This documentation and any related computer software help programs (hereinafter referred to as the "Documentation") is for the end user's informational purposes only and is subject to change or withdrawal by CA at any time.

This Documentation may not be copied, transferred, reproduced, disclosed, modified or duplicated, in whole or in part, without the prior written consent of CA. This Documentation is confidential and proprietary information of CA and protected by the copyright laws of the United States and international treaties.

Notwithstanding the foregoing, licensed users may print a reasonable number of copies of the Documentation for their own internal use, and may make one copy of the related software as reasonably required for back-up and disaster recovery purposes, provided that all CA copyright notices and legends are affixed to each reproduced copy. Only authorized employees, consultants, or agents of the user who are bound by the provisions of the license for the Product are permitted to have access to such copies.

The right to print copies of the Documentation and to make a copy of the related software is limited to the period during which the applicable license for the Product remains in full force and effect. Should the license terminate for any reason, it shall be the user's responsibility to certify in writing to CA that all copies and partial copies of the Documentation have been returned to CA or destroyed.

EXCEPT AS OTHERWISE STATED IN THE APPLICABLE LICENSE AGREEMENT, TO THE EXTENT PERMITTED BY APPLICABLE LAW, CA PROVIDES THIS DOCUMENTATION "AS IS" WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IN NO EVENT WILL CA BE LIABLE TO THE END USER OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF CA IS EXPRESSLY ADVISED OF SUCH LOSS OR DAMAGE.

The use of any product referenced in the Documentation is governed by the end user's applicable license agreement.

The manufacturer of this Documentation is CA.

Provided with "Restricted Rights." Use, duplication or disclosure by the United States Government is subject to the restrictions set forth in FAR Sections 12.212, 52.227-14, and 52.227-19(c)(1) - (2) and DFARS Section 252.227-7014(b)(3), as applicable, or their successors.

All trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.

Copyright © 2008 CA. All rights reserved.
CA Product References

This document references SPECTRUM®.

Contact CA

Contact Technical Support

For your convenience, CA provides one site where you can access the information you need for your Home Office, Small Business, and Enterprise CA products. At http://ca.com/support, you can access the following:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Provide Feedback

If you have comments or questions about CA product documentation, you can send a message to techpubs@ca.com.

If you would like to provide feedback about CA product documentation, complete our short customer survey, which is also available on the CA support website, found at http://ca.com/support.
Preface

The SPECTRUM Alarm Notification Manager User Guide (1308) provides the information you need to configure and use the SPECTRUM Alarm Notification Manager (SANM).

This guide contains the following information:

- **Introduction** (see page 9)– This chapter introduces the features and functionality of SANM.
- **Creating and Editing Alarm Notification Policies** (see page 13)– This chapter provides information for creating and editing alarm notification policies.
- **Associating Policies with Applications** (see page 21)– This chapter provides information for associating policies with applications.
- **Monitoring SANM Processes** (see page 27)– This chapter provides information for monitoring SANM processes.
- **SANM and AlarmNotifier** (see page 31)– This chapter describes the AlarmNotifier feature.
- **Using SANM in a Distributed SpectroSERVER Environment** (see page 41)– This chapter provides information for using SANM in a distributed SpectroSERVER environment.

**Text Conventions**

The following text conventions are used in this document:

<table>
<thead>
<tr>
<th>Element</th>
<th>Convention Used</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variables</td>
<td><em>Italic</em> in angle brackets (&lt;&gt;)</td>
<td>Type the following: DISPLAY=&lt;workstation name&gt;:0.0 export display</td>
</tr>
<tr>
<td>The directory where you installed SPECTRUM</td>
<td>&lt;$SPECROOT&gt;</td>
<td>Navigate to &lt;$SPECROOT&gt;/app-defaults</td>
</tr>
<tr>
<td>(The user supplies a value for the variable.)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(The user supplies a value for the variable.)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Element</td>
<td>Convention Used</td>
<td>Example</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Linux, Solaris, and Windows directory paths</td>
<td>Unless otherwise noted, directory paths are common to both operating systems, with the exception that slashes (/) should be used in Linux and Solaris paths, and backslashes () should be used in Windows paths.</td>
<td>&lt;$SPECROOT&gt;/app-defaults on Linux and Solaris is equivalent to &lt;$SPECROOT&gt;/app-defaults on Windows.</td>
</tr>
<tr>
<td>On-screen text</td>
<td>Courier</td>
<td>The following line displays:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>path=&quot;/audit&quot;</td>
</tr>
<tr>
<td>User-typed text</td>
<td>Courier</td>
<td>Type the following path name:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>C:\ABC\lib\db</td>
</tr>
<tr>
<td>References to documents (title and number)</td>
<td>Italic</td>
<td>Installation Guide (5136)</td>
</tr>
</tbody>
</table>

**Documentation Location and Feedback**

Check the following site for the latest updates and additions to SPECTRUM documents:

http://ca.com/support

To send feedback regarding SPECTRUM documentation, access the following web address:

http://supportconnectw.ca.com/public/ca_common_docs/docserver_email.asp

Thank you for helping us improve our documentation.
# Contents

## Chapter 1: Introduction

- About SPECTRUM Alarm Notification Manager ................................................... 9
- How SPECTRUM Monitors Alarms ........................................................................... 9
  - AlarmNotifier ........................................................................................................ 10
  - Attention! ................................................................................................................ 11
- The Alarm Resource File .......................................................................................... 11

## Chapter 2: Creating and Editing Alarm Notification Policies

- Create an Alarm Notification Policy ................................................................. 13
  - Define a New Filter For a New Policy ................................................................. 14
- Add a Filter to an Existing Policy ................................................................. 16
- Change the Filter Order ......................................................................................... 17
- Edit a Filter ............................................................................................................. 17
  - Introduce Additional Filter Parameters ............................................................. 18
- Delete a Filter .......................................................................................................... 18
- Add a Model or Alarm to a Policy ........................................................................... 19
- Editing an Alarm Notification Policy ................................................................. 19

## Chapter 3: Associating Policies with Applications

- The Association Process ......................................................................................... 21
- The SANM Default Policy ....................................................................................... 21
- Associate a Policy with an Application .................................................................... 22
- The Schedule Subview ............................................................................................ 23
- Schedule an Association .......................................................................................... 23
- Additional Utilities .................................................................................................. 24
  - Assignticket Utility ............................................................................................... 24
  - Clearticket Utility .................................................................................................. 25
  - Updatealarm Utility .............................................................................................. 25

## Chapter 4: Monitoring SANM Processes

- SANM Events ......................................................................................................... 27
- SANM Event Codes ................................................................................................. 27
- Tracing Policies ........................................................................................................ 28
  - The Summary Trace File ...................................................................................... 28
  - The Detailed Trace File ......................................................................................... 29
**Chapter 5: SANM and AlarmNotifier**

- AlarmNotifier Enhancements .......................................................... 31
- Start AlarmNotifier ..................................................................... 31
- Alarm Management Parameters .................................................. 33
  - Access Alarm Management Parameters .................................. 33
  - Receive Alarms from the Cache ............................................. 33
- Alarm Acknowledgement ............................................................. 34
- User-Clearable Alarms ................................................................. 34
- SANM-Enabled Script Parameters .............................................. 35
  - Email Notifications ................................................................. 36
  - Third-Party Applications ........................................................ 36
- SANM-Enabled .alarmrc Parameters .......................................... 39

**Chapter 6: Using SANM in a Distributed SpectroSERVER Environment**

- Landscapes and Alarm Monitoring ............................................ 41
- SANM Policy Management Across Multiple Landscapes ............ 41
  - How to Create SANM Policies in a Single Landscape ............. 42
  - How to Create SANM Policies on Multiple Landscapes ........ 42
  - Methods for Determining Monitored Landscapes .................. 43

**Index**

- 45
Chapter 1: Introduction

This section contains the following topics:

- **About SPECTRUM Alarm Notification Manager** (see page 9)
- **How SPECTRUM Monitors Alarms** (see page 9)
- **The Alarm Resource File** (see page 11)

### About SPECTRUM Alarm Notification Manager

SPECTRUM Alarm Notification Manager (SANM) enhances the functionality of SPECTRUM alarm processing applications, including AlarmNotifier and Attention!. These applications respond to SPECTRUM alarms by email notifications, trouble tickets, and so on. SANM lets you create and associate alarm notification policies with applications.

### How SPECTRUM Monitors Alarms

SPECTRUM, alarm-processing applications, and SANM work together in the alarm monitoring process.

The following diagram shows the SPECTRUM alarm monitoring process:
SPECTRUM monitors alarms by doing the following:

1. SPECTRUM polls the modeled network elements and updates the status of each element in the SpectroSERVER database.
2. SPECTRUM generates an alarm based on a trap received from the network or when it detects a critical status change in a network model. On the OneClick Console, the model icon's condition panel changes from green to another color that indicates the relative severity level of the alarm. SPECTRUM posts specific information for each alarm on the Alarm Details tab of the Component Detail panel and the alarm event information on the Events tab of the Component Detail panel.
3. Data about alarms that SPECTRUM has generated is passed to SANM. SANM lets you create and associate alarm notification policies with alarm processing applications. In addition, SANM's Schedule subview lets you schedule application and policy associations and automates the association process.
4. SANM passes the alarm information to alarm processing applications only when the alarm types specified in the policies occur.

**AlarmNotifier**

Alarms filtered by SANM are sent to SPECTRUM AlarmNotifier. When both SANM and SPECTRUM AlarmNotifier are installed, AlarmNotifier gains some additional capabilities:

- You can apply SANM's alarm-filtering policies to individual instances of AlarmNotifier.
- AlarmNotifier can generate alarm notifications from all landscapes of a distributed SpectroSERVER environment.
- Additional commands are available to acknowledge and clear alarms from AlarmNotifier.
- An additional startup command lets you start multiple instances of AlarmNotifier which can be associated with a different SANM alarm notification policy.
- Additional startup commands let you create summary or detailed trace files.
- AlarmNotifier scripts have additional parameters that contain information about troubleshooting alarms.
- The AlarmNotifier resource file has additional parameters to obtain further information about alarms.
- You can use SANM to automatically associate a different policy with AlarmNotifier at a specified time.
Attention!

Attention! is a client-server network monitoring and notification system. Attention! alerts system managers to critical system and network events through alphanumeric pagers, telephone, email, PA announcements, electronic message boards, and custom notifications. You can use SANM as a foundation for integration between SPECTRUM and Attention!.

The Alarm Resource File

The alarm resource file, .alarmrc, contains operating parameters that define the various defaults that SANM uses. You can modify these parameters at any time.

**Note:** We recommend that you take a backup copy of the file before modifying it.

If you re-install SPECTRUM or upgrade to another version, the installation process saves your resource file, .alarmrc, to a backup directory. Versions of the resource file you saved with another name are retained in the `<$SPECROOT>/SANM` directory along with the default resource file provided by the re-installation or upgrade.
Create an Alarm Notification Policy

An alarm notification policy specifies the alarm types that an alarm-processing application should receive and filters the unwanted alarms. You can create alarm notification policies to determine which applications receive what alarms.

To create an alarm notification policy

1. Open OneClick Console and select the Locater tab in the Navigation panel.
2. Select All Applications under SANM and click .
   The Select landscapes to search dialog displays.
3. Select the Spectrum server you want to search in the Exists in/Create in dialog and click OK.
   The available applications and the policy they are using appear in the Contents panel on the right. The policy details appear in the Component Detail panel below the Contents panel.
   Note: Run AlarmNotifier at least once, otherwise, the search returns no models. The AlarmNotifier file is located in the <$SPECROOT>/Notifier directory.
4. In the Component Detail panel, click the link create/set policy under General Information.
   The Select Policy dialog box displays.
5. Click Create.
   The Create SANM Policy dialog displays.
6. Enter the policy name in the Name text box, and optionally, create one or more filters if you want to associate filters with the new policy.

   **Note:** Policy names should preferably indicate when the policy is used, for example, ciscoRtrPM, so that it is easy to pick policies from a collection.

7. Click Ok.

   The new policy is created.

### Define a New Filter For a New Policy

You can define new filters. A filter must be associated with a policy.

**To define a new filter for a new policy**

1. Open OneClick Console and select the Locater tab in the Navigation panel.

2. Select All Applications under SANM and click ![](image).

   The available applications and the policy they are using appear in the Contents and Component Detail panels on the right.

3. In the Component Detail panel, click the create/set policy link in the General Information subview.

   The Select Policy dialog displays.

4. Click Create.

   The Create SANM Policy dialog displays.

5. Click the Add button.

   The Add Filter dialog displays.

6. Enter the following information:

   **Name**

   Defines the new filter name.

   **Notes**

   (Optional) Describes the filter.

   **Age Time**

   (Optional) Indicates the time for which the filter holds the alarm. The alarm passes to the alarm processing application after the age time.

   **Notification Data**

   (Optional) Defines the data that is sent with the alarm notification.
7. (Optional) Select the Landscapes tab to define the landscapes for the filter. To define the landscape, select servers in the Include and Exclude lists. You can move servers between the Include and Exclude lists by using the arrow buttons provided.

   **Note:** SPECTRUM OneClick combines the Landscapes and Servers parameters of the legacy SANM UI into a single parameter, Landscapes.

8. (Optional) Select the Severity tab to define the filter severity. To define the severity, select the alarm severity types to be filtered or allowed from the Include and Exclude lists.

9. (Optional) Select the Device Type tab to specify the device types for the filter, as follows:
   - Select Include device type(s) or Exclude device type(s) from the drop-down list to see the respective lists.
   - Enter a device type and click Add to add it to the included or excluded list.
     **Note:** You must either enter the name of an existing device type or enter the name of a device type you plan to create.
   - Click Browse to select from a list of existing device types.
   - Select a device type and click Remove to remove it from the list.
   - Select a device type and click Modify to edit that device type.

10. (Optional) Select the Collections tab to specify the collection of policies for the filter. Alarms on devices that are in these collections are filtered. The steps to include, exclude, add, remove, modify, and browse for containers are the same as for the previous tab.

11. (Optional) Select the Topology tab to specify the topology containers for the filter. Alarms on devices that are in these topologies are filtered. The steps to include, exclude, add, remove, modify, and browse for containers are the same as for the previous tabs.

12. (Optional) Select the Alarm Type tab to include or exclude the alarm types to be filtered.

13. (Optional) Select the Model Type tab to include or exclude the model types to which the filter applies.

14. (Optional) Select the Location tab to specify the location containers for the filter. Alarms on devices that are in these locations are filtered. The steps to include, exclude, add, remove, modify, and browse for containers are the same as for the previous tabs.

15. (Optional) Select the Organization tab to specify the organization containers for the filter. Alarms on devices that are in these organizations are filtered. The steps to include, exclude, add, remove, modify, and browse for containers are the same as for the previous tabs.
16. (Optional) Select the IP Address/Range tab to specify the Internet Protocol (IP) addresses for the filter. The alarm is generated within the specified network, subnet, or IP address range for SANM to pass the alarm to the alarm processing application. The steps to include, exclude, add, remove, and modify for IP addresses are the same as for the previous tabs.

17. (Optional) Select the Model Name tab to specify the model names for the filter. The steps to include, exclude, add, remove, and modify for model names are the same as for the previous tabs.

18. Click OK.

The new filter is defined.

19. Enter a name for the new policy in the Name field of the Create SANM Policy dialog and click OK.

   **Note:** Policy names should preferably indicate when the policy is used, for example, ciscoRtrPM, so that it is easy to pick policies from a collection.

   The new policy is created.

---

**Add a Filter to an Existing Policy**

You can add a filter to an existing policy.

**To add a filter to a policy**

1. Click the Locator tab on the Navigation panel of the OneClick Console.

2. Select SANM, All Policies and click .

   The existing polices display in the Contents panel on the right.

3. Select the policy for which you want to add a new filter.

   The policy details display in the Component Detail panel.

4. Expand the Filters menu under the Information tab in the Component Detail panel and click .

   The Add a Filter dialog displays.

5. Enter the filter information as explained in **Define a New Filter** (see page 14) and save the information.

   The filter is added to the policy.
Change the Filter Order

You can change the order in which the filters of a policy are processed.

This applies only to Notification Data. For instance if Notification Data on filter 1 has jack@xyz.com, and filter 2 has jill@xyz.com, then the alarm notifier will output jack@xyz.com:jill@xyz.com. If you change the order it will be jill@xyz.com:jack@xyz.com.

To change the filter order
1. Open OneClick in a browser and select the Locater tab in the Navigation panel.
2. Select SANM, All Policies and click .
   The existing policies display in the Contents panel on the right.
3. Select a policy for which you want to change the filter order.
   The policy details appear in the Component Detail panel below the Contents panel.
4. Expand the Filters menu under the Information tab and click .
   The Set Order dialog displays.
5. Select a filter and use the arrow buttons to push it up or down the order, and click OK.
   The filter order is changed.

Edit a Filter

Editing a filter is the same as editing its parameters. You can add, edit, and delete the filter parameters.

To edit a filter
1. Select the filter in the filter table and click .
   The filter opens for editing.
2. Edit the Name, Notes, Age Time, Notification Data fields as required.
3. Click each parameter tab to add, edit and delete the corresponding parameter values, as explained in Define a New Filter (see page 14).
   Note: If you delete all values of a parameter, the filter no longer filters by that parameter.
Delete a Filter

4. Select the check box Show only filtered by parameters if you want to view only the parameters included in the filter.

5. Click OK.
   The filter is edited.

Introduce Additional Filter Parameters

You can introduce additional parameters into a filter to increase the level of filtering.

**Note:** You can add a parameter to a filter by adding a value to that parameter. That is, if a parameter was not included when creating the filter, defining a value for that parameter adds that parameter to the filter.

**To add additional parameters to a filter**

1. Open the filter for editing as explained previously.
2. Click the tab of the parameter that you want to add to the filter.
3. Add one or more values to the parameter as explained in Define a New Filter (see page 14) and click OK.
   The parameter is added to the filter.

Delete a Filter

You can delete a filter that is no longer required.

To delete a filter, select the filter in the filter table and click ✗.

The filter is deleted.
Add a Model or Alarm to a Policy

Use the following procedure to add a model or alarm to a policy.

**To add a model or alarm to a policy**
1. Open OneClick in a browser and select a model or alarm.
   The model or alarm details display in the Contents panel on the right.
2. Right-click on the model or alarm and select Add to, SANM Policy, Add...
   The Select Policy dialog displays.
3. Select a policy and click OK.
   The Select Write Option dialog displays.
   **Note:** To remove the item, select Remove...
4. Select an option.
   The model or alarm is added to the selected policy.

Editing an Alarm Notification Policy

You can edit a policy before or after you save it, regardless of whether or not it is associated with an application. If the policy is associated with an application, SANM begins enforcing the new policy as soon as you save your changes.

**Important!** The Archive Manager must be running and connected to the SpectroSERVER for modified policies to take immediate effect.
Chapter 3: Associating Policies with Applications

This section contains the following topics:

- The Association Process (see page 21)
- The SANM Default Policy (see page 21)
- Associate a Policy with an Application (see page 22)
- The Schedule Subview (see page 23)
- Schedule an Association (see page 23)
- Additional Utilities (see page 24)

The Association Process

After you create an alarm notification policy, you associate the policy with one or more alarm processing applications. An association between a policy and an application remains in effect until you associate another policy with that application or delete the associated policy.

An application can have only one policy associated at a time because SANM enforces only one policy at a time on the application. If you want an application to process different alarms at different times, you can associate the policies with the applications manually at run time or use the Schedule subview to schedule the associations automatically at a specified date and time. To run the same application with different policies, you have to start multiple instances of the application, each with a unique name, and then associate the different policies with them.

To change the policy associated with an application, associate the default policy or another policy with the application. If you delete the policy associated with an application, SANM associates the default policy with the application. If you edit a policy that is associated with multiple applications, you change the policy for all the applications. You do not have to reassociate the policy with each application.

The SANM Default Policy

SANM associates a default policy with each application whenever you start the application for the first time, or when you delete a policy associated with that application. You can also explicitly associate the default policy with an application.
The default policy is a null policy; it does not filter any alarm. That is, applications associated with the default policy receive all alarm notifications that occur in every landscape in the landscape map of the SpectroSERVER to which SANM is connected.

You can modify the default policy if you do not want it to be null, but SANM continues to associate it with applications by default.

If you delete a policy associated with an application, SANM associates the default policy with it. Therefore, before you delete a policy, check if the default policy is modified. If you delete a policy associated with an application or modify the default policy, SANM warns you about the association.

If you do not want the default policy associated with an application when you delete the policy associated with it, associate a different policy with the application so that the current policy is automatically deleted.

## Associate a Policy with an Application

You can associate a policy with an application.

**To associate a policy with an application**

1. Open the OneClick Console and click the Locator tab on the Navigation panel.

2. Select SANM, All Applications and click ![locator button].
   The existing applications display in the Contents panel on the right.

3. Click the Create/Set Policy link.
   The Select Policy dialog displays.

4. Select a policy and click OK.
   The policy is associated with the application.
The Schedule Subview

The Schedule subview automates the association process and lets you implement the alarm notification policies you want. For example, if you want an application to take action in response to an alarm during the evening, you can create a special evening policy and schedule the association of this policy with the application for 6 PM every day. You can also schedule the association of a different daytime policy with the same application for 7 AM every day. The Schedule subview lets you perform associations at the specified times. Without the Schedule subview, you need to manually associate a new policy each time you want to filter alarms in a different way.

You can view the results of operations performed by the Schedule subview on the Events tab in OneClick.

**Note:** Windows schedules or cron jobs created prior to SPECTRUM r9.0 for SANM must be recreated in the OneClick interface.

Schedule an Association

Use the following procedure to schedule a policy association with an application.

**To schedule an association**

1. Open OneClick Console and select the Locater tab in the Navigation panel.
2. Select SANM, All Applications and click ➔.
   The existing applications display in the Contents panel on the right.
3. Select the SANM application for which you want to schedule a policy.
4. In the Component Detail panel, expand the Scheduled Policies menu under the Information tab and click ➔.
   The Select Policy And Schedule dialog displays.
5. Select a policy, select a schedule, and click OK.

   **Note:** You can create policies and schedules by clicking the respective Create button.
   The scheduled policy displays in the Scheduled Policies table.
**Additional Utilities**

AlarmNotifier includes three additional utilities that can be used to manipulate existing alarms:

- `assignticket`
- `clearticket`
- `updatealarm`

**Assignticket Utility**

The `assignticket` utility is used to populate an alarm's Trouble Ticket ID field with the name of the person the trouble ticket is assigned to.

This utility has the following format:

`assignticket <model handle> <alarm id> <assignee> [username]`

**model handle**

Indicates the handle of the SPECTRUM model on which the alarm exists.

**alarm id**

Indicates the ID of the alarm to write to.

**assignee**

Indicates the name of the user the ticket is assigned to.

**username**

(Optional) Specifies the name of the SPECTRUM user to connect to the SpectroSERVER as.
**Clearticket Utility**

The clearticket utility is used to clear an alarm.

This utility has the following format:

```
clearticket -mh <model_handle> -ai <alarm_ID> -su <username>
```

- **model handle**
  Indicates the handle of the SPECTRUM model on which the alarm exists.

- **alarm id**
  Indicates the ID of the alarm to clear.

- **username**
  Specifies the name of the SPECTRUM user to connect to the SpectroSERVER as.

**Updatealarm Utility**

The updatealarm utility is used to set the value of any attribute on any alarm.

This utility has the following format:

```
updatealarm <model handle> <alarm id> <attr id> <attr value> [username]
```

- **model handle**
  Indicates the handle of the SPECTRUM model on which the alarm exists.

- **alarm id**
  Indicates the ID of the alarm to write to.

- **attr id**
  Indicates the ID of the attribute to write to.

- **attr value**
  Indicates the value to write to the attribute.

- **username**
  (Optional) Specifies the name of the SPECTRUM user to connect to the SpectroSERVER as.
Chapter 4: Monitoring SANM Processes

This section contains the following topics:

*SANM Events* (see page 27)
*Tracing Policies* (see page 28)

**SANM Events**

The Events tab in OneClick lists events that occur on a SpectroSERVER. When a user performs a SANM operation, the results of the operation appear on the Events tab along with other SPECTRUM events. The following information about an event is listed:

- Date and time of the operation
- Application name and policy name
- User's host and user's name
- Explanation of the event
- Event code

*Note:* For information on the Events tab, see the *Event Configuration User Guide (5188).*

**SANM Event Codes**

Each SANM event code corresponds to a SANM operation. Use the following SANM event codes to locate SANM operation entries or filter out all but specific SANM operation entries.

<table>
<thead>
<tr>
<th>Event Code</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>00d70000</td>
<td>Application registered with SANM</td>
</tr>
<tr>
<td>00d70001</td>
<td>Application unregistered with SANM</td>
</tr>
<tr>
<td>00d70002</td>
<td>Association created</td>
</tr>
<tr>
<td>00d70004</td>
<td>Scheduled association created</td>
</tr>
<tr>
<td>00d70006</td>
<td>Policy created</td>
</tr>
<tr>
<td>00d70008</td>
<td>Policy modified</td>
</tr>
<tr>
<td>00d7000a</td>
<td>Application created</td>
</tr>
</tbody>
</table>
To get information about how a policy is working for a SANM-enabled application, you can specify at application start-up the creation of a detailed or summary trace file for the application. A detailed trace file indicates the filters in a policy alarm that did not match when they were evaluated against that policy. A summary trace indicates the time when an alarm notification is passed to the associated application when that application is started. A summary trace file does not include information about alarms that do not meet the criteria specified in a policy.

A record of policy-based actions by SANM can be used as a decision making tool. The results may confirm that you have the correct policy in effect for an application, or they may compel you to refine your policy, especially if you discover that you are inadvertently excluding alarms that should be passed to an application.

The Summary Trace File

The summary trace file includes a summary of all alarm notifications (set, cleared, updated) sent to the application, as follows:

<table>
<thead>
<tr>
<th>Date/Time</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>05/24/2000 15:48:44</td>
<td>SANM Trace Entry 1</td>
</tr>
<tr>
<td></td>
<td>Notification sent to AlarmNotifier for Alarm 52 set on landscape 0x540000</td>
</tr>
<tr>
<td>05/24/2000 15:48:44</td>
<td>SANM Trace Entry 2</td>
</tr>
<tr>
<td></td>
<td>Notification sent to AlarmNotifier for Alarm 21 updated on landscape 0x540000</td>
</tr>
<tr>
<td>05/24/2000 15:48:44</td>
<td>SANM Trace Entry 3</td>
</tr>
<tr>
<td></td>
<td>Notification sent to AlarmNotifier for Alarm 26 cleared on landscape 0x540000</td>
</tr>
</tbody>
</table>
The Detailed Trace File

A detailed trace file includes entries for alarms that meet and do not meet the criteria specified in a policy. An alarm entry includes the alarm’s attribute values that are compared to the filter parameter values. An arrow symbol under MATCH between ALARM VALUES and FILTER VALUES indicates a match. The arrow is absent if the values do not match.

The following is an example of a trace file that indicates that an alarm passed a policy:

```
AlarmNotifier Trace Entry 305

Applying first_shift to Alarm 8982 set on landscape 0x540060
Applying Filter 1, tag: Almar or Abbott

<table>
<thead>
<tr>
<th>ALARM VALUES</th>
<th>MATCH</th>
<th>FILTER VALUES</th>
</tr>
</thead>
<tbody>
<tr>
<td>LANDSCAPE</td>
<td>LANDSCAPE</td>
<td>0x54006000</td>
</tr>
<tr>
<td>Model Type</td>
<td>Pingable</td>
<td>Pingable</td>
</tr>
<tr>
<td>Device Location</td>
<td>World:USA/NorthEast:</td>
<td>USA</td>
</tr>
<tr>
<td>Alarm Severity</td>
<td>CRITICAL</td>
<td>MAINTENANCE</td>
</tr>
<tr>
<td>Alarm Cause</td>
<td>0x10007</td>
<td>0x100065</td>
</tr>
<tr>
<td>Server Host</td>
<td>coffee</td>
<td>coffee</td>
</tr>
</tbody>
</table>

Alarm Passed Policy

Notification sent to AlarmNotifier for Alarm 8982 set on landscape 0x540060
```
The following is an example of a trace file that indicates that an alarm failed a policy:

```
AlarmNotifier Trace Entry 366
Applying force_shift to Alarm 8986 set on landscape 0x45006

ALARM VALUES MATCH FILTER VALUES
------------------ ------ ----------------------
LANDSCAPE 0x45006  --> 0x450060
SINGLE_TYPE Umgable  --> Umgable

DEVICE_LOCATION
World: USA/NorthEast  -->  USA

FILTER 1 FAILED
Notification NOT sent to AlarmNotifier for Alarm 8986 set on
```
Chapter 5: SANM and AlarmNotifier

This section contains the following topics:

- **AlarmNotifier Enhancements** (see page 31)
- **Start AlarmNotifier** (see page 31)
- **Alarm Management Parameters** (see page 33)
- **Alarm Acknowledgement** (see page 34)
- **User-Clearable Alarms** (see page 34)
- **SANM-Enabled Script Parameters** (see page 35)
- **SANM-Enabled .alarmrc Parameters** (see page 39)

### AlarmNotifier Enhancements

AlarmNotifier gains additional capabilities when you install SANM on your system. These capabilities include additional start-up options that allow for specifying application names and for creating trace files, alarm acknowledge and alarm clear commands, and script and resource file parameters. SANM also allows AlarmNotifier to operate in a distributed environment.

### Start AlarmNotifier

AlarmNotifier is located in the `<$SPECROOT>/Notifier` directory. This directory contains the following files by default:

- .alarmrc
- AlarmNotifier
- ClearScript
- README
- SetScript
- UpdateScript

AlarmNotifier includes the following additional files and directory:

**AlarmAck**

Acknowledges an alarm.

**AlarmClear**

Clears an alarm.

**Trace**

Displays trace files.
To start AlarmNotifier, use the following AlarmNotifier command located in the `<$SPECROOT>/Notifier` directory:

```
AlarmNotifier [-r <resource file>] [-n <application>] [-tl summary|details] [-tn <trace file>] [-ts <size>]
```

- **-r**
  Lets you specify a resource file other than the default resource file .alarmrc.

- **-n**
  Lets you override the application name value specified by the APPLICATION parameter in the resource file and specify a different name for an AlarmNotifier application instance. This option lets you start multiple instances of AlarmNotifier, each of which you can associate with a different SANM alarm-filtering policy. If a name is not assigned to the APPLICATION parameter in the resource file, you must use the -n option at start-up to specify an application name.

- **-tl**
  Lets you activate tracing at a specified level, summary or detailed. The default format for an AlarmNotifier trace file is the application name together with the date when the trace file was created.

- **-tn**
  Lets you specify a trace file name other than the default name provided when only the -tl option is used. Use this option with the -tl option.

  When using the trace file option, the output file is written by default to the `<$SPECROOT>/Notifier/trace` directory. To explicitly name an output file and path, use the [-tn filename] option. If [set the File Name variable] is a relative path, trace output is written to a file relative to the current directory. If [set the File Name variable] is an absolute path, trace output is written to the absolute path specified.

- **-ts**
  Lets you specify the number of lines in the trace file. Use this option with the -tl option. The application writes this number of lines to the file and then wraps around to the beginning of the file. Entries are numbered sequentially and there is an END OF TRACE line after the last entry. The default number of lines in a trace file is 10000.
Alarm Management Parameters

The Alarm Management view allows you to control some aspects of alarm management. Two parameters in this view, Generate Alarm Events and Add Events to Alarms, have an impact on how AlarmNotifier processes alarm event updates.

Generate Alarm Events

Enables generation of alarm change events (alarms being generated, updated, or cleared). It is enabled by default.

*Note:* If Generate Alarm Events is disabled, the user will not see Alarm History in the Alarm view.

Add Events to Alarms

Controls whether alarm change events are added to each alarm. If disabled, alarm change events are not displayed in the Events tab of the Component Detail panel for the alarm. This option is disabled by default.

*Note:* For more information on the Alarm Management view, see the *Distributed SpectroSERVER Administrator Guide (2770)*.

Access Alarm Management Parameters

You can access alarm management parameters to control some aspects of alarm management.

**To access alarm management parameters**

1. Open OneClick and in the Navigation panel, select a VNM model in the Universe view.
   
   The corresponding details appear in the Contents panel and Component Detail panel on the right.

2. In the Component Detail panel, select the Information tab and open the Alarm Management menu.
   
   The alarm management parameters are visible.

Receive Alarms from the Cache

If the Archive Manager is not consistently providing the events associated with reported alarms due to high traffic, you can retrieve alarm events from the event cache rather than the Archive Manager. To do this, you must set the Store_Originating_Event attribute to True. The event cache can only provide the AlarmNotifier with a single event that created the alarm.
To set the attribute to receive alarms from the cache
1. Change to edit mode from the Alarm Management view.
2. Add a new attribute field for the Store-Originating_Event attribute.
3. Change the value from FALSE to TRUE, and save.
   The attribute is set.

Alarm Acknowledgement

The AlarmAck command allows you to acknowledge alarms. It can be used at any shell command prompt to acknowledge specific alarms or it can be incorporated into a script. AlarmAck returns a value of 0 if the operation is successful. Otherwise, it returns a non-zero value.

You can use the AlarmAck command to acknowledge an alarm using the following syntax:

```
AlarmAck -a <alarm> -l <landscape>
```

- **-a**
  Defines the alarm ID.

- **-l**
  Defines the landscape handle for the landscape where the alarm occurred.
  (Available only for distributed SpectroSERVER environments)

You can use the AlarmAck command to acknowledge all alarms for a model using the following syntax:

```
AlarmAck -m <model handle>
```

where -m is the model handle for the model with the alarm conditions.

User-Clearable Alarms

The AlarmClear command clears user-clearable alarms. To determine if an alarm is user-clearable, check the value of the UserClearable parameter in alarm notifications. It can be used at any shell command prompt to clear specific alarms or it can be incorporated into a script. AlarmClear returns a value of 0 if the operation is successful. Otherwise, it returns a non-zero value.
You can use the AlarmClear command to clear alarms using the following syntax:

```
AlarmClear -a <alarm> -l <landscape>
```

- **-a**
  Defines the alarm ID number.
- **-l**
  Defines the landscape handle of the landscape where the alarm occurred.

## SANM-Enabled Script Parameters

The SetScript, UpdateScript, and ClearScript scripts have additional parameters when then run on a computer that has SANM installed.

The following table describes the SANM-enabled script parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FlashGreen</td>
<td>True or False. This parameter displays in ClearScript notifications but not in SetScript or UpdateScript notifications. True indicates that the cleared alarm is exhibiting the <em>flash green</em> condition (the flash green option for the model is enabled and the GET_FLASH_GREEN parameter in the .alarmrc resource file is set to True). See the <em>SPECTRUM Menus (2519)</em> guide for information on enabling the flash green option. Note that even though this field is not displayed by SetScript and UpdateScript notifications, the parameter is actually passed to these scripts, and because it is invalid, it has the default value of False in each script.</td>
</tr>
<tr>
<td>Location</td>
<td>The location model that contains the network element for which the alarm is set, updated, or cleared if the element is modeled in the OneClick Location view. The location model that contains the model for the errant network element displays in a hierarchical list of location models separated by colons. For example, an alarm for a model contained in Room 222 on the first floor of the Boston building in the northeast region of the United States would appear as follows: USA:Northeast:BostonBldg:FirstFloor:Room222.</td>
</tr>
<tr>
<td>AlarmAge</td>
<td>The length of time, specified in the filters in a SANM policy, that SANM retains an alarm from an instance of AlarmNotifier that is associated with that policy. If the alarm has to pass multiple filters with different ages, then SANM uses the shortest, non-zero alarm age interval.</td>
</tr>
<tr>
<td>NotificationData</td>
<td>A list of notification data entries (names of persons), specified in the filters in a SANM policy, that SANM passes to an instance of AlarmNotifier that is associated with that policy. AlarmNotifier scripts can be configured to initiate email notifications to those persons specified by notification data entries.</td>
</tr>
<tr>
<td>ProbableCause</td>
<td>The probable cause text associated with the alarm.</td>
</tr>
</tbody>
</table>

---

**Chapter 5: SANM and AlarmNotifier** 35
### SANM-Enabled Script Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EventMessage</td>
<td>The message about the events associated with the alarm. This field is blank if the alarm generated by SPECTRUM does not have associated events or if the event does not include additional information about the alarm.</td>
</tr>
</tbody>
</table>

## Email Notifications

If you use an AlarmNotifier script to send an email notification, you must set the value for the VARFORMAIL parameter in the script. This parameter specifies who the email should be sent to.

If you are using SANM-enabled AlarmNotifier, you can use the NotificationData parameter to set the value for VARFORMAIL. If you use NotificationData as the value for VARFORMAIL, email is sent to the persons specified in the NotificationData parameter in the SANM policy that is associated with the instance of AlarmNotifier that invokes the script. For example, if the Notification Data entry is formatted as "John: Mary or Sue: Lynn, Jeff", email is sent to John, Mary, Lynn, and Jeff, but not to Sue, because AlarmNotifier interprets the colon as an AND operator and does not act upon the OR operator.

Other possible values for the VARFORMAIL parameter are RepairPerson or both. The RepairPerson option is the only option available for AlarmNotifier when it is not running with SANM. Both indicate that the email notification is sent to the designated RepairPerson and to the person specified by the NotificationData parameter.

For more information on configuring an AlarmNotifier script to send an email notification, see the *AlarmNotifier User Guide (1503)*.

## Third-Party Applications

You can customize or replace the SetScript, ClearScript, or UpdateScript for integration with a third-party application. If you create your own script or executable, understand which arguments are passed from SPECTRUM to the receiving script or executable. The script or executable should receive all of the arguments passed to it by SPECTRUM in the correct order.

**Note:** Any SPECTRUM attribute of the model involved in the alarm can be passed to AlarmNotifier and used in a script. See the *AlarmNotifier User Guide (1503)* for information.
The following table shows the argument number, name, and format of each argument passed to each script when the USE_NEW_INTERFACE .alarmrc parameter is set to TRUE:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Name</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Date</td>
<td>mm/dd/yy</td>
</tr>
<tr>
<td>2</td>
<td>Time</td>
<td>hh:mm:ss</td>
</tr>
<tr>
<td>3</td>
<td>Model Type</td>
<td>Text</td>
</tr>
<tr>
<td>4</td>
<td>Model Name</td>
<td>Text</td>
</tr>
<tr>
<td>5</td>
<td>Alarm ID</td>
<td>Integer</td>
</tr>
<tr>
<td>6</td>
<td>Severity</td>
<td>Text</td>
</tr>
<tr>
<td>7</td>
<td>Cause</td>
<td>Text</td>
</tr>
<tr>
<td>8</td>
<td>Repair Screen</td>
<td>Text</td>
</tr>
<tr>
<td>9</td>
<td>Server</td>
<td>Text</td>
</tr>
<tr>
<td>10</td>
<td>Landscape</td>
<td>Hexadecimal</td>
</tr>
<tr>
<td>11</td>
<td>Model Handle</td>
<td>Hexadecimal</td>
</tr>
<tr>
<td>12</td>
<td>Model Type Handle</td>
<td>Hexadecimal</td>
</tr>
<tr>
<td>13</td>
<td>IP Address</td>
<td>xxx.xxx.xxx.xxx</td>
</tr>
<tr>
<td>14</td>
<td>Security String</td>
<td>Text</td>
</tr>
<tr>
<td>15</td>
<td>Alarm State</td>
<td>Text</td>
</tr>
<tr>
<td>16</td>
<td>Acknowledged</td>
<td>Text</td>
</tr>
<tr>
<td>17</td>
<td>Clearable</td>
<td>Text</td>
</tr>
<tr>
<td>18</td>
<td>Flash_Green</td>
<td>Text</td>
</tr>
<tr>
<td>19</td>
<td>Location</td>
<td>Text</td>
</tr>
<tr>
<td>20</td>
<td>Age</td>
<td>Integer</td>
</tr>
<tr>
<td>21</td>
<td>Notifdata</td>
<td>Text</td>
</tr>
</tbody>
</table>

The following table shows the argument number, name, and format of each argument passed to each script when the USE_NEW_INTERFACE .alarmrc parameter is set to FALSE:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Name</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Date</td>
<td>mm/dd/yy</td>
</tr>
<tr>
<td>2</td>
<td>Time</td>
<td>hh:mm:ss</td>
</tr>
</tbody>
</table>
### SANM-Enabled Script Parameters

<table>
<thead>
<tr>
<th>Argument</th>
<th>Name</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Model Type</td>
<td>Text</td>
</tr>
<tr>
<td>4</td>
<td>Model Name</td>
<td>Text</td>
</tr>
<tr>
<td>5</td>
<td>Alarm ID</td>
<td>Integer</td>
</tr>
<tr>
<td>6</td>
<td>Severity</td>
<td>Text</td>
</tr>
<tr>
<td>7</td>
<td>Cause</td>
<td>Text</td>
</tr>
<tr>
<td>8</td>
<td>Repair Screen</td>
<td>Text</td>
</tr>
<tr>
<td>9</td>
<td>Status</td>
<td>Text</td>
</tr>
<tr>
<td>10</td>
<td>Server</td>
<td>Text</td>
</tr>
<tr>
<td>11</td>
<td>Landscape</td>
<td>Hexadecimal</td>
</tr>
<tr>
<td>12</td>
<td>Model Handle</td>
<td>Hexadecimal</td>
</tr>
<tr>
<td>13</td>
<td>Model Type Handle</td>
<td>Hexadecimal</td>
</tr>
<tr>
<td>14</td>
<td>IP Address</td>
<td>xxx.xxx.xxx.xxx</td>
</tr>
<tr>
<td>15</td>
<td>Security String</td>
<td>Text</td>
</tr>
<tr>
<td>16</td>
<td>Alarm State</td>
<td>Text</td>
</tr>
<tr>
<td>17</td>
<td>Acknowledged</td>
<td>Text</td>
</tr>
<tr>
<td>18</td>
<td>Clearable</td>
<td>Text</td>
</tr>
<tr>
<td>19</td>
<td>Flash_Green</td>
<td>Text</td>
</tr>
<tr>
<td>20</td>
<td>PCause</td>
<td>Text</td>
</tr>
<tr>
<td>21</td>
<td>Location</td>
<td>Text</td>
</tr>
<tr>
<td>22</td>
<td>Age</td>
<td>Integer</td>
</tr>
<tr>
<td>23</td>
<td>Notifdata</td>
<td>Text</td>
</tr>
<tr>
<td>24</td>
<td>EventMsg</td>
<td>Text</td>
</tr>
</tbody>
</table>

If USE_NEW_INTERFACE is set to TRUE, the Status, PCause, and EventMsg arguments are sent as environmental variables and argument order is therefore affected. If USE_NEW_INTERFACE is set to FALSE, you must use the following syntax in your script to read data from the PCause and the EventMsg argument into a variable as follows:

```
<variablename>=`echo "$2" | tr \350 \012 | tr \351 ''`
```

This is required in order to avoid problems with the script's parsing of the extra data caused by new lines or other special characters.

**Note:** For information on the USE_NEW_INTERFACE .alarmrc parameter, see the AlarmNotifier User Guide (1503).
SANM-Enabled .alarmrc Parameters

The AlarmNotifier resource file, .alarmrc, has several additional parameters when run on a computer that has SANM installed.

The following table shows SANM-enabled parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>APPLICATION</td>
<td>Defines the application name that identifies this AlarmNotifier application. If you use multiple AlarmNotifier applications on your network, distinguish between applications by using unique application names such as AlarmNotifier1 or AlarmNotifier2. This allows you to use unique SANM alarm-notification policies with each application. If you use the n option when invoking AlarmNotifier, the APPLICATION parameter value is ignored. The default value is AlarmNotifier.</td>
</tr>
<tr>
<td>GET_LOCATIONS</td>
<td>Lets you specify whether or not you want to be notified of the location of the device that the alarm is on. This parameter can be True (the default value) or False. If you are not interested in location information, set this parameter to False. A False setting will override any location specified as a filter parameter in an alarm-notification policy. This will reduce the network traffic.</td>
</tr>
<tr>
<td>GET_PROBABLE_CAUSES</td>
<td>True (the default value) or False. This parameter lets you specify whether you want to receive the Probable Cause text associated with each alarm. If you are not interested in Probable Cause information, set this parameter to False. This will improve the performance of AlarmNotifier.</td>
</tr>
<tr>
<td>GET_EVENTS</td>
<td>True (the default value) or False. This parameter lets you specify whether you want to receive the Event message that may be associated with an alarm. If you are not interested in event information, set this parameter to False. This will reduce network traffic generated by AlarmNotifier and improve its performance.</td>
</tr>
<tr>
<td>GET_FLASH_GREEN</td>
<td>True (the default value) or False. This parameter lets you specify whether you want to receive the Flash Green status of a model. ClearScript is the only script that displays the Flash Green status. When Flash Green is enabled for a model, the model continues to flash green even after alarms are cleared. This signals that alarms have occurred even though they no longer exist. If the value of GET_FLASH_GREEN is set to False, the Flash Green status is always passed to the ClearScript as false. If set to True, the Flash Green status is correctly passed as either False or True.</td>
</tr>
</tbody>
</table>
## SANM-Enabled .alarmrc Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSG_TIMESTAMP_FORMAT</td>
<td>The format for the time stamp on all SANM messages. The maximum length of the output string is 127 characters. Any characters other than the conversion strings will be output as text in the time stamp. The default setting is %X %x:. The colon (:) is output at the end of the time stamp. For example, to output the date/time for the current locale and the time-zone name, the string %x %X %Z is entered as the value. If left blank, no time stamp is output on the messages. If an incorrect string is entered, that string displays as text in the output.</td>
</tr>
<tr>
<td>POLICY_LANDSCAPE</td>
<td>This setting lets you specify the landscape that AlarmNotifier uses for all SANM policy definitions. This parameter works in conjunction with the POLICY_LANDSCAPE setting in SANM's .sanmrc file.</td>
</tr>
<tr>
<td>SHOW_ALL_EVENTS</td>
<td>False (the default value) or True. This parameter lets you specify whether you want to receive the most recent event for an alarm or all events generated for that alarm. If SHOW_ALL_EVENTS is False, AlarmNotifier will only forward the most recent event. For example, if an alarm was created based on a given event, and then if someone updates the status of that alarm thereby generating another event related to that alarm, AlarmNotifier only receives the status of that second event. The purpose of this is to eliminate events that may have already been forwarded. This is especially important if the size of the message is relevant, i.e. if the event message is sent through a page.</td>
</tr>
</tbody>
</table>
Chapter 6: Using SANM in a Distributed SpectroSERVER Environment

This section contains the following topics:

- Landscapes and Alarm Monitoring (see page 41)
- SANM Policy Management Across Multiple Landscapes (see page 41)

Landscapes and Alarm Monitoring

A Distributed SpectroSERVER (DSS) environment allows you to divide SPECTRUM’s network management tasks among several SpectroSERVERs. When you create a network model with multiple SpectroSERVERs, it is possible for SANM to access information from more than one SpectroSERVER at the same time.

A landscape is the SPECTRUM term for a network domain that is managed by a single SpectroSERVER. When SANM operates in a distributed environment, it monitors alarms from all landscapes. Even though different landscapes may model each other in a DSS environment, SANM-enabled applications do not receive duplicate alarm information.

Because SANM evaluates alarms across VNMs in a DSS environment, you may want to limit the type of alarm notifications that you receive. If you work in a DSS environment, you can limit the number of alarm notifications by carefully defining the parameters, Landscape, Subnet IP Address, and Device Location in the alarm notification policy.

SANM Policy Management Across Multiple Landscapes

There are two ways to configure SANM in a distributed environment. You can either create SANM policies on any landscape and allow SANM to read all policies from all landscapes, OR you can create all SANM policies on one landscape and only allow SANM to read policies from that landscape. In either case, alarm-processing applications from any landscape can be associated with the SANM policies.
How to Create SANM Policies in a Single Landscape

If you set up a distributed environment so that all policies for all landscapes are defined and managed from a single SpectroSERVER, alarm-processing applications can be installed on any of the SpectroSERVERs in the distributed environment. By setting the appropriate values in the application's resource file, the application finds the server that contains the SANM policy definitions and associates it with the appropriate policy. This configuration cuts down on the initial traffic exchanged on the network as alarm processing applications and SANM policies are associated, and it also makes managing SANM policies easier on an ongoing basis.

**Note:** It is not possible to migrate or move a SANM policy from one landscape to another. If you want to institute this configuration and already have policies defined on various landscapes, you must recreate these policies on the new landscape from which you will manage SANM policies.

To set up this configuration, do the following:

1. Change the POLICY_LANDSCAPE parameter in the .alarmrc file to the landscape handle of the SpectroSERVER where SANM is installed and policies will be created and managed.
2. Change the POLICY_LANDSCAPE parameter in the alarm-processing application's resource file (.alarmrc, .arsgrc) to the landscape handle of the SpectroSERVER where SANM is installed. This tells the application where to look for defined policies.
3. Restart the SpectroSERVER where SANM is installed and restart the alarm-processing applications so that the changes to the resource file parameters are read.
4. Open SANM, All Policies on the Locater tab of OneClick and click . The only policies available are the policies that are created on this landscape. All alarm-processing applications that have their POLICY_LANDSCAPE parameter set to the landscape handle of this landscape are seen in the applications list.

How to Create SANM Policies on Multiple Landscapes

You can set up a distributed environment so that SANM policies can be defined and managed on any SpectroSERVER in the distributed environment. Alarm-processing applications on any SpectroSERVER in the distributed environment have access to all of these policies.
Ensure the following before you set up this configuration:

- The POLICY_LANDSCAPE parameter in the .sanmrc file must have no value associated with it.
- Change the POLICY_LANDSCAPE parameter in the alarm-processing application's resource file (.alarmrc, .arsgrc, etc.) must also have no value associated with it.

Restart the SpectroSERVER where SANM is installed and restart the alarm-processing applications so that the changes to the resource file parameters are read. Open OneClick. Note that all policies that have been created within the distributed environment are available and all alarm-processing applications in the distributed environment are available for association.

**Methods for Determining Monitored Landscapes**

You can use the following methods to determine which landscapes are monitored by SANM:

- Use the SPECTRUM Command Line Interface (CLI) application to connect to the SpectroSERVER to which SANM is connected, and then enter `show landscapes` on the command line. The CLI application displays a list of all the landscapes modeled in that server.

  **Note:** See the *Command Line Interface User Guide (0664)* for more information.

- Open any one of the detailed trace files you specified for SANM-enabled applications. A trace file indicates the connection status of each landscape in the landscape map for the SpectroSERVER to which SANM is connected to that SpectroSERVER. Trace files are stored by default in a trace directory in a SANM-enabled application’s home directory.
Index

A
acknowledging alarms • 34
adding
alarms to a policy • 19
filters to a policy • 16
models to a policy • 19
alarm conditions • 9
alarm management parameters • 33
alarm monitoring • 9, 41
alarm notification policies • 13, 19, 21
alarm resource file • 11
AlarmAck • 31, 34
AlarmAge • 35
AlarmClear • 31, 34
AlarmNotifier • 10, 31
alarm-processing applications • 41
alarms
receiving from the cache • 33
APPLICATION • 31, 39
applications, associating policies with • 22
assignticket • 24
associations
about • 21
scheduling • 23
Attention! • 11

B
backups • 11

C
ClearScript • 31, 36
clearticket • 24, 25
command options
- n • 31
- p • 31
- r • 31
- tl • 31
- tn • 31
- ts • 31
creating
alarm notification policies • 13
detailed trace file • 29
Distributed SpectroSERVERs • 41

E
email notifications • 36
event cache • 33
event codes • 27
EventMessage • 35
events • 27

F
filters
adding a policy to • 16
additional parameters for • 18
changing the order of • 17
defining • 14
deleting • 18
editing • 17
introducing additional parameters for • 18
FlashGreen • 35

G
GET_EVENTS • 39
GET_FLASH_GREEN • 39
GET_LOCATIONS • 39
GET_PROBABLE_CAUSES • 39

L
landscapes • 41
Location • 35

M
monitored landscapes • 43

N
-n command option • 31
NotificationData • 14, 35

P
parameters • 35, 39
policies • 22, 41, 42
policy associations • 23
POLICY_LANDSCAPE • 39
ProbableCause • 35

R
-r command option • 31

S
SANM
  about • 9
  default policy • 21
  event codes • 27
  events • 27
  policies • 41, 42
SANM-enabled functionality • 31
SANM-enabled script parameters
  AlarmAge • 35
  EventMessage • 35
  FlashGreen • 35
  Location • 35
  NotificationData • 35
  ProbableCause • 35
scheduled associations • 23
Scheduled Entries list • 23
Scheduled Policies table • 23
Scheduler • 23
script parameters • 35
SetScript • 31, 36
show landscapes command • 43
summary trace file • 28

T
third-party applications • 36
-tl command option • 31
-tn command option • 31
Trace • 31
trace files • 43
trace log options • 31
tracing policies • 28
-ts command option • 31

U
updatealarm • 24, 25
UpdateScript • 31, 36
USE_NEW_INTERFACE • 36