifier Adapter)
- DfmTrapNotifier (for the Trap Notifier Adapter)
- DfmMail Notifier (for the Mail Notifier Adapter)
- DfmChangeProbe (for the RME Adapter)
- DfmServer (for the SNMP Trap Adapter)

If you have to stop the DfmServer process, you will first have to stop any running processes that depend on the DfmServer process.

Step 5 If necessary, register the adapter for automatic startup by running these commands to unregister and re-register the adapters. *NMSROOT* is the directory where CiscoWorks is installed (for Solaris, the default value for *NMSROOT* is /opt/CSCOpx, and for Windows, the default value is C:\Program Files\CSCOpx).

Note If you want to specify other options, do so in one pdcmd instance. For more information on other options, refer to the "Registering and Unregistering the DFM Processes Using pdcmd" section on page 11-15.

On Solaris, the commands are:

- For DfmFileNotifier (the second command is one line):

```
# NMSROOT/bin/pdcmd -u DfmFileNotifier
# NMSROOT/bin/pdcmd -r DfmFileNotifier -d DfmServer
-e NMSROOT/objects/smarts/bin/sm_notify -f "--adapter=filelog --output=sm_file_notifier"
```

- For DfmTrapNotifier (the second command is one line):

```
# NMSROOT/bin/pdcmd -u DfmTrapNotifier
# NMSROOT/bin/pdcmd -r DfmTrapNotifier -d DfmServer -e
NMSROOT/objects/smarts/bin/sm_notify -f "--adapter=trap --output=sm_trap_notifier"
```

- For DfmMailNotifier (the second command is one line):

```
# NMSROOT/bin/pdcmd -u DfmMailNotifier
# NMSROOT/bin/pdcmd -r DfmMailNotifier -d DfmServer
-e NMSROOT/objects/smarts/bin/sm_notify -f "--adapter=mail --output=sm_mail_notifier"
```

- For DfmChangeProbe (the second command is one line):

```
# NMSROOT/bin/pdcmd -u DfmChangeProbe
# NMSROOT/bin/pdcmd -r DfmChangeProbe -d EssentialsDbEngine,EssentialsDbMonitor
-e NMSROOT/bin/cwjava -f "-Xnoclassgc com.cisco.nm.dfm.changeprobe.DfmChangeProbe”
```

On Windows, the commands are:

- For DfmFileNotifier (the second command is one line):

```
# NMSROOT\bin\pdcmd.exe -u DfmFileNotifier
# NMSROOT\bin\pdcmd.exe -r DfmFileNotifier -d DfmServer
-e NMSROOT\objects\smarts\bin\sm_notify.exe -f "--adapter=filelog
--output=sm_file_notifier"
```

- For DfmTrapNotifier (the second command is one line):

```
# NMSROOT\bin\pdcmd.exe -u DfmTrapNotifier
# NMSROOT\bin\pdcmd.exe -r DfmTrapNotifier -d DfmServer -e
NMSROOT\objects\smarts\bin\sm_notify.exe -f "--adapter=trap --output=sm_trap_notifier"
```

- For DfmMailNotifier (the second command is one line):

```
# NMSROOT\bin\pdcmd.exe -u DfmMailNotifier
# NMSROOT\bin\pdcmd.exe -r DfmMailNotifier -d DfmServer
-e NMSROOT\objects\smarts\bin\sm_notify.exe -f "--adapter=mail --output=sm_mail_notifier"
```

- For DfmChangeProbe (the second command is one line):

```
# NMSROOT\bin\pdcmd.exe -u DfmChangeProbe
# NMSROOT\bin\pdcmd -r DfmChangeProbe -d EssentialsDbEngine,EssentialsDbMonitor
-e NMSROOT\bin\cwjava -f "-Xnoclassgc com.cisco.nm.dfm.changeprobe.DfmChangeProbe"
```

To start or stop individual adapter processes, refer to the "DFM and CiscoWorks Processes" section on page 11-12.

Changing Subscriptions for Notification Adapters

The File, Trap, and Mail Notifier Adapter configuration files contain a SubscribesTo parameter, which specifies the devices and notifications the adapter should monitor. This allows you to tailor the adapters to track only the information you want to track.

DFM provides two methods for specifying the events to which a notification adapter should subscribe:

- Specifying a subscription profile, as described in the "Specifying a Notification Adapter Subscription Profile" section on page 10-11
- Specifying subscription choices, as described in the "Specifying a Notification Adapter Subscription Choices" section on page 10-12

Note You can only use one method in a single configuration file.

Specifying a Notification Adapter Subscription Profile

This method allows you to filter what is monitored by notification adapters by specifying a subscription profile (profileName) to be used by the adapter. When you use this method, the subscription profile—rather than the adapter configuration file—contains the list of classes, instances and events you want the notification adapter to monitor. (Subscription profiles can be changed as described in the "Changing Your Subscription Profile" section on page 13-22.) Only one subscription profile can be used by a notification adapter.

```
SubscribesTo =
{
    GA_ProfileSubscription::Descriptive-Text
    {
        profileName = "filename"
    }
}
```

Note that this code fragment specifies GA_*Profile*Subscription, while GA_*Choice*Subscription is used in the "Specifying a Notification Adapter Subscription Choices" section on page 10-12. This method allows you to specify two parameters:

Descriptive-Text	User-specified text (for example, to help you remember which profile you are using)
profileName	The *filename* of an existing subscription profile. If you specify default, the default subscription profile (named PROFILE-default) will be used. Modified subscription profiles always use the name format PROFILE-*username* (*username* is your CiscoWorks username).

For example, this code fragment from the File Notifier Adapter would use the default subscription profile:

```
SubscribesTo =
{
    GA_ProfileSubscription::File-Notifier-DefaultProfile
    {
        profileName = "default"
    }
}
```

For more information on subscription profiles, see the "Changing Your Subscription Profile" section on page 13-22.

Specifying a Notification Adapter Subscription Choices

This method allows you to filter what is monitored by notification adapters by providing class, instance and event details in the adapter configuration file itself. When you use this method, the adapter configuration file—rather than a subscription profile—contains the list of classes, instances and events you want the adapter to monitor.

This section describes:

- Using Subscription Choices, page 10-12
- Examples of Subscription Choices, page 10-14

Using Subscription Choices

Use this code fragment in notification adapter configuration files to specify the classes, devices, and events you want to monitor:

```
GA_ChoiceSubscription::Descriptive-Text
    {
    # Subscribe to events whose class, instance, and event
    # names match the given pattern.
         className = "class"
          instanceName = ".*"
         eventName = "event"
        aggregates = {TRUE|FALSE}
          symptoms = {TRUE|FALSE}
    }
```

Note that this code fragment specifies GA_*Choice*Subscription, while GA_*Profile*Subscription is used in the method described in the "Specifying a Notification Adapter Subscription Profile" section on page 10-11. The device parameters for subscriptions include:

Descriptive-Text	User-specified text (for example, to help you remember which profile you are using)
className	Type of managed element (for example, Chassis). Valid classes are listed in Appendix E, "Valid Classes, Instances, and Events."
instanceName	Objects that describe the managed element (for example, NumberOfSlots, NumberOfPowerSupplies, ChassisType, device IP address).
eventName	Name of compound or symptom to be reported for the managed element. Valid events are listed in Appendix E, "Valid Classes, Instances, and Events."
aggregates	Compound events. If set to TRUE, sends a notification when the selected compound events occur.
symptoms	Symptomatic events. If set to TRUE, sends a notification when the selected symptomatic events occur.

This method is normally used with the Mail Notifier Adapter and Trap Notifier Adapter. The File Notifier Adapter is ready to use; normally, you will only want to enable the File Notifier Adapter to create one inventory for all generated alarms.

Examples of Subscription Choices

These examples shows the kind of monitoring granularity you can achieve using this method.

- In this example, the Mail Notifier Adapter will report all Operational Exception aggregates (compound events) that occur on any router managed by DFM:

```
SubscribesTo =
{

    GA_ChoiceSubscription::Router-All-Subscriptions
    {
    # Subscribe to events whose class, instance, and event
    # names match the given pattern.
         className = "Router"
          instanceName = ".*"
         eventName = "OperationalException"
        aggregates = TRUE
          symptoms = FALSE
    }
}
```

- In this example, the Mail Notifier Adapter will report when the state of any power supply managed by DFM is not normal:

```
SubscribesTo =
{

    GA_ChoiceSubscription::PowerSupply_Fault-All-Subscriptions
    {
    # Subscribe to events whose class, instance, and event
    # names match the given pattern.
         className = "PowerSupply_Fault.*"
          instanceName = ".*"
         eventName = ".*"
        aggregates = TRUE
          symptoms = TRUE
    }
}
```

- In this example, the Mail Notifier Adapter will report all high broadcast rate events that occur on any port managed by DFM:

```
SubscribesTo =
{

    GA_ChoiceSubscription::Port-All-Subscriptions
    {
    # Subscribe to events whose class, instance, and event
    # names match the given pattern.
         className = ".*Port.*"
          instanceName = ".*PORT-.*"
         eventName = "HighBroadcastRate"
        aggregates = TRUE
          symptoms = TRUE
    }
}
```

- In this example, the Mail Notifier Adapter will report all interface performance events that occur on any interface managed by DFM:

```
SubscribesTo =
{

    GA_ChoiceSubscription::Interface_Performance-All-Subscriptions
    {
    # Subscribe to events whose class, instance, and event
    # names match the given pattern.
         className = "Interface_Performance.*"
          instanceName = ".*"
         eventName = ".*"
        aggregates = TRUE
          symptoms = TRUE
    }
}
```

- In this example, the Mail Notifier Adapter will report all high utilization events that occur on any interface managed by DFM:

```
SubscribesTo =
{

    GA_ChoiceSubscription::Mail-All-Subscriptions
    {
    # Subscribe to events whose class, instance, and event
    # names match the given pattern.
         className = ".*"
          instanceName = ".*IF-.*"
         eventName = "HighUtilization"
        aggregates = TRUE
          symptoms = TRUE
    }
}
```

You can also configure a single notification adapter with multiple subscription choices. To do so, follow these rules:

- Each *Descriptive-Text* must be unique.
- Separate each GA_ChoiceSubscription code fragment with a comma (except for the last fragment).

In this example, the Mail Notifier Adapter will track all router compound events (aggregates), all switch compound events (aggregates), and all chassis symptoms:

```
SubscribesTo =
{

    GA_ChoiceSubscription::Router-All-Subscriptions
    {
    # Subscribe to events whose class, instance, and event
    # names match the given pattern.
         className = "Router"
          instanceName = ".*"
         eventName = ".*"
        aggregates = TRUE
          symptoms = TRUE
    },
```

```
    GA_ChoiceSubscription::Switch-All-Subscriptions
    {
    # Subscribe to events whose class, instance, and event
    # names match the given pattern.
         className = "Switch"
          instanceName = ".*"
         eventName = ".*"
        aggregates = TRUE
          symptoms = TRUE
    },
    GA_ChoiceSubscription::Chassis-All-Subscriptions
    {
    # Subscribe to events whose class, instance, and event
    # names match the given pattern.
         className = ".*"
          instanceName = ".*"
         eventName = ".*"
        aggregates = FALSE
          symptoms = TRUE
    }
}
```

Configuring the File Notifier Adapter

Note The File Notifier Adapter must be started before it can be used. See the "Configuring and Starting Adapters" section on page 10-5 for detailed procedures.

You can use either the GUI or command line to configure the File Notifier Adapter. The GUI lets you do essential configuration, while the command line lets you fine-tune the adapter.

Note The File Notifier Adapter logs all alarms once it is started. You cannot modify the adapter to store alarms for specified lengths of time.

Using the DFM GUI to Configure the File Notifier Adapter

When you enable the File Notifier Adapter using the DFM GUI, the adapter process is registered with CiscoWorks, and you can check its status using **Server Configuration > Administration > Process Management**. If you stop the adapter process from the CiscoWorks GUI, the adapter will be marked disabled in the DFM GUI.

To enable or disable logging and storing alarms detected by DFM in a log file:

Step 1 Select **Device Fault Manager > Administration > Fault Notification > File Notifier**.

Step 2 In the Adapter field, select **ENABLED** or **DISABLED** to start or stop event logging in the alarms file.

Step 3 Click **OK**.

This procedure does the following:

- Enables/disables the process and registers/unregisters the process with CiscoWorks. When the process is unregistered, you will no longer see it using **Server Configuration > Administration > Process Management**.
- Creates the File Notifier log file *NMSROOT*/objects/smarts/logs/sm_file_notifier.log.
- Creates the alarm file *NMSROOT*/objects/smarts/logs/DFM-alarms.log.

You do not have to restart CiscoWorks for your changes to take effect. To specify the types of notifications you want forwarded to the alarms file, use the command line (refer to the "Using the Command Line to Configure the File Notifier Adapter" section on page 10-19).

Using the Command Line to Configure the File Notifier Adapter

This topic describes the contents of the File Notifier Adapter file (*NMSROOT*/objects/smarts/conf/notifier/filelog_notify.conf). Table 10-2 lists the parameters you can change using the command line. Additional configuration of this file is normally not required.

```
#
# This is a configuration file which contains objects for the
# file notification adapter.
#
# Based on GNA - the Generic Notification Adapter framework
#
# $Id: filelog_notify.conf,v 1.1.2.5.2.2 2000/12/20 17:50:07 current Exp $
#


#
# The GNA notifier object.
#
GNA_Notifier::filelog-Notifier
{
    serverName = "DFM"

    # How long to wait, in seconds, before beginning to send events. Default
    # is 1 sec., to prevent a flood of notifications upon adapter startup.
    initialEventDelay = 1

    ProvidesAdditionalParams = Filelog_AdapterParams::file_Notifier-Parameters
    {
    }

    ReadsInputFrom = GA_SubscriberFE::Filelog_Subscriber-FrontEnd
    {
        # How long an event must remain active before the adapter sends a
        # notification, in units of seconds.
        eventSmoothingInterval = 0

        # Notification threshold; discard notifications with a certainty
        # below this value.
        minimumCertainty = 0.01
        SubscribesTo =
        {
            GA_ProfileSubscription::Filelog-Default-Profile-Subscriptions
            {
                profileName = "default"
            }
```

```
        }
    }

    # No user-serviceable parts below here.
    #
   start_stopRuleSet = "filelog/filelogInit.asl"

     adapterRuleSet = "filelog/filelogNotify.asl"

       PerformsSend = FilelogAction::filelog_Interface
}
```

File Notifier Adapter Command Line Parameters

Table 10-2 lists the File Notifier Adapter parameters you can change.

Table 10-2 File Notifier Adapter Parameters

Parameter	Description
serverName	Default name of the DFM to connect to. Note that this is the name of the DFM, not the name of the host it is running on. The default is DFM.
initialEventDelay	Time interval (in seconds) the adapter should wait before accepting events from the DFM. The default value is 1.
eventSmoothingInterval	Time (in seconds) that an event must remain in its current state before the adapter sends a notification. If the event is cleared before the event smoothing interval expires, it is not sent. The default value is 0 seconds.
minimumCertainty	Threshold above which notifications are logged. Any notification with a certainty below the threshold is discarded. Values may range from 0.0 to 1.0. The default value is 0.01.
SubscribesTo	Devices and types of notifications an adapter subscribes to. The default is all notifications and devices. Refer to the "Changing Subscriptions for Notification Adapters" section on page 10-10.

When the adapter starts, it creates the File Notifier log file *NMSROOT*/objects/smarts/logs/sm_file_notifier.log. Anytime you change the File Notifier Adapter file, you must restart the corresponding CiscoWorks process as described in the "DFM and CiscoWorks Processes" section on page 11-12. To configure the File Notifier Adapter for automatic startup with CiscoWorks, refer to the "Using the Command Line to Configure Adapters" section on page 10-8.

Alarm Log Example

The following is an example of the DFM alarm log file, *NMSROOT*/objects/smarts/logs/DFM-alarms.log:

```
02-Feb-2001 11:58:38 NOTIFY Switch::172.16.0.0::DiscoveryError 100% An error was
encountered during the last discovery probe of this System.

02-Feb-2001 12:00:19 CLEAR Switch::172.16.0.0::DiscoveryError An error was encountered
during the last discovery probe of this System.

02-Feb-2001 12:03:14 NOTIFY Switch::172.16.0.0::PowerSupplyException 100% System is
experiencing power supply problems.

02-Feb-2001 12:03:14 NOTIFY PowerSupply_Fault_CiscoStack::PWR-172.16.0.0/2
[]::StateNotNormal 100% The power supply is not in the normal state.

02-Feb-2001 12:05:09 NOTIFY Undiscovered::172.31.255.255::DiscoveryError 100% An error was
encountered during the last discovery probe of this System.
```

Configuring the Trap Notifier Adapter

Note The Trap Notifier Adapter must be started before it can be used. See the "Configuring and Starting Adapters" section on page 10-5 for detailed procedures.

You can use either the GUI or command line to configure the Trap Notifier Adapter. The GUI lets you do essential configuration, while the command line lets you fine-tune the adapter.

The contents of the MIB are provided in Appendix C, "SNMP Trap Notifier MIB."

Using the DFM GUI to Configure the Trap Notifier Adapter

When you enable the Trap Notifier Adapter using the DFM GUI, the adapter process is registered with CiscoWorks, and you can check its status using **Server Configuration > Administration > Process Management**. If you stop the adapter process from the CiscoWorks GUI, the adapter will be marked disabled in the DFM GUI.

To notify other hosts of DFM events and alarms:

Step 1 Select **Device Fault Manager > Administration > Fault Notification > Trap Notifier**.

Step 2 In the Adapter field, select **ENABLED** or **DISABLED** to start or stop trap notification.

Step 3 If you want to add a recipient, in the Add Recipients field, enter the hostname and port number of the machine you want to notify about events and alarms.

Step 4 If you want to remove a recipient, select the recipient from the Remove Recipient field.

Step 5 Select the Restart DFM Server check box if you want to restart the server when you click **OK**. (You must restart the DFM server for your changes to take effect.)

Step 6 Click **OK**.

This procedure does the following:

- Enables/disables the process and registers/unregisters the process with CiscoWorks. When the process is unregistered, you will no longer see it using **Server Configuration > Administration > Process Management**.
- Updates the Trap Notifier Adapter configuration file *NMSROOT*/objects/smarts/conf/notifier/trap_notify.conf.
- Creates the Trap Notifier log file *NMSROOT*/objects/smarts/logs/sm_trap_notifier.log.

If you did not select the Restart DFM Server check box, you must do so using **Server Configuration > Administration > Process Management**. To save information about all handled traps in the log file, specify the type of notifications

you want to listen for, or specify another community string outgoing traps, use the command line (refer to the "Using the Command Line to Configure the Trap Notifier Adapter" section on page 10-23).

Using the Command Line to Configure the Trap Notifier Adapter

This topic describes the contents of the Trap Notifier Adapter file (*NMSROOT*/objects/smarts/conf/notifier/trap_notify.conf). Table 10-3 lists the parameters you can change using the command line.

Note To create a log file for every trap handled by the adapter, set dumpTrap to TRUE. Traps are saved in the adapter log file, *NMSROOT*/objects/smarts/logs/sm_trap_notifier.log.

```
#
# This is a configuration file which contains objects for the
# trap notification adapter.
#
# Based on GNA - the Generic Notification Adapter framework
#
# $Id: trap_notify.conf,v 1.1.2.10.2.3 2004/02/20 19:43:57 guw Exp $
#

#
# The GNA notifier object.
#
GNA_Notifier::trap-Notifier
{
     serverName = "DFM"

     # How long to wait, in seconds, before beginning to send events. Default
     # is 1 sec., to prevent a flood of notifications upon adapter startup.
     initialEventDelay = 1


     ProvidesAdditionalParams = Trap_AdapterParams::trap_Notifier-Parameters
     {
     # For recipients using default community string "public":
     # a list of hosts (specified by host name,
     # UDP ports and SNMP version) to which the traps are sent.
     # To add more hosts to the list, follow this format
     # { {"host_name1", port_num1, "1"},
     #   {"host_name2", port_num2, "2"} }
```

```
  # Version number can only be 1 or 2. Currently only 1 is supported
 recipients = {}

  # DFM Patch/IDU 1.2.8 or later: For recipients using specified community string
  # a list of hosts (specified by host name,
  # UDP ports, SNMP version and Community String) to which the traps are sent.
  # To add more hosts to the list, follow this format
  # { {"host_name1", port_num1, "1", "community_string1"},
  #   {"host_name2", port_num2, "2", "community_string2"} }
  # Version number can only be 1 or 2. Currently only 1 is supported
 recipientsWithCommunity = {}
  }

  ReadsInputFrom = GA_SubscriberFE::trap_Subscriber-FrontEnd
  {
       # How long an event must remain active before the adapter sends a
       # notification, in units of seconds.
       eventSmoothingInterval = 0

       # Notification threshold; discard notifications with a certainty
       # below this value.
       minimumCertainty = 0.01
       SubscribesTo =
       {
      GA_ProfileSubscription::Trap-Default-Profile-Subscriptions
       {
      profileName = "default"
       }
   }
  }

  # No user-serviceable parts below here.
  #
   filterRuleSet = "trap-notify/trapFilterNotify.asl"
  adapterRuleSet = "trap-notify/trapNotify.asl"

    PerformsSend = Trap_NotifierAction::trap_NotifierInterface {
     dumpTrap = FALSE
 }
}
```

Trap Notifier Adapter Command Line Parameters

Table 10-3 lists the Trap Notifier Adapter parameters you can change.

Table 10-3 Trap Notifier Adapter Parameters

Parameter	Description
serverName	Default name of the server to connect to. Note that this is the name of the DFM, not the name of the host it is running on. The default is DFM.
initialEventDelay	Time interval (in seconds) the adapter should wait before accepting events from the DFM. The default value is 1.
recipients	Table of addresses to send the SNMP traps to. Each row consists of three different values: host or IP address, socket number, and SNMP version (usually equal to 1). The values are separated by commas and enclosed in curly braces. In turn, the rows are also separated by commas and enclosed in curly braces. The default is null.
recipientsWithCommunity	Table of addresses to send the SNMP traps to. Each row consists of four different values: host or IP address, socket number, SNMP version (usually equal to 1), and community string (“public” by default). The values are separated by commas and enclosed in curly braces. In turn, the rows are also separated by commas and enclosed in curly braces. The default is null. **Note** The recipientsWithCommunity table is only supported if you have downloaded and installed DFM 1.2 Patch/IDU 1.2.8 or later from Cisco.com at: http://www.cisco.com/cgi-bin/tablebuild.pl/cw2000-dfm.
eventSmoothingInterval	Time (in seconds) that an event must remain in its current state before the adapter sends a notification. If the event is cleared before the event smoothing interval expires, it is not sent. The default value is 0 seconds.
minimumCertainty	Threshold above which events are logged. Any notification with a certainty below the threshold is discarded. Values may range from 0.0 to 1.0. The default value is 0.01.

Table 10-3 Trap Notifier Adapter Parameters (continued)

Parameter	Description
dumpTrap	If set to TRUE, causes information to be sent to the log file for every trap handled by the adapter. The default value is FALSE. If set to TRUE, traps are saved to the adapter log file, *NMSROOT*/objects/smarts/logs/sm_trap_notifier.log.
SubscribesTo	Devices and types of notifications an adapter subscribes to. The default is all notifications and devices. Refer to the "Changing Subscriptions for Notification Adapters" section on page 10-10.

When the adapter starts, it creates the File Notifier log file *NMSROOT*/objects/smarts/logs/sm_trap_notifier.log. Anytime you change the Trap Notifier Adapter file, you must restart the corresponding CiscoWorks process as described in the "DFM and CiscoWorks Processes" section on page 11-12. To configure the Trap Notifier Adapter for automatic startup with CiscoWorks, refer to the "Using the Command Line to Configure Adapters" section on page 10-8.

Examples

The following examples show how to configure the Trap Notifier Adapter so that DFM traps can be forwarded to a number of recipients (such as network management systems).

This example forwards traps to the recipient *host_name1*:

```
#For case of one recipient:

ProvidesAdditionalParams =
Trap_AdapterParams::trap_Notifier-Parameters
     {
        recipients = {{"host_name1", 162, "1"}}
     }
```

This example forwards DFM traps to the recipients *host_name1*, *host_name2*, and *host_name3*:

```
#For case of  three recipients:

ProvidesAdditionalParams =
Trap_AdapterParams::trap_Notifier-Parameters
     {
        recipients = {{"host_name1", 162, "1"},
                       {"host_name2", port_num2, "1"},
                        {"host_name3", port_num3, "1"} }
     }
```

For information on editing subscriptions (the devices and types of notifications the adapter monitors), refer to the "Changing Subscriptions for Notification Adapters" section on page 10-10.

Configuring the Mail Notifier Adapter

Note The Mail Notifier Adapter must be configured and started before it can be used. See the "Configuring and Starting Adapters" section on page 10-5, along with the information in this topic, for detailed procedures.

You can use either the GUI or command line to configure the Mail Notifier Adapter. The GUI lets you do essential configuration, while the command line lets you fine-tune the adapter.

Note You cannot modify the content of the mail notification sent by this adapter. All recipients will receive the same information; you cannot configure the adapter to send different recipients different information.

Using the DFM GUI to Configure the Mail Notifier Adapter

When you enable the Mail Notifier Adapter using the DFM GUI, the adapter process is registered with CiscoWorks, and you can check its status using **Server Configuration > Administration > Process Management**. If you stop the adapter process from the CiscoWorks GUI, the adapter will be marked disabled in the DFM GUI.

To notify other users of DFM alarms:

Step 1 Select **Administration > Fault Configuration > Mail Notifier**.

Step 2 In the Adapter field, select **ENABLED** or **DISABLED** to start or stop trap notification.

Step 3 If you want to add a recipient, in the Add Recipient field, enter a comma-separated list of the recipients in the format *username@host.domain*.

Step 4 If you want to remove a recipient, select the recipient from the Remove Recipient field.

Step 5 In the SenderID field, enter the address associated with the adapter.

Step 6 In the MailServer field, enter the fully qualified domain name for the mail host.

Step 7 Select the Restart DFM Server check box if you want to restart the server when you click **OK**. (You must restart the DFM server for your changes to take effect.)

Step 8 Click **OK**.

This procedure does the following:

- Enables/disables the process and registers/unregisters the process with CiscoWorks. When the process is unregistered, you will no longer see it using **Server Configuration > Administration > Process Management**.
- Updates the Mail Notifier Adapter configuration file *NMSROOT*/objects/smarts/conf/notifier/mail_notify.conf.
- Creates the log file *NMSROOT*/objects/smarts/logs/sm_mail_notifier.log.

If you did not select the Restart DFM Server check box, you must do so using **Server Configuration > Administration > Process Management**. To log all mail messages or specify the types of notifications you want to monitor, use the command line (refer to the "Using the Command Line to Configure the Mail Notifier Adapter" section on page 10-29).

Using the Command Line to Configure the Mail Notifier Adapter

This topic describes the contents of the Mail Notifier file (*NMSROOT*/objects/smarts/conf/notifier/mail_notify.conf). Table 10-4 lists the parameters you can change using the command line.

Note To create a log file for every mail message sent by the adapter, set trace to TRUE.This information is saved in the adapter log file, *NMSROOT*/objects/smarts/logs/sm_mail_notifier.log.

```
# This is a configuration file which contains objects for the
# mail notification adapter.
#
# Based on GNA - the Generic Notification Adapter framework
#
# $Id:mail_notify.conf,v 1.1.2.3.2.2.2.4 2000/11/03 14:24:36 boaz Exp $
#
#
# The GNA notifier object.
#
GNA_Notifier::mail-Notifier
{
    serverName = "DFM"

    # How long to wait, in seconds, before beginning to send events. Default
    # is 1 sec., to prevent a flood of notifications upon adapter startup.
    initialEventDelay = 1
    # Additional parameters: A comma-separated list of recipients (who to
    # send to); the  address of the sender; and the fully qualified
    # domain name of the mail server.
    ProvidesAdditionalParams = MailAdapterParams::mail_Notifier-Parameters
    {
#       Recipients = "username1@host.domain,username2@otherhost.otherdomain"
#         SenderId = "daemon@localhost"
#       MailServer = "mailhost.domain"
    }
```

```
    ReadsInputFrom = GA_SubscriberFE::mail_Notifier-Subscriber-FrontEnd
    {
        # How long an event must remain active before the adapter sends
        # a notification, in units of seconds.
        eventSmoothingInterval = 0

        # Notification threshold; discard notifications with a certainty
        # below this value.
        minimumCertainty = 0.01
        SubscribesTo =
        {

            GA_ChoiceSubscription::Mail-All-Subscriptions
            {
            # Subscribe to events whose class, instance, and event
            # names match the given pattern.
                   className = ".*"
                instanceName = ".*"
                   eventName = ".*"
            # Include aggregates, but omit symptoms.
                 aggregates = TRUE
                   symptoms = FALSE
            }
        }
    }
    PerformsSend = MailAction::mail_NotifierInterface
    {
        # Trace all outgoing email messages to stderr
        trace = FALSE
    }   # No user-serviceable parts below here.
    #
    filterRuleSet = "mail-notify/mailFilter_Notify.asl"
    adapterRuleSet = "mail-notify/mail_Notify.asl"
}
```

Mail Notifier Adapter Command Line Parameters

Table 10-4 lists the Mail Notifier Adapter parameters you can change.

Table 10-4 Mail Notifier Adapter Parameters

Parameter	Description
serverName	Default name of the server to connect to. Note that this is the name of the DFM domain manager, not the name of the host it is running on. The default is DFM.
initialEventDelay	Time interval (in seconds) the adapter should wait before accepting events from the DFM domain manager. The default value is 1.
Recipients	Comma-separated list of the recipients. The default is null.
SenderId	Address associated with the adapter.
MailServer	Fully qualified domain name for the mail server.
eventSmoothingInterval	Time (in units of seconds) that an event must remain active before the adapter sends a notification. If the notification is cleared before the event smoothing interval expires, the notification is not sent. The smoothing interval also controls when notifications are changed. The default value is 0 seconds.
minimumCertainty	Threshold above which notifications are sent. Any notification with a certainty below the threshold is discarded. Values may range from 0.0 to 1.0. The default value is 0.01.
SubscribesTo	Devices and types of notifications an adapter subscribes to. The default is all devices and notifications. Refer to the "Changing Subscriptions for Notification Adapters" section on page 10-10.
trace	Setting for logging mail messages. The default is FALSE. If set to TRUE, mail messages are saved in the adapter log file, *NMSROOT*/objects/smarts/logs/sm_mail_notifier.log.

When the adapter starts, it creates the Mail Notifier log file *NMSROOT*/objects/smarts/logs/sm_mail_notifier.log. Anytime you change the Mail Notifier Adapter file, you must restart the corresponding CiscoWorks process as described in the "DFM and CiscoWorks Processes" section on page 11-12. To configure the Mail Notifier Adapter for automatic startup with CiscoWorks, refer to the "Using the Command Line to Configure Adapters" section on page 10-8.

Examples

The following examples show how to configure the Mail Notifier Adapter so that DFM alarms can be forwarded to a number of recipients (such as network management systems).

This example sends email to two users when DFM can ping a device, but SNMP requests timeout with no response. (You could also modify the recipients fields to send a page, referencing the appropriate paging mechanism for your environment.)

```
ProvidesAdditionalParams = MailAdapterParams::mail_Notifier-Parameters
    {
#       Recipients = "username1@host.domain,username2@otherhost.otherdomain"
#         SenderId = "daemon@localhost"
#       MailServer = "mailhost.domain"
    }
.
.
.
SubscribesTo =
        {

            GA_ChoiceSubscription::Mail-All-Subscriptions
            {
            # Subscribe to events whose class, instance, and event
            # names match the given pattern.
                   className = "SNMPAgent"
               instanceName = ".*"
                  eventName = "Unresponsive"
                 aggregates = FALSE
                   symptoms = TRUE
            }
        }
```

For more information on editing subscriptions, including specific examples of edited subscriptions for the Mail Notifier Adapter, refer to the "Changing Subscriptions for Notification Adapters" section on page 10-10.

Configuring the HPOV-NetView Adapter

The HPOV-NetView Adapter configuration file server.conf is used to configure the adapter for an OpenView or NetView platform. See the "Event Adapters and their Uses" section on page 10-3 for a description of the HPOV-NetView Adapter.

You can use this adapter with local or remote hosts running:

- HP OpenView 6.2
- NetView 6.01
- NetView 7.1 (supported only on DFM 1.2 Updated for Common Services Version 2.2 when it is installed on CiscoWorks Common Services 2.2)

The configuration file is stored in the *NMSROOT*/objects/smarts/conf/OV (OpenView) or the *NMSROOT*/objects/smarts/conf/NV (NetView) directory. When the adapter starts, it reads the configuration information from the appropriate location. This adapter also creates log files: *NMSROOT*/objects/smarts/logs/sm_ov_fwd.log (HP OpenView) and *NMSROOT*/objects/smarts/logs/sm_nv_fwd.log (NetView).

The server.conf file should not require editing. If it is necessary to change the name of the DFM server, edit the following line in the file:

```
remoteServerName = "DFM"  # This field must be specified.
```

Note Do not change any other values in this file.

The HPOV-NetView Adapter is started and stopped automatically by the OpenView or NetView network management system. If you change the configuration file using the command line, you must stop and restart the adapter using the appropriate HP OpenView and NetView commands.

Configuring the SNMP Trap Adapter

Note The SNMP Trap Adapter must be configured before it can be used. See the "Configuring and Starting Adapters" section on page 10-5 for detailed procedures.

You can use either the GUI or command line to configure the SNMP Trap Adapter. The GUI lets you do essential configuration, while the command line lets you fine-tune the adapter.

Using the DFM GUI to Configure the SNMP Trap Adapter

You can use the DFM GUI to configure the SNMP Trap Adapter in these ways:

- Using the DFM GUI to Configure the SNMP Trap Adapter to Receive Traps, page 10-34
- Using the DFM GUI to Configure the SNMP Trap Adapter to Forward Traps, page 10-36

Using the DFM GUI to Configure the SNMP Trap Adapter to Receive Traps

When you enable the SNMP Trap Adapter using the DFM GUI, the adapter process is registered with CiscoWorks, and you can check its status using **Server Configuration > Administration > Process Management**. If you stop the adapter process from the CiscoWorks GUI, the adapter will be marked disabled in the DFM GUI.

To configure which port will be monitored for SNMP traps DFM receives from other NMSs:

Step 1 Make sure the devices and/or NMS that are forwarding traps to DFM are configured to send them to the port defined in the adapter configuration file.

If another NMS is already listening for traps on the standard UDP trap port (162), you must configure this adapter to use another port, such as port 9000.

Step 2 Select **Device Fault Manager > Administration > Trap Configuration > Trap Receiving**.

Step 3 In the Listening Port field, enter the port number of the local host where the DFM server is running.

Step 4 Select the Restart DFM Server check box if you want to restart the server when you click **OK**. (You must restart the DFM server for your changes to take effect.)

Step 5 Click **OK**.

This procedure does the following:

- Enables/disables the process and registers/unregisters the process with CiscoWorks. When the process is unregistered, you will no longer see it using **Server Configuration > Administration > Process Management**.
- Updates the SNMP Trap Adapter configuration file *NMSROOT*/objects/smarts/conf/trapd/trapd.conf.
- Saves SNMP Trap Adapter log information in the DFM domain manager log file, *NMSROOT*/objects/smarts/logs/DFM.log.

If you did not select the Restart DFM Server check box, you must do so using **Server Configuration > Administration > Process Management**. To further configure the adapter—for example, to specify how non-printable characters will be formatted—use the command line (refer to the "Using the Command Line to Configure the SNMP Trap Adapter" section on page 10-38).

You may also need to complete one or more of the following steps:

- If your network devices are already sending traps to another management application, configure that application to forward traps to DFM.
- If your devices are already configured to send traps to port 162, and no other management application is using port 162, then you can configure DFM to listen on port 162 instead of port 9000.
- If your network devices are already sending traps to another NMS, but that NMS cannot forward traps, you should:
  - Configure the devices to send traps to DFM.
  - Configure DFM to forward the traps to the other NMS.

Using the DFM GUI to Configure the SNMP Trap Adapter to Forward Traps

When you enable the SNMP Trap Adapter using the DFM GUI, the adapter process is registered with CiscoWorks, and you can check its status using **Server Configuration > Administration > Process Management**. If you stop the adapter process from the CiscoWorks GUI, the adapter will be marked disabled in the DFM GUI.

To have DFM send received traps to other NMSs:

Step 1 Select **Device Fault Manager > Administration > Trap Configuration > Trap Forwarding**.

Step 2 In the Forwarding field, select **ON** or **OFF** to enable or disable trap forwarding.

Step 3 If you want to add a recipient, in the Add Recipient field, enter the hostname and port number of the machine you want to forward traps to.

Step 4 If you want to remove a recipient, select the recipient from the Remove Recipient field.

Step 5 Click **OK**.

Step 6 If you want to add or remove any other recipients, repeat the appropriate steps and click **OK**. Repeat these steps until you have added or removed all recipients.

Step 7 Select the Restart DFM Server check box if you want to restart the server when you click **OK**. (You must restart the DFM server for your changes to take effect.)

This procedure does the following:

- Enables/disables the process and registers/unregisters the process with CiscoWorks. When the process is unregistered, you will no longer see it using **Server Configuration > Administration > Process Management**.
- Updates the SNMP Trap Adapter configuration file *NMSROOT*/objects/smarts/conf/trapd/trapd.conf.
- Saves SNMP Trap Adapter log information in the DFM domain manager log file, *NMSROOT*/objects/smarts/logs/DFM.log.

If you did not select the Restart DFM Server check box, you must do so using **Server Configuration > Administration > Process Management**. To further configure the adapter—for example, to specify how non-printable characters will be formatted, or to specify the community string for forwarding destinations—use the command line (refer to the "Using the Command Line to Configure the SNMP Trap Adapter" section on page 10-38).

Using the Command Line to Configure the SNMP Trap Adapter

This topic describes the contents of the SNMP Trap Adapter file (*NMSROOT*/objects/smarts/conf/trapd/trapd.conf). Table 10-5 lists the parameters you can change using the command line.

If another NMS is already listening for traps on the standard UDP trap port (162), you must configure this adapter to use another port, such as port 9000.

You may also want to do the following:

- If your network devices are already sending traps to another management application, configure that application to forward traps to DFM.
- If your devices are already configured to send traps to port 162, and no other management application is using port 162, then you can configure DFM to listen on port 162 instead of port 9000.

```
#
# trapd.conf - Configuration file for SNMP Trap Adapter
#
# Copyright (C) 1997-2004, System Management ARTS (SMARTS)
# All Rights Reserved
#
# RCS $Id: trapd.conf,v 1.1.2.6.2.1 2004/02/20 19:42:44 guw Exp $
#

#
# Lines starting with "#" are comments.
#
# Following is a list of parameters with a brief description.
# More information is available at the end of this file.
#
#
# PORT      UDP port number the trap adapter listens to.
#
# WINDOW    De-duplication window, in seconds.
#
# ASCII     Controls formatting of non-printable characters.
#
# TAG       Enables tagging of varbind values.
#
# ENABLE_FWD   Enables trap forwarding.
#
# MATCH     Determines whether traps are tested against all forwarding
#           criteria or up to the first criterion that matches.
#
```

```
# FORWARD   Specifies matching criteria for traps and the forwarding
#           destinations for matched traps.


##########
#
# Set the parameters here.

PORT: 9000

#WINDOW: 10

#ASCII: FALSE

#TAG: FALSE

ENABLE_FWD: TRUE

#MATCH: all

#FORWARD: *.*.*.* .* * * host:port


##########
#
# Detailed parameter explanations
#
# PORT      UDP port number the trap adapter listens to.
#           Valid values: <0-65535>
#           If NetView or OpenView is running on this system, during
#           installation, the value is set to 9000, otherwise it is set
#           to 162.
#
# WINDOW    De-duplication window, in seconds. The maximum amount of time
#           between receiving similar traps before the second trap is
#           considered unique.
#           To disable de-duplication feature (make all traps unique), don't
#           specify a value for this parameter or use zero.
#           Valid values: non-negative integer
#           If not set (or zero is used), de-duplication is disabled.
#
# ASCII     Controls formatting of non-printable characters.
#           When TRUE, non-printable characters are streamed as '.'.
#           When FALSE, non-printable characters are streamed as
#           their hex value.
#           Valid values: < TRUE | FALSE >
#           If not set, value is FALSE.
#
```

```
# TAG       Enables tagging of varbind values. When enabled, it streams
#           value's type before each value, e.g. INTEGER-32 3.
#           Valid values: < TRUE | FALSE >
#           If not set, value is FALSE.
#
# ENABLE_FWD   Enables trap forwarding. Traps are only forwarded
#              if forwarding criteria is specified.
#              See the FORWARD parameter for more information.
#              Valid values < TRUE | FALSE >
#              If not set, value is TRUE.
#
# MATCH     Determines whether traps are tested against all forwarding
#           criteria or up to the first criterion that matches. If forwarding
#           criteria is not specified, this parameter is ignored.
#           Forwarding criteria is specified in this file using the
#           FORWARD parameter.
#           Valid values: < all | first >
#           If not set, value is first.
#
# FORWARD   Specifies matching criteria for traps and the forwarding
#           destinations for matched traps. The default community "public" will
#           be used if community is not specified (DFM Patch/IDU 1.2.8 or
#           later) or is empty string.
#           Valid syntax: <address> <OID> <generic type> <specific type> \
#                         <host <[:port] | [:port:community]>> [<host<[:port] |
[:port:community]>> ...]
#
# examples  All specific traps received from network devices in the range
#           from 192.168.230.32 to 192.168.230.120 will be sent to asterix
#           port 5000 and jupiter port 162.
#
#     FORWARD: 192.168.250.<32-120> .* 6 * asterix:5000 jupiter
#
#           All generic traps (0 to 5) from network devices whose address
#           matches 193.20.*.* will be sent to obelix port 7000, surya port 162
#           and host 194.56.78.23 port 9000.
#
#     FORWARD: 193.20.*.* .* <0-5> * obelix:7000 surya 194.56.78.23:9000
#
#           All traps received from all network devices will be sent to
#           host snake port 9099.
#
#     FORWARD: *.*.*.* .* * * snake:9099
#
#           All traps received from all network devices will be sent to
#           host snake port 9099 using community string public1.
#
```

```
#     FORWARD: *.*.*.* .* * * snake:9099:public1
#
# fields      Globbing can be used to specify values of the following fields:
#             <address>, <OID>, <generic type>, <specific type>.
#
#           <address>  Source IP address or host name of the SNMP agent
#                      sending the trap. Globbing can be used to specify
#                      ranges of IP addresses.
#                      Examples of valid values:
#                        192.168.114.5
#                        *.*.*.*
#                        192.168.*.*
#                        *.*.*.168
#                        192.168.<120-123>.*
#                        cobra
#                        obelix.planet.net
#
#           <OID>      Enterprise OID of incoming trap. Globbing can be used
#                      to specify ranges of OIDs.
#                      Examples of valid values:
#                        .1.3.6.4.1.9
#                        .1.3.*
#                        .*
#                        .1.3.6.4.1.<1-18>.*
#
#           <generic type>
#                      Generic type of incoming trap. Globbing can be used
#                      to specify ranges of trap numbers.
#                      Examples of valid values:
#                        0
#                        3
#                        <0-5>
#                        *
#
#           <specific type>
#                     Specific type of incoming trap, if valid. For
#                     standard (generic) traps this field is ignored.
#                     Globbing can be used to specify ranges of trap numbers.
#                     Examples of valid values:
#                       733
#                       <130-156>
#                       *
#
```

```
#            <host <[:port] | [:port:community]>>
#                      Destination to forward matched traps to. Multiple
#                      destinations can be specified. Host is specified as an
#                      IP address or a host name. Port and Community are optional;
#                      if omitted, port 162 is used and "public" community string
#                      is used.
#                      Globbing can not be used in this field.
#                      Examples of valid values:
#                        cobra
#                        snake.planet.net:6789
#                        192.76.70.21
#                        192.168.70.190:6789
#                        192.178.70.151:1554:private
#
```

SNMP Trap Adapter Command Line Parameters

Table 10-5 lists the SNMP Trap Adapter parameters you can change.

Table 10-5 SNMP Trap Adapter Parameters

Parameter	Description
PORT	Specifies the UDP port number the adapter listens to. If NetView or OpenView is running on this system, during installation, the value is set to 9000; otherwise it is set to 162. The value is overridden if the port number is specified (-p) when the adapter is started. Valid values are 0-65535.
WINDOW	Specifies the maximum amount of time between similar traps before the second trap is considered unique. To make all traps unique, do not specify a value for this parameter or use zero. Valid values are any non-negative integer.
ASCII	Controls how non-printable characters are formatted. When the value is TRUE, non-printable characters are represented with a period (.). When the value is FALSE, non-printable characters are represented by their hex values. If the value is not set, it is considered FALSE.
TAG	Enables tagging of varbind values. When the value is TRUE, the value's type appears before each value. For example, INTEGER-32 3. If the value is not set, it is considered FALSE.
ENABLE_FWD	Enables trap forwarding. Even if this value is TRUE, traps are not forwarded unless forwarding criteria are specified (see FORWARD). If the value is not set, it is considered TRUE.

Table 10-5 SNMP Trap Adapter Parameters (continued)

Parameter	Description
MATCH	Determines whether traps are tested against all forwarding criteria or up to the first criterion that matches. If forwarding criteria are not specified, this parameter is ignored. Valid values are all or first. If the value is not set, it is considered *first*.
FORWARD	Specifies matching criteria for traps and the forwarding destinations for matched traps. **FORWARD:***address*.*OID_number trap_number specific_trap host*:*port* If you have installed DFM 1.2 Patch/IDU 1.2.8 or later, you can also specify the community string for the forwarding destination, using this syntax: **FORWARD:***address*.*OID_number trap_number specific_trap* [[*host*:*port* | *host*:*port*:*community*] ...] To download the latest DFM 1.2 Patch/IDU, log in to Cisco.com and go to http://www.cisco.com/cgi-bin/tablebuild.pl/cw2000-dfm.

When the adapter starts, it saves SNMP Trap Adapter log information in the domain manager log file, *NMSROOT*/objects/smarts/logs/DFM.log. Anytime you change the SNMP Trap Adapter file, you must restart the corresponding CiscoWorks process as described in the "Using the Command Line to Configure Adapters" section on page 10-8.