Nakina
Systems
Network Integrity

Release 8

NI-Director Operations Console Installation and Administration Guide

April 02, 2013

The content of this document is subject to change without notice.

Table of contents

1. Revision history 3

2. NI-Director Operations Console overview 4

2.1 What you can do with the NOC 4

2.2 Network Integrity and the NOC 4

2.2.1 Network adapters and the NOC 5

2.3 Base and optional plugins 5

3. Installing and configuring the NI-Director Operations Console 8

3.1 Check system requirements 8

3.1.1 NOC system requirements for Windows. 8

3.1.2 NOC system requirements for Solaris 8

3.2 Install the NOC software. 8

3.3 (Optional) Configure custom NE menu items. 10

3.3.1 Default ne_launch_pad.xml file. 11

3.4 (Optional) Configure the Command and Control plugin 15

3.4.1 Command and Control configuration summary 15

3.4.2 TL1 command file overview 16

4. Upgrading from a previous version of the NOC 23

4.1 Save your customized files to a temporary location 23

4.2 Uninstall the NOC 23

4.3 Install the Release 8 NOC. 23

4.4 Replace the customized files in the new NOC install directory 23

5. Administering the NOC 25

5.1 Change the login server, port or protocol 25

5.2 Add applications to the NOC launch-pad 26

5.3 View plugin details 28

5.4 Uninstall the NOC software 28

5.5 Configure online help for Solaris. 29

Appendix A: Sample TL1 command file 31

1 Revision history

The following table provides the revision history for the NI-Director Operations Console (NOC) Installation and Administration Guide. Each new version of this document supersedes all earlier versions until re-issued.

Issue	Description
April 02, 2013	First release of document for NI-Director Operations Console Release 8 or higher unless re-issued

2 NI-Director Operations Console overview

This section introduces the NOC and provides a summary of NOC functionality.

The NI-Director Operations Console (NOC) is a thick-client Windows or Solaris application for Network Integrity. It allows network operators to monitor and troubleshoot a multi-vendor network from a single graphical user interface (GUI). Authorized users can view network topology, alarm conditions, and performance monitoring information, as well as detailed equipment information right down to the interface level. They can also manage cross-connects and manual topology links.

The NOC handles thousands of network elements from multiple vendors consistently using the same GUI and command set available, making it easy to learn and use. New network element types can be managed simply by adding a hot-deployable adapter to the Network Integrity Framework.

If you already have an understanding of the NOC, you can proceed to install the NOC, log in and manage the network. See “Installing and configuring the NI-Director Operations Console” on page 8.

2.1 What you can do with the NOC

From the NOC, authorized users can perform the tasks listed below. These tasks are described in the NOC User Guide or online help.

- configure equipment
- provision Ethernet services
- monitor the network for alarms at the NE group, NE, or circuit pack level
- monitor performance
- manage interfaces
- manage SONET cross-connects
- manage manual topology links
- send maintenance commands to NEs, circuit packs, ports, and interfaces
- use plugins, customized scripts, or proprietary NE tools to communicate with an NE.
- manage virtual concatenation groups (VCGs)

Some of these tasks are dependent on the plugins that are installed and may not be available, depending on your license agreement. See “Base and optional plugins” on page 5 for details.

You may also have additional tasks available if you developed custom plugins. Custom plugins are beyond the scope of this document.

2.2 Network Integrity and the NOC

The NOC obtains its network data from the Network Integrity database. The Network Integrity database continuously monitors the network for autonomous messages from the NEs. Each time an autonomous message is received, the database is updated and the information is reflected in the NOC.

The NOC uses Network Integrity to provide network-wide security across all multi-vendor elements in the managed network. The Network Integrity security configuration dictates which users can log in and which NE groups a user is authorized to manage. Only authorized Nakina users can access the NEs contained in their assigned NE groups, preventing unauthorized access to the entire network.

2.2.1 Network adapters and the NOC

Network adapters are the interface between Network Integrity and specific models of vendor NEs. Adapters are hot-deployable and can be installed as required to manage a wide variety of vendor-specific NEs.

In order to use the NOC to perform management tasks on an NE, the task must be supported by both the NE and by the installed adapter.

☞ Note: Each adapter that has been certified by Nakina Systems includes an adapter release note, which is a document that provides important information about the adapter, the NE, and the implementation and support for Network Integrity applications. Before deploying an adapter and using the NOC to manage an NE, administrators and operators must read and become familiar with the adapter release note issues.

2.3 Base and optional plugins

The NOC uses a combination of base plugins and feature plugins to build the functionality available in the user interface. In general, you enable these plugins when you install the NOC, but a procedure is also provided to install plugins after you have installed the NOC ("Configuration data for the TL1 command XML file" on page 17).

Optional feature plugins provide additional functionality as shown in Table 2–1 and can be enabled based on your license agreement.

The list is organized in the order that the plugins appear in the installation wizard.

Table 2–1: Optional NOC plugins

Plugin	Description
Equipment Configuration	Allows users to view and set values for attributes on network elements, shelves, circuit packs, slots, ports, pluggable transceiver modules, and interfaces.
Application Launcher	Allows users to launch applications or executables: • for an NE (for example, to access an NE's element management system (EMS)) without having to log in to each one • globally (not in the context of a specific NE). See the Network Integrity Guardian Configuration and Management Guide for configuration information. If this option is configured, an Application menu is added to the NOC to the right of the Help menu.

Plugin	Description
Command and Control	Provides two options: • TL1 Command and Control — Allows users to either type a TL1 command or choose from a list of predefined TL1 commands that authorized users can execute for selected NEs. • RTRV-COND and RTRV-AO Value Pack — Provides additional tabs and stand-alone windows for automatically viewing outstanding conditions and autonomous messages. You can select one or both options.
Historical Event View	Allows users to find historical events on the selected NE for any time period.
NE LaunchPad	Allows administrators to configure the NOC with customized right-click menus that allow a user to access and launch scripts, applications or proprietary NE tools. Menu items are displayed in context with the vendor and model of NE selected. Configuration instructions are provided in “(Optional) Configure custom NE menu items” on page 10. Note: The NE LaunchPad plugin has been superceded by the Application Launcher plugin and will be discontinued in an upcoming release. Nakina recommends that you use the Application Launcher plugin.
SONET Cross Connect Manager	Allows users to view, create, modify, and delete cross-connects on an NE.
Performance Monitoring Viewer	Allows users to view the most current performance monitoring (PM) counts and the last completed binned counts for an NE interface.
Quick Login	Stores login information so that when the NOC is launched, the user is automatically logged in and taken directly to the main NOC view without entering username, password and server details. Use this plugin in place of Standard Login. The data is stored in a quicklogin.config file in the user's directory under <NOC>/config/users. The password parameter is encrypted in the file but all other attributes are stored in plain text. Note: If you enable this plugin, you must disable the Standard Login plugin. The Standard Login plugin is a base plugin that provides the standard login screen with no option to save the settings. The user must provide the username, password and server details each time the NOC is launched. Note: If you enable the Quick Login plugin, you should disable the Standard Login plugin.
Secure NE Access	Provides two options: • FTP — Allows you to establish a File Transfer Protocol (FTP) or SSH File Transfer Protocol (SFTP) session to an NE. • Telnet — Allows you to establish a Telnet session or a Secure Shell (SSH) session with an NE. You can select any combination of options.
VCG Manager	Allows you to troubleshoot a virtual concatenation group (VCG) on a network element.

Plugin	Description
Launch SONET Domain Controller	Enables you to launch the SONET Domain Controller (SDC) from the NOC. SDC adds subnetwork-level capabilities to Network Integrity, including the discovery and provisioning of subnetwork connections (SNCs). SDC is installed separately from the NOC. See the SDC documentation for more information.
Ethernet Service Provisioning	Allows users to view and set values for Ethernet services, such as Direct Internet Access, E-LAN, and E-Tree.

3 Installing and configuring the NI-Director Operations Console

This section contains the following information and procedures:

- "Check system requirements" on page 8
- "Install the NOC software" on page 8
- "(Optional) Configure custom NE menu items" on page 10
- "(Optional) Configure the Command and Control plugin" on page 15

For an overview of the NOC features, see "NI-Director Operations Console overview" on page 4.

3.1 Check system requirements

This section provides the minimum specification for installing and running the NOC on Windows or Solaris. To accommodate larger networks and more data, increase the system specifications.

3.1.1 NOC system requirements for Windows

The minimum system requirements for the Windows computer on which the NOC is installed are:

- Microsoft Windows 32-bit operating system
- 1.5 gigahertz processor
- 1 GB of RAM for each NOC session that is running
- 250 megabytes (MB) of available hard disk space
- DVD drive or access to a server that contains the NI-Director Operations Console software

3.1.2 NOC system requirements for Solaris

The minimum system requirements for the Solaris computer on which the NOC is installed are:

- 32-bit Solaris server
- 1.5 gigahertz processor
- 1 GB of RAM for each NOC session that is running
- 250 megabytes (MB) of available hard disk space
- DVD drive or access to a server that contains the NI-Director Operations Console software

3.2 Install the NOC software

Use this procedure to install the NOC software on either a Windows or Solaris computer using the installer wizard, for example:

- for Windows: NOC-GA-8-2-n.exe
- for Solaris: NOC-GA-8-2-n_SOLARIS.bin

Note: You may have been provided with a custom installer. If so, follow the instructions in the installer.

1. Copy the required NOC installer for your platform from the Network Integrity software DVD or server to your computer.

2. Launch the installer for your platform, for example:

   - for Windows, double-click the NOC-GA-8-2-n.exe file
   - for Solaris, type: sh ./NOC-GA-8-2-n_SOLARIS.bin

   The NI-Director Operations Console installation wizard opens.

3. Read the introductory text and click **Next**.

4. On the **Choose Optional Nakina Open Console Plugins** screen:

   a. Choose an option:

      - Basic (go to Step 6.)
      - Basic with Optional Plugins (go to Step 5.)

   b. Click **Next**.

5. Select each of the optional NOC plugins that you are licensed to use.

   The main NOC application (Base) is always installed.

   For a description of each plugin, see “Base and optional plugins” on page 5.

6. On the **Choose Install Folder** screen, do one of the following:

   - Click **Choose**, and in the **Browse for Folder** dialog box, select the folder where you want to install the NOC and click **OK**.
   - Accept the default location, which is indicated in the installer.

7. Click **Next**.

8. On the **Choose Shortcut Folder** screen for Windows, or the **Choose Link Folder** for Solaris, select the shortcut or link options for the NOC.

9. Click **Next**.

10. On the **SSL Keystore File** screen, choose one of the following options and click **Next**:

    - Yes
    - No

    If you chose No, the system uses a default keystore file. Go to Step 12.

    If you chose Yes, you must provide your own SSL keystore file. Go to Step 11.

    Note: If you want to change the keystore file after installation is complete, you must uninstall and reinstall the software, specifying the new file.

11. On the second **SSL Keystore File** screen, click **Choose**, navigate to the file that you want to use and click **OK**.

    The file is copied to the config/login folder in the installation folder.

12. Click **Next**.

13. On the **Enter Default Web Tier Server** screen, enter the primary and, optionally, secondary host names of the Network Integrity web tier servers and port to which the NOC will connect.

    Separate multiple entries with commas.

    When the user logs in to the NOC, the system selects a web tier from the list randomly. It attempts to use a server from the primary host names list first. If none are available, the system will attempt to use one from the secondary list.

    At a minimum, you must enter a primary host name.

14. Click **Next**.

15. On the **Select Default Web Browser** screen, do the following:

    a. Click **Choose**, navigate to the location of the web browser that you want to use (for example, to launch Network Integrity Framework applications) and click **Open**.

    b. Click **Next**.

16. Review the **Pre-Installation Summary** and click **Install**.

    When the installation is complete, the Install Complete screen is displayed. If there are any warnings about the install, a note is displayed on this screen.

17. Click **Done** to close the installer.

    Note: You may need to choose an option for restarting your computer before you click Done.

    The wizard closes.

3.3 (Optional) Configure custom NE menu items

Note: The NE LaunchPad plugin has been superceded by the Application Launcher plugin and will be discontinued in an upcoming release. Nakina recommends that you use the Application Launcher plugin.

Use this procedure to configure the optional NE Launch-pad plugin. The NE Launch-pad plugin allows administrators to configure the NOC with customized right-click menu selections that launch applications in context with the currently selected vendor and model of NE.

Launch-pad menu selections are added to a NOC deployment by editing the **ne_launch_pad.xml** file in the **<**NOC**>\config\plugins\nelaunchpad** directory.

When the NOC is launched the information in the xml file is parsed to build a collection of menus and sub-menu commands for each specified vendor and model of NE.

If you are going to configure the launch-pad for a TL1 or CLI telnet session to an NE, the vendor-specific port information is documented in the Adapter Notes that came with the adapter package.

Note: To maintain the usability of the sub-menu, you should try to limit the number of menu selections when defining a list of menu commands.

3.3.1 Default ne_launch_pad.xml file

The following is an example of the default ne_launch_pad.xml file that you edit and add launch_pad commands. Remove the comment tags from the <ne> element and configure the launch_pad elements and parameters according to Table 3–1.

```
 <ne-launch-pad>
  <menu menu_name="Menu name">
  <!-- valid variables: ${ne_name} ${ne_ip} -->
   <!-- vendor and model attributes are mandatory-->
  <!-- absolute path must be provided in the command string.
MANDATORY -->
  <!-- params is OPTIONAL -->
  <!-- management_state="Managed" OPTIONAL, valid
values={Managed, Unmanaged, Under Test} -->
  <!-- communication_state="In Contact" OPTIONAL, valid
values={In Contact, Out of Contact} -->
  <!-- confirmation="true" OPTIONAL, valid values={true,
false} default to false -->
  <!-- label="Sub-menu name" MANDATORY and UNIQUE-->
  <!-- See Online Help for more details -->
  <!-- <ne vendor="Network Element vendor" model="Network
Element model">
   <command_tag label="Sub-menu name"
    command="c:\...\application name"
    params="http://${ne_ip}"
    management_state="Managed"
    communication_state="In Contact"
    confirmation="false"/>
  </ne> -->
 </menu>
 </ne-launch-pad>
```

Table 3–1 describes the mandatory and optional configuration data for the ne_launch_pad.xml file.

Table 3–1: Configuration data for the ne_launch_pad.xml file

Element	**Attribute**	**Description and values**
ne-launch-pad		Mandatory - parent element for the xml file
menu	menu_name=”Menu name”	Mandatory - provides the top level menu selection that will appear for all NEs when the user right-clicks on an NE in the Network Elements table or Topology view. For example, the following example would display “Reports” as a menu selection for all NEs. <ne-launch-pad> <menu menu_name="Reports"> After you define the top level menu command, use the <ne> element to define the submenu commands for specific a vendor/model of NE.
ne	vendor=”NE vendor”	Mandatory - The name of the vendor of the network element, such as <ne **vendor="nortel"** model="OPTera Metro 3500 MSP">. The entry is not case sensitive.
	model=”NE model”	Mandatory - The model of the network element, such as <ne vendor="nortel" <ne vendor="Nortel" **model="OPTera Metro 3500 MSP"**>. The entry is not case sensitive.

Element	Attribute	Description and values
command_tag (Mandatory attributes)		Parent element that defines the submenu command. You can have as many command_tag elements under an <ne> element as long as each label attribute is unique for each command_tag within the <ne> element. For example, you could have two commands for the Nortel OpteraMetro 3500 NE as follows: <ne vendor="nortel" model="OPTera Metro 3500 MSP"> **<command_tag** **label="Web session"** command="C:\Program Files\Internet Explorer\IEXPLORE.EXE" /> **<command_tag** **label="TL1 session"** command="C:\Program Files\Accessories\Telnet.EXE" /> </ne>

	label=”sub-menu name”	Mandatory: This attribute uniquely identifies the command_tag and defines what gets displayed as the name of the submenu selection in the NOC. In this example “Web session” would appear in the right-click submenu:
		<command_tag
		label="Web session"
		command="C:\Program Files\Internet Explorer\IEXPLORE.EXE"
		params="http://${ne_ip}"
		management_state="Managed"
		communication_state="In Contact"
		confirmation=”true” **/**>

	command=”path”	Mandatory - Type the absolute path to the command executable, as in the following example:
		<command_tag
		label="Web session"
		command="C:\Program Files\Internet Explorer\IEXPLORE.EXE"
		params="http://${ne_ip}"
		management_state="Managed"
		communication_state="In Contact"
		confirmation=”true” **/**>

Element	**Attribute**	**Description and values**

command_tag (Optional attributes)	params=”parameters”	Optional - if specified passes a parameter to the menu command. You can use variables to pass the NE Name $(ne_name) or NE IP Address $(ne_ip), as in the following example:
		<command_tag
		label="Web session"
		command="C:\Program Files\Internet Explorer\IEXPLORE.EXE"
		params="http://${ne_ip}"
		management_state="Managed"
		communication_state="In Contact"
		confirmation=”true” **/**>
		You can also pass port information for telnet sessions as in this example:
		params=”${ne_ip} 3082”

	management_state=”state”	Optional - if specified, the menu command will only be displayed if the NE management state matches the defined value (Managed, Unmanaged, or Under Test). In this example, the NE must be in the Managed state:
		<command_tag
		label="Web session"
		command="C:\Program Files\Internet Explorer\IEXPLORE.EXE"
		params="http://${ne_ip}"
		management_state="Managed"
		communication_state="In Contact"
		confirmation=”true” **/**>

	communication_state=”state”	Optional - if specified, the menu command will only be displayed if the NE communication state matches the defined value (In Contact or Out of Contact). In this example, the NE must be in the In Contact state:
		<command_tag
		label="Web session"
		command="C:\Program Files\Internet Explorer\IEXPLORE.EXE"
		params="http://${ne_ip}"
		management_state="Managed"
		communication_state="In Contact"
		confirmation=”true”**/**>

	confirmation value=”value”	Optional - If the confirmation value is set to ”true”, the NOC prompts the user for confirmation before executing the command. If the confirmation attribute is set to “false”, the NOC does not prompt the user for confirmation. If not defined, the default is “false”.

1. Navigate to the folder where the NOC files were installed.
2. Open the **plugins/nelaunchpad** folder.

3. Using a text editor or an XML editor, open the **ne_launch_pad.xml** configuration file.

4. Edit the file and create your menu and submenu selections based on the information in Table 3–1 on page 12.

5. Save and close the file.

   The NE launch pad items will be available in the NOC the next time it is started.

6. Repeat this procedure on each computer where the NOC is installed, or copy the file to other computers that are running the NOC.

3.4 (Optional) Configure the Command and Control plugin

Use this procedure to configure predefined commands for the optional Command and Control plugin. Authorized users can execute these commands on the currently selected NE if they have permission to do so.

Configuring predefined commands is optional. If you do not configure commands, users can type commands directly into the Command field, provided they have the appropriate permissions.

3.4.1 Command and Control configuration summary

1. Create a folder called **TL1Commands** in the **<**NOC**>/config** directory. The TL1Commands folder will contain the **EnableVendorModels.xml** file and a directory for each vendor.

2. Create a file called **EnableVendorModels.xml** in the **<**NOC**>/config/TL1Commands** directory. In this file, define each of the vendor/models of NE that will require TL1 Command and Control.

3. For each NE vendor that will require TL1 Command and Control, create a directory with the vendor’s name, such as **fujitsu** in **<**NOC**>/config/TL1Commands** directory.

4. Within each vendor-specific directory in **<**NOC**>/config/TL1Commands** create a **<model>.TL1** file that contains the TL1 commands for a specific model of NE. This file will contain the NE-, card- and port-level TL1 commands in an XML format.

Note: For the purposes of this document, the terms “card” and “circuit pack” are synonymous.

Sample EnableVendorModels.xml file

This file defines the NE Vendor and Model types that will have TL1 commands defined, as shown in the following example:

```
<command-integrator>
  <plugins>
  <ne vendor="nortel" model="OPTera Metro 3500 MSP"/>
  <ne vendor="nortel" model="OPTera Metro 3000 MSP Series NP"/>
  <ne vendor="fujitsu" model="Flashwave4500"/>
  <ne vendor="ciena" model="Corestream"/>
</plugins>
</command-integrator>
```

3.4.2 TL1 command file overview

For each model of network element, you must create a **<model>.TL1** file that lists the NE-, card- and port-level TL1 commands that can be executed on that specific model of NE. For identification, the name of the command definition file must correspond to the value of the “model” field from the “EnableVendorModels.xml” file with all capital letters converted to lower case and all spaces replaced by underscores "_". For example, if the model in the “EnableVendorModels.xml” is **OPTera Metro 3500 MSP**, the TL1 command file must be named **optera_metro_3500_msp.TL1**.

You must also define the format of the responses that the NE can provide.

Note: An example of a completed XML file along with how it appears in the user interface is provided in “Sample TL1 command file” on page 31.

The structure of the command definition file is as follows:

```
<TL1Commands>
  <table-response>
    <table name="table 1">
      <column
        block="1"  <!–TL1 block index ◊
        index="6"  <!--TL1 field index inside a block ◊
        name="LOCN"  <!–Column name ◊
        description="Location near end or far end"  <!–Column
          description ◊
        visible="true" />  <!–Column visibility ◊

        …
      <column …. />
  </table>
  <table name=”table 2>
    ….
  </table>
  …
  </table-response>

  <table-event>  <!–This table-event is not supported yet -◊
  </table-event>

  <ne-level>
    <tl1command description="…" confirm="false">
        …
    <response-table name="table 1" />
    </tl1command>
  </ne-level>
  <card-level>
    <tl1command description="…" type="common" confirm="false">
         …
    <response-table name="table 2" />
    </tl1command>
  </card-level>
  <port-level>
    <tl1command description="…" type="common" confirm="false"
    confirmtext=”…”>
    </tl1command>
  </port-level>
</TL1Commands>
```

Table 3–2 describes the elements, attributes and values that make up the TL1 XML file.

Table 3–2: Configuration data for the TL1 command XML file

Element	Attribute	Description and values
TL1Commands		mandatory - Parent element for the TL1 command file.
table-response		Child element of <TL1Commands>. Parent level for tables that define response formats for NEs, cards, and ports.
table	name=”string”	Child element of <table-response>. Parent level for responses that will be displayed when a command is executed from an NE, card, or port. The name is used at each command level (NE, card, and port) to identify the response format at that level. For example, to use a response table called “table 1” to format the responses at the NE level, you would specify “response-table name=”table1” within the <ne-level> element. This is shown in the command definition structure earlier in this section.
column		Child element of <table>. Defines the values for a col.umn in the response table.
	block=”integer”	The TL1 block index.
	index=”integer”	The TL1 field index inside a block.
	name=”string”	The name of the column.
	description=”string”	A description of the column contents.
	visible=”string”	Column visibility. Value is “true” if the column is visible, “false” if it is not visible.

Element	Attribute	Description and values
ne-level		Child element of <TL1Commands>. Parent level for NE commands that will be available from an NE selected in the Network Elements table, Topology view or SLG.
card-level		Child element of <TL1Commands>. Parent level for card commands that will be available from a card selected in the SLG.
port-level		Child element of <TL1Commands>. Parent level for port commands that will be available from a port selected in the SLG.
tl1command		mandatory - Child element to <ne-level>, <card-level> or <port-level>. Defines the details of the well-formatted TL1 command sent to the TID of the object that is selected in the NOC. The colon-separated blocks that make up any command are as follows: `<command>:TID:<aid>:CTAG:<general>:<payload-1>:<payload-2>:<payload-3>:<payload-4>:<payload-5>:<payload-6>:<payload-7>:<payload-8>`
	description=”string”	mandatory - Defines the text string that will appear in the menu of commands displayed in the NOC.
	type=”string”	optional - Used only for card-level and port-level commands: “common” applies to all cards and ports, or Used only for card-level and port-level commands: defines the entity type that the TL1 command is valid for. The defined string must correspond to the Inventory entity type. Default is “common”. If the values match, then the command is presented in the list of available commands for the user to select. If the values do not match, then the command is not presented in the list of available commands. If no type field is specified then the command is assumed to apply to all "Entity Types".
	confirm=”value”	mandatory - If the confirm value is set to ”true”, the NOC prompts the user for confirmation before executing the command. If the confirm attribute is set to “false”, the NOC does not prompt the user for confirmation.
	confirmtext=”string”	If confirm=”true” this attribute can be used to define the text string that will appear in confirmation dialog displayed to the user.

Element	Attribute	Description and values
command		mandatory - Child element of <tl1command>. The first block of the <tl1command> that defines the actual TL1 command to be executed, such as <command>RTRV-ALM-ALL</command>. The following additional elements can be used to refine the command: `TID:<aid>:CTAG:<general>:<payload-1>:<payload-2>:<payload-3>:<payload-4>:<payload-5>:<payload-6>:<payload-7>:<payload-8>`
aid		mandatory - Used to define Access Identifier (aid) for the <tl1command> to be sent to the device. The following variables can be used: • **%ENTITYTYPE** This parameter will be replaced with the value of the entity type attribute of the object that is in context when the TL1 Command and Control is invoked. Note: %ENTITYTYPE should not be used as a parameter for Command and AID blocks if the entity types reported from the Network Integrity Inventory do not match the necessary TL1 command entity types. • **%SLOT** This parameter will be replaced with the number of the slot containing the circuit pack that is in context when the TL1 Command and Control is invoked. This parameter is only valid if the selected object is a card or port. If the selected object is not a card or port this parameter will be set to "". • **%PORT** This parameter will be replaced with the number of the port that is in context when the TL1 Command and Control is invoked. This parameter is only valid if the selected object is a port. If the selected object is not a port this parameter will be set to "".
general		optional - This element represents the General Block of a TL1 command. If not defined, the <tl1command> will be sent with this block empty.
payload-1		optional - This element is the first block of the Message Payload Block of a TL1 command and is the fifth block of the <tl1command>. If not defined, the <tl1command> will be sent with this block empty. If no other Message Payload Blocks are defined for the <tl1command>, then the tl1command will be terminated with a semi-colon and sent as is.
payload-2		optional - This element is the second block of the Message Payload Block of a TL1 command and is the sixth block of the <tl1command>. If not defined, the <tl1command> will be sent with this block empty. If no other Message Payload Blocks are defined for the <tl1command>, then the tl1command will be terminated with a semi-colon and sent as is.

Element	Attribute	Description and values
payload-3		optional - This element is the third block of the Message Payload Block of a TL1 command and is the seventh block of the <tl1command>. If not defined, the <tl1command> will be sent with this block empty. If no other Message Payload Blocks are defined for the <tl1command>, then the tl1command will be terminated with a semi-colon and sent as is.
payload-4		optional - This element is the fourth block of the Message Payload Block of a TL1 Command and is the eighth block of the <tl1command>. If not defined, the <tl1command> will be sent with this block empty. If no other Message Payload Blocks are defined for the <tl1command>, then the <tl1command> will be terminated with a semi-colon and sent as is.
payload-5		optional - This element is the fifth block of the Message Payload Block of a TL1 command and is the ninth block of the <tl1command>. If not defined, the <tl1command> will be sent with this block empty. If no other Message Payload Blocks are defined for the <tl1command>, then the <tl1command> will be terminated with a semi-colon and sent as is.
payload-6		optional - This element is the sixth block of the Message Payload Block of a TL1 command and is the tenth block of the <tl1command>. If not defined, the <tl1command> will be sent with this block empty. If no other Message Payload Blocks are defined for the <tl1command>, then the <tl1command> will be terminated with a semi-colon and sent as is.
payload-7		optional - This element is the seventh block of the Message Payload Block of a TL1 command and is the eleventh block of the <tl1command>. If not defined, the <tl1command> will be sent with this block empty. If no other Message Payload Blocks are defined for the <tl1command>, then the <tl1command> will be terminated with a semi-colon and sent as is.
payload-8		optional - This element is the eighth block of the Message Payload Block of a TL1 command and is the twelfth block of the <tl1command>. If not defined, the <tl1command> will be sent with this block empty. If no other Message Payload Blocks are defined for the <tl1command>, then the <tl1command> will be terminated with a semi-colon and sent as is.

Element	Attribute	Description and values
parameter	index=”integer”	Defines additional parameters to be added to the blocks that comprise the <tl1command>. The parameter element can be added to the blocks <general> through <payload-8> to collect information from the NOC user through menus or input fields. (For an example see “Sample TL1 commands as they appear in the NOC user interface” on page 40). The value of the index parameter is a number that partitions the specific TL1 block into multiple comma-separated fields. The value of the index parameter indicates in which comma-separated field the collected information should be placed. Multiple <parameter> blocks can be used within one XML command block specification. Each <parameter> block should specify a different value for its index. If an index value is re-used, then the <parameter> block that specifies the particular index value last in the file will be retained and the previous <parameter> block that specified the index value will be discarded. The number of comma-separated fields created within the associated TL1 block will equal the highest index value of all parameter blocks contained within a specific XML command block. If no <parameter> block specifies an index value associated with a field to be created, then the field will be left empty and separated from the next field by a comma. For example if there are two <parameter> blocks within a particular command block and one parameter has index="3" and the other has index="5", then the TL1 block that is sent will appear as: :,,,parmValue1,,parmValue2:
optiontype		mandatory within the <parameter> element - Defines the way the <parameter> data is to be collected from the user: “text” for a text field, “pulldown” for a menu of options (requires one or more option elements), or “static” for static parameter that is not shown in the options menu.
optiondescription		mandatory within the parameter element - Any string used to provide a label for the <parameter> optiontype, such as “Host IP Address” or “Card types”.
option		optional - Used only when optiontype is “pulldown”. Defines a set of values to use as menu items. The values contained within each <option> element are appended to the menu in the order they are specified in the file.
optionprepend		optional - Defines a string to prepend to the collected value for the <parameter> when the value is being added to the TL1 command. This block is used when the collected <parameter> is to be sent as a name-defined <parameter> in the form <parameterName>=<parameterValue>. For example, MSGTYPE=ALM.
response-table	name=”string”	mandatory - Child element of <tl1command>. Identifies the response table to use to format the command response.

4 Upgrading from a previous version of the NOC

This section describes how to upgrade from a previous version of the NOC. It explains how to maintain any customizations that you have made to the NOC.

These instructions apply to both Windows and Solaris installation, however, the examples used are windows examples.

The contains the following steps:

- Save your customized files to a temporary location.
- Uninstall the NOC.
- Install the Release 8 NOC.
- Replace the customized files in the new NOC install directory.

4.1 Save your customized files to a temporary location

The files that you need save depend on the features that you have customized.

1. Navigate to the NOC installation directory, for example:

   C:\Program Files\Nakina\Nakina Open Console

2. Save the launch_pad.xml file from the \plugins\launchpad directory to a temporary location.

3. Save the ne_launch_pad.xml file from the \plugins\nelaunchpad directory to a temporary location.

4. If the Command and Control plugin is configured, save the contents of the \config\TL1Commands directory to a temporary location.

4.2 Uninstall the NOC

Once you have saved your customized files, you can do the following:

1. Uninstall the NOC. See “Uninstall the NOC software” on page 28

2. Delete the NOC installation directory (for example, the Nakina Open Console directory in the C:\Program Files directory).

4.3 Install the Release 8 NOC

See “Install the NOC software” on page 8

4.4 Replace the customized files in the new NOC install directory

The files that you need to replace depend on the features that you have customized.

1. Navigate to the Release 8 NOC installation directory, for example:

   C:\Program Files\Nakina\Nakina Open Console

2. Copy the launch_pad.xml file from the temporary location to the \plugins\launchpad directory.

3. Copy the ne_launch_pad.xml file from the temporary location to the \plugins\nelaunchpad directory.

4. If the Command and Control plugin is configured, copy the contents of the \config\TL1Commands directory from the temporary location to the \config\TL1Commands directory in the installation directory.

5. If you modified your NOC to use a non-SSL protocol, “Change the login server, port or protocol” on page 25.

5 Administering the NOC

This section contains the following procedures for administering and performing additional configuration of the NOC:

- “Change the login server, port or protocol” on page 25
- “Add applications to the NOC launch-pad” on page 26
- “View plugin details” on page 28
- “Uninstall the NOC software” on page 28
- “Configure online help for Solaris” on page 29

☞

Note: If while using the NOC you do not see menu items, tabs or buttons that are described in this documentation, it is because the Network Integrity administrator has not assigned the permission to your user account.

5.1 Change the login server, port or protocol

Use this procedure if you want to change the login settings that were defined during the installation of the NOC. Changes are made by editing the login.config file as described in the table “login.config file parameters” on page 25. Changes made to this file take effect the next time the NOC is launched.

Table 5–1: login.config file parameters

Parameter	Description
primary.webtier.hostnames=<host, host1, host2...>	Defines the primary Web tier servers to which the NOC will connect. If more than one server is defined, the system attempts to connect to one of the servers in the list randomly, and if a connection is not successful, the system tries each additional server sequentially until a connection is made. If a connection can not be made to any of the primary servers, the system attempts to use the secondary servers. If no connection can be made, an error message is displayed to the user. where <web_tier> is a comma-separated list of the primary web tier servers Replace any of the host names with the new name of the web tier server or add a new server name to the list.

Parameter	Description
secondary.webtier.hostnames=<host, host1, host2...>	Defines the secondary Web tier servers to which the NOC will connect. If more than one server is defined, the system attempts to connect to one of the servers in the list, and if a connection is not successful, the system tries another server in the list, until a connection is made. If a connection can not be made to any of the secondary servers, an error message is displayed to the user. where <web_tier> is a comma-separated list of the secondary web tier servers Replace any of the host names with the new name of the web tier server or add a new server name to the list.
http.port=<port>	Defines the default HTTPS login port that the NOC uses to access the web tier server. where <port> is the HTTPS port number. Default: Port 8012

1. Navigate to the folder where the NOC files are installed.
2. Open the folder **/config/login**.
3. Use any text editor to open the file **login.config**.
4. Change the desired parameters as described in the table “login.config file parameters” on page 25.
5. Save and close the file.
6. Repeat this procedure on each computer where the NOC is installed, or copy this **login.config** file to the same folder on other computers that are running the NOC and overwrite the existing file.

5.2 Add applications to the NOC launch-pad

Use the following procedure to configure the NOC launch-pad with buttons for launching third-party applications. This procedure is optional.

The icons that appear in the toolbar can be any Java-compatible graphic, such as GIF, JPG, PNG, or TIFF. Keep each icon graphic small because the NOC does not scale the graphic to fit in the toolbar. A size of approximately 24 x 24 pixels works well.

1. Navigate to the folder where the NOC files are installed.
2. Open the **plugins/launchpad** folder.
3. Using a text editor or an XML editor, open the **launch_pad.xml** configuration file.

   The following example shows the launch_pad.xml file configured for two applications: Pave Web and ICO.

```
<launch-pad>
  <plugins>
    <plugin name="Pave Web"
```

```
        icon="C:/Program Files/Nakina/Nakina Open Console/config/brand/
pave.jpg"
        separator="true"
        command="C:\Program Files\PaveWeb.exe"
        <description> Pave Web application</description>
        </plugin>

        <plugin name="ICO"
        icon="C:/Program Files/Nakina/Nakina Open Console/config/brand/
ICO.png"
        command="C:\Program Files\ICO\icon.exe">
        <description> ICO application </description>
        </plugin>
      </plugins>
  </launch-pad>
```

4. Edit the launch_pad.xml configuration file with the data described in Table 5–2.

Table 5–2: Configuration data required for a third-party application

Parameter	Description
plugin name=	Type a brief name for the application. This value appears in the NOC toolbar.
icon=	Type the full path to the graphic file that contains the application icon that will appear in the NOC toolbar, for example, C:/Program Files/ Nakina/Nakina Open Console/config/brand/ pave.jpg.
separator=	Type “true” to add a separator line between launch pad icons. Type “false” or remove the parameter to have no separator line.
command=	Type the path to the executable to be launched when the toolbar item is clicked.
params=	If a Web browser such as Internet Explorer is entered in the “command=” parameter, type the required URL for the browser to launch. If the “command=” parameter does not launch a browser, do not include this attribute.
description	Between <description> and </description> type a description for the application being launched. This text appears as a pop-up tool tip when the user places the cursor over the toolbar item.

5. Save and close the file.
6. Start the NOC and verify that the new toolbar button appears in the toolbar.
7. Verify that you can launch the application by clicking on the toolbar button.
8. Repeat this procedure on each computer where the NOC is installed, or copy the file to each computer.

5.3 View plugin details

Use the following procedure to see information about the plugins that are installed with the NOC.

1. On the **Help** menu, click **Plug-In Details**.

   The Plug-In Details window opens.

2. On the left side of the window, click on the sub-system whose details you want to view.

   The details of the plugin are displayed on the right side of the window.

3. Click **OK** to close the window.

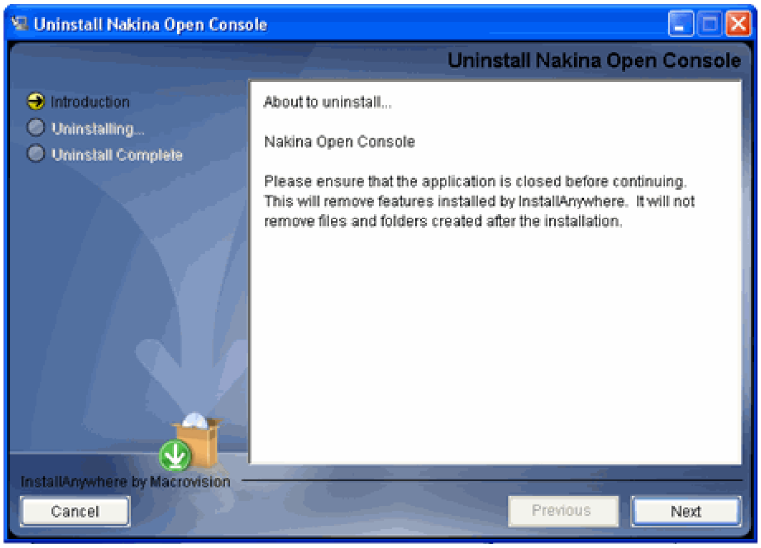
5.4 Uninstall the NOC software

Use this procedure to uninstall the NOC software by using the NOC uninstaller. (Alternatively, for Windows, you can uninstall NOC by using the standard Windows Add or Remove Programs control panel.)

4. Launch the uninstaller for your platform, as follows:

   - for Windows, click the **Start** menu, choose **All Programs** > **Nakina** > **Nakina Open Console** > **Uninstall** NI-Director Operations Console.
   - for Solaris, open the **Uninstall** NI-Director Operations Console directory and launch the uninstaller shell called **Uninstall**_NI-Director Operations Console**.**

   The system displays the **Uninstall** NI-Director Operations Console **window**.



Uninstall Nakina Open Console
Uninstall Nakina Open Console
Introduction
Uninstalling...
Uninstall Complete
InstallAnywhere by Macrovision
Cancel
Previous
Next

About to uninstall...

Nakina Open Console

Please ensure that the application is closed before continuing. This will remove features installed by InstallAnywhere. It will not remove files and folders created after the installation.

5. Click **Next**.

   The uninstaller displays the Uninstall Options screen.

6. Choose one of the following options and click **Next**.

   - **Complete Uninstall** — uninstalls the entire **NOC** application
   - **Uninstall Specific Features** — allows you to choose the features to uninstall

7. If you chose the **Uninstall Specific Features** option, do the following:
   a. On the **Choose Product Features** screen, clear the check boxes for any features that you do not want to uninstall.

      By default, all features are selected and the Uninstall button is grayed out. If you want to uninstall all features, return to the Uninstall Options screen and choose the Complete Uninstall option.

   b. Click **Uninstall**.

8. On the **Uninstall Complete** screen, click **Done**.

   The wizard closes.

5.5 Configure online help for Solaris

To view the online help in the Solaris version of the NI-Director Operations Console, you must set the path to the installed browser. (This procedure is not required for a Windows installation).

In a command window on the Solaris machine where the NOC is installed, use either of the following commands to set the path to the resident web browser:

- setenv PATH $PATH:<path to web browser>
- export PATH=$PATH:<path to web browser>

  where: <path to web browser> is the actual path to the web browser installed on the machine.

Appendix A: Sample TL1 command file

The following is an example of a TL1 command file. To see how this XML file will look in the NOC user interface, see “Sample TL1 commands as they appear in the NOC user interface” on page 40.

This example contains an indexlocked=”true” attribute, which indicates that the parameters that are specified within the block are positioned in the locations specified by their index values separated by commas in the TL1 command. Missing indexes will have no data. With no indexlocked attribute, the parameters specified within the block will be positioned one after the other in the order of their indexes in the TL1 command. No blanks are left for undefined indexes.

For example, consider a command defined with 6 parameters and indexed as follows (the last parameter is indexed as 7 instead of 5 and there are no parameter indexes for 5 and 6):

```
  <command>RTRV-PM-OC48</command>
<payload-1 indexlocked="true">
<parameter index="0"> .... </parameter>
<parameter index="1"> .... </parameter>
<parameter index="2"> .... </parameter>
<parameter index="3"> .... </parameter>
<parameter index="4"> .... </parameter>
<parameter index="7"> .... </parameter>
</payload-1>
```

- With the indexlocked attribute set, even though nothing is defined for indexes 5 and 6, their positions in the command are kept, but left blank.
- Without the indexlocked attribute set, no spots are held for non-defined indexes. The parameters come one right after another and the index number is ignored.

Table 3–2 on page 17 describes the XML elements for the TL1 command file in detail.

```
<!--

     tl1 command blocks will follow the following:

<command>:TID:<aid>:CTAG:<general>:<common>:<specific>:<state>:<block-
8>:<block-9>:<block-10>:<block-11>:<block-12>

        block index and associated block names
        0  = command
        1  = TID
        2  = aid
        3  = CTAG
        4  = general
        5  = common
        6  = specific
        7  = state
        8  = block-8
```

```
           ...
        12 = block-12

        Note:  TID and CTAG values will be filled in automatically.

     tl1command attributes:
        description The TL1 command description that is shown in the pulldown
        type    The entitytype that the TL1 command is valid for, must
correspond to inventory entitytype
        confirm Whether confirmation is required prior to sending the command

     AID parameters:
        Entity type= %ENTITYTYPE
        Slot #= %SLOT
        Port #= %PORT
        Timeslot= %TIMESLOT


     Blocks that allow optional parameters:
        parameter
           index= the placement of the param within the block
        optiontype
           static  = static parameter that is not shown in the options menu
           pulldown = pulldown menu (requires at least 1 or more option)
           text = text input field
        optiondescription = title of the parameter option (or static value if
optiontype=static)
        optionprepend = some optional parameters require TL1 syntax values
             (ie. MSGTYPE=, or ALMTYPE=)
        option= an optional value for the parameter

-->

<TL1Commands>
   <table-response>
        <!-- Alarm response format
        <TID><DATE><TIME>
        <AID>,<AIDTYPE>:<NTFCNCDE>,<CONDTYPE>,<SRVEFF>,<OCRDAT>,
        <OCRTM>,<LOCN>,<DIRN>:<CONDDESCR>
         -->
   <table name="RTRV-ALM-ALL">

     <!-- Column 1 -->
    <column
        block="0"
        index="0"
        name="AID"
        description="AID"
        visible="true" />
   <!-- Column 2 -->
   <column
        block="0"
        index="1"
        name="AIDTYPE"
     description="AIDtype"
     visible="true" />
<!-- Column 3 -->
<column
     block="1"
     index="0"
     name="NTFCNCDE"
     description="Severity condition"
     visible="true" />
<!-- Column 4 -->
<column
     block="1"
     index="1"
     name="CONDTYPE"
     description="Condition type"
     visible="true" />
<!-- Column 5 -->
<column
     block="1"
     index="2"
     name="SRVEFF"
     description="Service affecting"
     visible="true" />
<!-- Column 6 -->
<column
     block="1"
     index="3"
     name="OCRDAT"
     description="Occurrence date"
     visible="true" />
<!-- Column 7 -->
<column
     block="1"
     index="4"
     name="OCRTM"
     description="Occurrence time"
     visible="true" />
<!-- Column 8 -->
<column
     block="1"
     index="5"
     name="LOCN"
     description="Location near end or far end"
     visible="true" />
<!-- Column 9 -->
<column
     block="1"
     index="6"
     name="DIRN"
     description="Direction incoming or outgoing"
     visible="true" />
<!-- Column 10 -->
<column
     block="2"
     index="0"
     name="CONDDESCR"
     description="Condition"
     visible="true" />
  </table>
</table-response>

<table-event>
</table-event>

 <ne-level>
     <tl1command description="Retrieve NE Alarm All" confirm="false">
         <command>RTRV-ALM-ALL</command>
         <payload-2>
             <parameter index="0">
                 <optiontype>pulldown</optiontype>
                 <optiondescription>Alarm Type</optiondescription>
                 <optionprepend>ALRMSTAT=</optionprepend>
                 <option>ENABLED</option>
                 <option>DISABLED</option>
                 <option>BOTH</option>
            </parameter>
         </payload-2>
         <response-table name="RTRV-ALM-ALL" />
     </tl1command>

     <tl1command description="Retrieve OC48 PMs" confirm="false">
        <command>RTRV-PM-OC48</command>
        <aid>ALL</aid>
        <payload-1 indexlocked="true">
        <parameter index="0">
           <optiontype>static</optiontype>
           <optiondescription>ALL</optiondescription>
        </parameter>
        <parameter index="1">
           <optiontype>static</optiontype>
           <optiondescription>0-UP</optiondescription>
        </parameter>
        <parameter index="2">
           <optiontype>pulldown</optiontype>
           <optiondescription>Location</optiondescription>
           <option>NEND</option>
           <option>FEND</option>
        </parameter>
        <parameter index="3">
           <optiontype>static</optiontype>
           <optiondescription>ALL</optiondescription>
        </parameter>
        <parameter index="4">
           <optiontype>pulldown</optiontype>
           <optiondescription>Time Period</optiondescription>
           <option>15-MIN</option>
           <option>1-DAY</option>
        </parameter>
```

```
           <parameter index="7">
              <optiontype>pulldown</optiontype>
              <optiondescription>Index</optiondescription>
              <option>0</option>
              <option>1</option>
              <option>2</option>
              <option>3</option>
              <option>4</option>
              <option>5</option>
              <option>6</option>
              <option>7</option>
              <option>8</option>
              <option>9</option>
              <option>10</option>
              <option>11</option>
              <option>12</option>
              <option>13</option>
              <option>14</option>
              <option>15</option>
              <option>16</option>
              <option>17</option>
              <option>18</option>
              <option>19</option>
              <option>20</option>
              <option>21</option>
              <option>22</option>
              <option>23</option>
              <option>24</option>
              <option>25</option>
              <option>26</option>
              <option>27</option>
              <option>28</option>
              <option>29</option>
              <option>30</option>
              <option>31</option>
              <option>32</option>
           </parameter>
        </payload-1>
     </tl1command>
     <tl1command description="Retrieve NE Condition All" confirm="false">
        <command>RTRV-COND-ALL</command>
        <payload-2>
           <parameter index="0">
              <optiontype>pulldown</optiontype>
              <optiondescription>Alarm Type</optiondescription>
              <optionprepend>ALRMSTAT=</optionprepend>
              <option>ENABLED</option>
              <option>DISABLED</option>
              <option>BOTH</option>
           </parameter>
        </payload-2>
     </tl1command>
     <tl1command description="Retrieve NE Alarm/Condition All"
confirm="false">
```

```
         <command>
            <parameter index="0">
               <optiontype>pulldown</optiontype>
               <optiondescription>Command Type</optiondescription>
               <optionprepend>RTRV-</optionprepend>
               <option>ALM-ALL</option>
               <option>COND-ALL</option>
             </parameter>
         </command>
         <payload-2>
            <parameter index="0">
               <optiontype>pulldown</optiontype>
               <optiondescription>Alarm Type</optiondescription>
               <optionprepend>ALRMSTAT=</optionprepend>
               <option>ENABLED</option>
               <option>DISABLED</option>
               <option>BOTH</option>
            </parameter>
         </payload-2>
      </tl1command>
      <tl1command description="Retrieve NE Autonomous Output" confirm="false">
         <command>RTRV-AO</command>
         <payload-2 indexlocked="true">
            <parameter index="1">
               <optiontype>pulldown</optiontype>
               <optiondescription>Message Type</optiondescription>
               <optionprepend>MSGTYPE=</optionprepend>
               <option>ALM</option>
               <option>EVT</option>
               <option>ALM-EVT</option>
            </parameter>
         </payload-2>
      </tl1command>
      <tl1command description="Retrieve NE Default Security" confirm="false">
         <command>RTRV-SECU-DFLT</command>
      </tl1command>
      <tl1command description="Retrieve NE Active Users" confirm="false">
         <command>RTRV-ACTIVE-USER</command>
      </tl1command>
      <tl1command description="Retrieve NE Type" confirm="false">
         <command>RTRV-NETYPE</command>
      </tl1command>
      <tl1command description="Retrieve NE Inventory All" confirm="false">
         <command>RTRV-INVENTORY</command>
         <aid>ALL</aid>
      </tl1command>
      <tl1command description="Enter OC12 1+1 Linear Protection" type="OC12"
confirm="true">
         <command>ENT-FFP-OC12</command>
         <aid>
            <parameter index="0">
               <optiontype>pulldown</optiontype>
               <optiondescription>Working AID</optiondescription>
               <optionprepend>OC12-</optionprepend>
             <option>3-1</option>
             <option>5-1</option>
             <option>7-1</option>
             <option>9-1</option>
             <option>11-1</option>
           </parameter>
           <parameter index="1">
             <optiontype>pulldown</optiontype>
             <optiondescription>Protection AID</optiondescription>
             <optionprepend>OC12-</optionprepend>
             <option>4-1</option>
             <option>6-1</option>
             <option>8-1</option>
             <option>10-1</option>
             <option>12-1</option>
           </parameter>
           </aid>
        <payload-2>
           <parameter index="0">
             <optiontype>pulldown</optiontype>
             <optiondescription>Direction</optiondescription>
             <optionprepend>PSDIRN=</optionprepend>
             <option>UNI</option>
             <option>BI</option>
            </parameter>
         </payload-2>
      </tl1command>
   </ne-level>
   <card-level>
     <tl1command description="Retrieve Card Performance Monitoring - EQPT"
type="common" confirm="false">
        <command>RTRV-PM-ALL</command>
        <aid>ALL</aid>
     </tl1command>
     <tl1command description="Retrieve Card Performance Monitoring - Optical
Carrier" type="OC3" confirm="false">
        <command>RTRV-PM-OC3</command>
        <aid>OC3-%SLOT-ALL</aid>
     </tl1command>
     <tl1command description="Retrieve Card Performance Monitoring - Optical
Carrier" type="OC12" confirm="false">
        <command>RTRV-PM-OC12</command>
        <aid>OC12-%SLOT-ALL</aid>
     </tl1command>
     <tl1command description="Retrieve Card Performance Monitoring - T1"
type="DS1" confirm="false">
        <command>RTRV-PM-T1</command>
        <aid>DS1-%SLOT-ALL</aid>
     </tl1command>
     <tl1command description="Retrieve Card Performance Monitoring - T3"
type="DS3" confirm="false">
        <command>RTRV-PM-T3</command>
        <aid>DS3-%SLOT-ALL</aid>
     </tl1command>
     <tl1command description="Retrieve Card Performance Monitoring"
type="common" confirm="true">
        <command>
           <parameter index="0">
             <optiontype>pulldown</optiontype>
             <optiondescription>Command</optiondescription>
             <optionprepend>RTRV-PM-</optionprepend>
             <option>OC3</option>
             <option>OC12</option>
             <option>T1</option>
             <option>T3</option>
           </parameter>
        </command>
        <aid>
           <parameter index="0">
             <optiontype>pulldown</optiontype>
             <optiondescription>AID</optiondescription>
             <optionprepend />
             <option>OC3-%SLOT-ALL</option>
             <option>OC12-%SLOT-ALL</option>
             <option>DS1-%SLOT-ALL</option>
             <option>DS3-%SLOT-ALL</option>
           </parameter>
        </aid>
     </tl1command>
     <tl1command description="Enter OC12 1+1 Linear Protection" type="OC12"
confirm="true">
        <command>ENT-FFP-OC12</command>
        <aid>OC12-%SLOT-1
           <parameter index="1">
             <optiontype>pulldown</optiontype>
             <optiondescription>Protection AID</optiondescription>
             <optionprepend>OC12-</optionprepend>
             <option>4-1</option>
             <option>6-1</option>
             <option>8-1</option>
             <option>10-1</option>
             <option>12-1</option>
           </parameter>
        </aid>
        <payload-2>
           <parameter index="0">
             <optiontype>pulldown</optiontype>
             <optiondescription>Direction</optiondescription>
             <optionprepend>PSDIRN=</optionprepend>
             <option>UNI</option>
             <option>BI</option>
           </parameter>
        </payload-2>
     </tl1command>
   </card-level>
   <port-level>
     <tl1command description="Retrieve Card Performance Monitoring - EQPT"
type="common" confirm="false">
```

```
        <command>RTRV-PM-ALL</command>
        <aid>ALL</aid>
     </tl1command>
     <tl1command description="Retrieve Card Performance Monitoring - Optical
Carrier" type="OC3" confirm="false">
        <command>RTRV-PM-OC3</command>
        <aid>OC3-%SLOT-%PORT</aid>
     </tl1command>
     <tl1command description="Retrieve Card Performance Monitoring - Optical
Carrier" type="OC12Port" confirm="false">
        <command>RTRV-PM-OC12</command>
        <aid>OC12-%SLOT-%PORT</aid>
     </tl1command>
     <tl1command description="Retrieve Shelf Performance Monitoring - T1"
type="DS1" confirm="false">
        <command>RTRV-PM-T1</command>
        <aid>DS1-%SLOT-%PORT</aid>
     </tl1command>
     <tl1command description="Retrieve Shelf Performance Monitoring - T3"
type="DS3" confirm="false">
        <command>RTRV-PM-T3</command>
        <aid>DS3-%SLOT-%PORT</aid>
     </tl1command>
     <tl1command description="Enter OC3 1+1 Linear Protection" type="OC3"
confirm="true">
        <command>ENT-FFP-OC3</command>
        <aid>OC3-%SLOT-%PORT
           <parameter index="1">
              <optiontype>pulldown</optiontype>
              <optiondescription>Protection AID</optiondescription>
              <optionprepend>OC3-</optionprepend>
              <option>4-1</option>
              <option>6-1</option>
              <option>8-1</option>
              <option>10-1</option>
              </parameter>
              </aid>
        <payload-2>
           <parameter index="0">
              <optiontype>pulldown</optiontype>
              <optiondescription>Direction</optiondescription>
              <optionprepend>PSDIRN=</optionprepend>
              <option>UNI</option>
              <option>BI</option>
           </parameter>
        </payload-2>
     </tl1command>
     <tl1command description="Enter OC3x4 1+1 Linear Protection" type="OC3"
confirm="true">
        <command>ENT-FFP-OC3</command>
        <aid>OC3-%SLOT-%PORT
           <parameter index="1">
              <optiontype>pulldown</optiontype>
              <optiondescription>Protection AID</optiondescription>
```

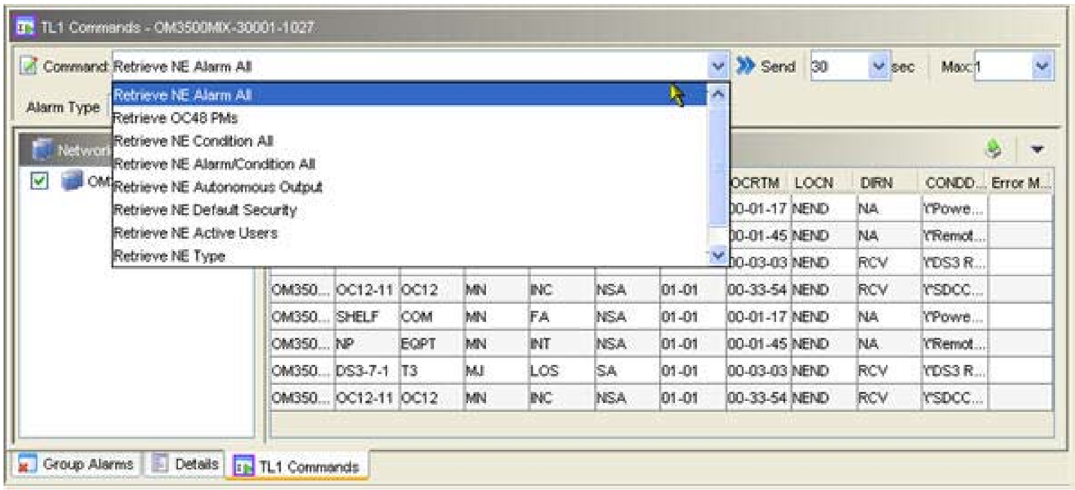
```
                  <optionprepend>OC3-</optionprepend>
                  <option>4-1</option>
                  <option>4-2</option>
                  <option>4-3</option>
                  <option>4-4</option>
                  <option>6-1</option>
                  <option>6-2</option>
                  <option>6-3</option>
                  <option>6-4</option>
                  <option>8-1</option>
                  <option>8-2</option>
                  <option>8-3</option>
                  <option>8-4</option>
                  <option>10-1</option>
                  <option>10-2</option>
                  <option>10-3</option>
                  <option>10-4</option>
               </parameter>
            </aid>
            <payload-2>
               <parameter index="0">
                  <optiontype>pulldown</optiontype>
                  <optiondescription>Direction</optiondescription>
                  <optionprepend>PSDIRN=</optionprepend>
                  <option>UNI</option>
                  <option>BI</option>
               </parameter>
            </payload-2>
         </tl1command>
      </port-level>

</TL1Commands>
```

A.1 Sample TL1 commands as they appear in the NOC user interface

The following examples show what the NOC user interface will display for the sample TL1 command file.

When Command and Control is invoked in the context of a selected NE, then the ne-level commands defined for the corresponding NE vendor and model will be made available to the user. The tl1command in the ne-level block of the following example will produce the example shown in this screen (the screen includes the response table for the “Retrieve NE Alarm All” command).



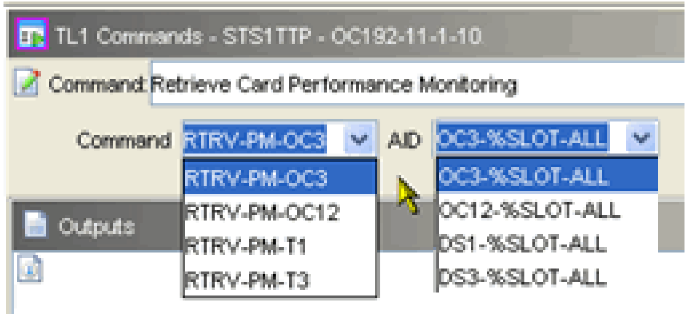
TL1 Commands - OM3500MIX-30001-1027

Command: Retrieve NE Alarm All

Send 30 sec Max: 1

							OCRTM	LOCN	DIRN	CONDD...	Error M...
							00-01-17	NEND	NA	YPowe...	
							00-01-45	NEND	NA	YRemot...	
							00-03-03	NEND	RCV	YDS3 R...	
OM350...	OC12-11	OC12	MN	INC	NSA	01-01	00-33-54	NEND	RCV	YSDCC...	
OM350...	SHELF	COM	MN	FA	NSA	01-01	00-01-17	NEND	NA	YPowe...	
OM350...	NP	EQPT	MN	INT	NSA	01-01	00-01-45	NEND	NA	YRemot...	
OM350...	DS3-7-1	T3	MJ	LOS	SA	01-01	00-03-03	NEND	RCV	YDS3 R...	
OM350...	OC12-11	OC12	MN	INC	NSA	01-01	00-33-54	NEND	RCV	YSDCC...	

When Command and Control is invoked with the context of a selected circuit pack, then the card-level commands defined for the corresponding circuit pack type will be made available to the user. The tl1command in the **card-level** block of the following example will produce the example shown in this screen.

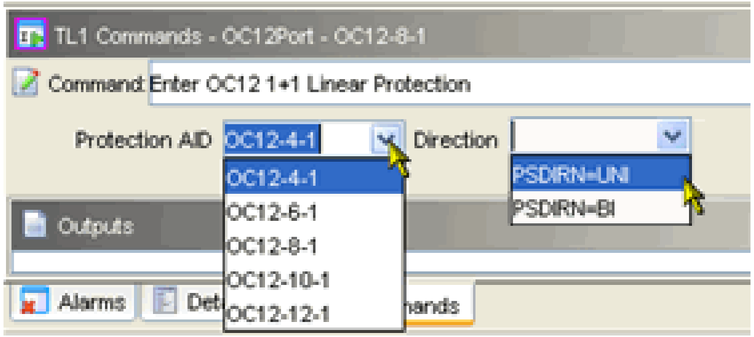


TL1 Commands - STS1TTP - OC192-11-1-10

Command: Retrieve Card Performance Monitoring

Command RTRV-PM-OC3 AID OC3-%SLOT-ALL

When Command and Control is invoked with the context of a selected Port, then the port-level commands defined for the corresponding Port type will be made available to the user. The tl1command in the **port-level** block will produce the example shown in this screen.



TL1 Commands - OC12Port - OC12-8-1

Command: Enter OC12 1+1 Linear Protection

Protection AID OC12-4-1 Direction