

IBM® Tivoli® Netcool/OMNIbus Probe for
Alcatel-Lucent Wavestar SNMS (CORBA)
7.0

Reference Guide
March 02, 2012



Notice

Before using this information and the product it supports, read the information in [Appendix A, “Notices and Trademarks,” on page 15.](#)

Edition notice

This edition (SC23-7800-01) applies to version 7.0 of IBM Tivoli Netcool/OMNIbus Probe for Alcatel-Lucent Wavestar SNMS (CORBA) and to all subsequent releases and modifications until otherwise indicated in new editions.

This edition replaces SC23-7800-00.

© **Copyright International Business Machines Corporation 2010, 2012.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

About this guide.....	v
Document control page.....	v
Conventions used in this guide.....	v
 Chapter 1. Probe for Alcatel-Lucent Wavestar SNMS (CORBA).....	1
Summary.....	1
Installing probes.....	2
Firewall considerations.....	3
Data acquisition.....	3
Connecting to the CORBA interface.....	3
Resynchronization.....	4
Timeout.....	4
Peer-to-peer failover functionality.....	4
Properties and command line options.....	5
Elements.....	7
Error messages.....	9
ProbeWatch messages.....	12
Running the probe.....	13
 Appendix A. Notices and Trademarks.....	15
Notices.....	15
Trademarks.....	16

About this guide

The following sections contain important information about using this guide.

Document control page

Use this information to track changes between versions of this guide.

The IBM Tivoli Netcool/OMNIBus Probe for Alcatel-Lucent Wavestar SNMS (CORBA) documentation is provided in softcopy format only. To obtain the most recent version, visit the IBM® Tivoli® Information Center:

<https://www.ibm.com/support/knowledgecenter/SSSHTQ/omnibus/probes/common/Probes.html>

Table 1. Document modification history		
Document version	Publication date	Comments
SC23-7800-00	October 29, 2010	First IBM publication.
SC23-7800-01	March 02, 2012	<p>Information about operating system conventions was added in “Conventions used in this guide” on page v.</p> <p>Requirements were updated in “Summary” on page 1.</p> <p>Installation information was updated in “Installing probes” on page 2.</p> <p>The following properties were added or updated in “Properties and command line options” on page 5:</p> <ul style="list-style-type: none">• Interface• ORBCharEncoding• ORBLocalPort• ORBWCharDefault <p>ProbeWatch messages were updated in “ProbeWatch messages” on page 12.</p> <p>Information about running the probe was added in “Running the probe” on page 13.</p>

Conventions used in this guide

All probe guides use standard conventions for operating system-dependent environment variables and directory paths.

Operating system-dependent variables and paths

All probe guides use standard conventions for specifying environment variables and describing directory paths, depending on what operating systems the probe is supported on.

For probes supported on UNIX and Linux operating systems, probe guides use the standard UNIX conventions such as **\$variable** for environment variables and forward slashes (/) in directory paths. For example:

\$OMNIHOME/probes

For probes supported only on Windows operating systems, probe guides use the standard Windows conventions such as **%variable%** for environment variables and backward slashes (\) in directory paths. For example:

%OMNIHOME%\probes

For probes supported on UNIX, Linux, and Windows operating systems, probe guides use the standard UNIX conventions for specifying environment variables and describing directory paths. When using the Windows command line with these probes, replace the UNIX conventions used in the guide with Windows conventions. If you are using the bash shell on a Windows system, you can use the UNIX conventions.

Note : The names of environment variables are not always the same in Windows and UNIX environments. For example, %TEMP% in Windows environments is equivalent to \$TMPDIR in UNIX and Linux environments. Where such variables are described in the guide, both the UNIX and Windows conventions will be used.

Operating system-specific directory names

Where Tivoli Netcool/OMNIBus files are identified as located within an *arch* directory under NCHOME or OMNIHOME, *arch* is a variable that represents your operating system directory. For example:

\$OMNIHOME/probes/*arch*

The following table lists the directory names used for each operating system.

Note : This probe may not support all of the operating systems specified in the table.

Table 2. Directory names for the arch variable	
Operating system	Directory name represented by arch
AIX® systems	aix5
Red Hat Linux® and SUSE systems	linux2x86
Linux for System z	linux2s390
Solaris systems	solaris2
Windows systems	win32

OMNIHOME location

Probes and older versions of Tivoli Netcool/OMNIBus use the OMNIHOME environment variable in many configuration files. Set the value of OMNIHOME as follows:

- On UNIX and Linux, set \$OMNIHOME to \$NCHOME/omnibus.
- On Windows, set %OMNIHOME% to %NCHOME%\omnibus.

Chapter 1. Probe for Alcatel-Lucent Wavestar SNMS (CORBA)

Alcatel-Lucent Wavestar SNMS is an element management system (EMS) for the WaveStar product family. It provides Element Management Layer (EML) functions, including configuration, fault, performance, and security management, for these products.

The Probe for Alcatel-Lucent Wavestar SNMS (CORBA) acquires data from Alcatel-Lucent Wavestar SNMS using the G7 Common Object Request Broker Architecture (CORBA) interface.

This guide contains the following sections:

- [“Summary” on page 1](#)
- [“Installing probes” on page 2](#)
- [“Firewall considerations” on page 3](#)
- [“Data acquisition” on page 3](#)
- [“Properties and command line options” on page 5](#)
- [“Elements” on page 7](#)
- [“Error messages” on page 9](#)
- [“ProbeWatch messages” on page 12](#)
- [“Running the probe” on page 13](#)

Summary

Each probe works in a different way to acquire event data from its source, and therefore has specific features, default values, and changeable properties. Use this summary information to learn about this probe.

The following table provides a summary of the Probe for Alcatel-Lucent Wavestar SNMS (CORBA).

<i>Table 3. Summary</i>	
Probe target	Alcatel-Lucent Wavestar SNMS version 10
Probe executable name	nco_p_lucent_snms
Probe installation package	omnibus-arch-probe-nco-p-lucent-snms-version
Package version	7.0
Probe supported on	For details of supported operating systems, see the following Release Notice on the IBM Software Support website: https://www-304.ibm.com/support/docview.wss?uid=swg21579736
Properties file	\$OMNIBHOME/probes/arch/lucent_snms.props
Rules file	\$OMNIBHOME/probes/arch/lucent_snms.rules

<i>Table 3. Summary (continued)</i>	
Requirements	A currently supported version of IBM Tivoli Netcool/OMNIBus. Nonnative Probe Server: probe-nonnative-base-14 IBM Object Request Broker (IBM ORB, supplied with Netcool/OMNIBus) Java Runtime Environment (JRE) 1.5
Connection method	CORBA
Remote connectivity	The probe can connect to a remote device using a CORBA interface.
Multicultural support	Available
Peer-to-peer failover functionality	Available
IP environment	IPv4 and IPv6
Federal Information Processing Standards (FIPS)	IBM Tivoli Netcool/OMNIBus uses the FIPS 140-2 approved cryptographic provider: IBM Crypto for C (ICC) certificate 384 for cryptography. This certificate is listed on the NIST website at http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/1401val2004.htm . For details about configuring Netcool/OMNIBus for FIPS 140-2 mode, see the <i>IBM Tivoli Netcool/OMNIBus Installation and Deployment Guide</i> .

Installing probes

All probes are installed in a similar way. The process involves downloading the appropriate installation package for your operating system, installing the appropriate files for the version of Netcool/OMNIBus that you are running, and configuring the probe to suit your environment.

The installation process consists of the following steps:

1. Downloading the installation package for the probe from the Passport Advantage Online website.

Each probe has a single installation package for each operating system supported. For details about how to locate and download the installation package for your operating system, visit the following page on the IBM Tivoli Knowledge Center:

http://www-01.ibm.com/support/knowledgecenter/SSSHTQ/omnibus/probes/all_probes/wip/reference/install_download_intro.html

2. Installing the probe using the installation package.

The installation package contains the appropriate files for all supported versions of Netcool/OMNIBus. For details about how to install the probe to run with your version of Netcool/OMNIBus, visit the following page on the IBM Tivoli Knowledge Center:

http://www-01.ibm.com/support/knowledgecenter/SSSHTQ/omnibus/probes/all_probes/wip/reference/install_install_intro.html

3. Configuring the probe.

This guide contains details of the essential configuration required to run this probe. It combines topics that are common to all probes and topics that are peculiar to this probe. For details about additional configuration that is common to all probes, see the *IBM Tivoli Netcool/OMNIBus Probe and Gateway Guide*.

Firewall considerations

When using CORBA probes in conjunction with a firewall, the firewall must be configured so that the probe can connect to the target system.

Most CORBA probes can act as both a server (listening for connections from the target system) and a client (connecting to the port on the target system to which the system writes events). If you are using the probe in conjunction with a firewall, you must add the appropriate firewall rules to enable this dual behavior.

There are three possible firewall protection scenarios, for which you must determine port numbers before adding firewall rules:

1. If the host on which the probe is running is behind a firewall, you must determine what remote host and port number the probe will connect to.
2. If the host on which the target system is running is behind a firewall, you must determine the incoming port on which the probe will listen and to which the target system will connect.
3. If each host is secured with its own firewall, you must determine the following four ports:
 - a. The outgoing port (or port range) for the probe.
 - b. The hostname and port of the target system.
 - c. The outgoing port on which the target system sends events if the probe is running as a client.
 - d. The incoming port on which the probe listens for incoming events.

Note : Most, but not all, CORBA probes listen on the port specified by the **ORBLocalPort** property. The default value for this property is 0, which means that an available port is selected at random. If the probe is behind a firewall, the value of the **ORBLocalPort** property must be specified as a fixed port number.

CORBA probes that use EventManager or NotificationManager objects may use different hosts and ports from those that use NamingService and EntryPoint objects. If the probe is configured to get object references from a NamingService or EntryPoint object, you must obtain the host and port information from the system administrator of the target system. When you have this information, you can add the appropriate firewall rules.

Data acquisition

Each probe uses a different method to acquire data. Which method the probe uses depends on the target system from which it receives data.

The Probe for Alcatel-Lucent Wavestar SNMS (CORBA) gathers events from Alcatel-Lucent Wavestar SNMS using the G7 CORBA interface.

Data acquisition is described in the following topics:

- [“Connecting to the CORBA interface” on page 3](#)
- [“Resynchronization” on page 4](#)
- [“Timeout” on page 4](#)
- [“Peer-to-peer failover functionality” on page 4](#)

Connecting to the CORBA interface

The probe acquires data using either a Naming Service or Interoperable Object Reference (IOR) files.

Alcatel-Lucent Wavestar SNMS stores event data in the following objects:

- `NetworkR1` - this object contains details of all active alarms.
- `Observable` - this object contains details of all new alarms as they occur.

Using a Naming Service

The probe can acquire the details of the Naming Service using either the IBM Object Request Broker (ORB) or the Naming Context IOR file. To use the IBM ORB, specify host and port values for the **ORBInitialHost** and **ORBInitialPort** properties. To use the Naming Context IOR file, specify its location using the **NamingContextIOR** property.

Whichever method you use to acquire the Naming Service details, you must also use the **NetworkR1NS** and **ObservableNS** properties to specify the locations of the NetworkR1 object and the Observable object in the Naming Service.

Using IOR Files

To acquire event data using IOR files, you must use the **NetworkR1IOR** and **ObservableIOR** properties to specify the locations of the NetworkR1 and Observable IOR files.

Resynchronization

If the **Resynch** property is set to true, the probe requests all active alarms from Alcatel-Lucent Wavestar SNMS before connecting to the Notification Service to get new alarms.

When management of a network element is added to Alcatel-Lucent Wavestar SNMS, alarms received from that network element are not automatically forwarded to the probe. To ensure that the probe receives all such alarms, specify a value of true for the **Resynch** property and a value of 1 or greater for the **ResynchInterval** property.

The **ResynchInterval** property controls the interval (in seconds) at which the probe repeats the resynchronization process. Specifying a value of 0 causes the probe to perform one resynchronization at startup.

Timeout

The probe has a timeout facility that allows it to disconnect from the system if there has been no activity for a predefined amount of time.

You can use the **Timeout** property to specify how long the probe waits before disconnecting. The default is 0, which instructs the probe to maintain the connection indefinitely.

Peer-to-peer failover functionality

The probe supports failover configurations where two probes run simultaneously. One probe acts as the master probe, sending events to the ObjectServer; the other acts as the slave probe on standby. If the master probe fails, the slave probe activates.

While the slave probe receives heartbeats from the master probe, it does not forward events to the ObjectServer. If the master probe shuts down, the slave probe stops receiving heartbeats from the master and any events it receives thereafter are forwarded to the ObjectServer on behalf of the master probe. When the master probe is running again, the slave probe continues to receive events, but no longer sends them to the ObjectServer.

Example property file settings for peer-to-peer failover

You set the peer-to-peer failover mode in the properties files of the master and slave probes. The settings differ for a master probe and slave probe.

Note : In the examples, make sure to use the full path for the property value. In other words replace \$OMNIHOME with the full path. For example: /opt/IBM/tivoli/netcool.

The following example shows the peer-to-peer settings from the properties file of a master probe:

```
Server      : "NCOMS"
RulesFile   : "master_rules_file"
```

```

MessageLog : "master_log_file"
PeerHost   : "slave_hostname"
PeerPort   : 6789 # [communication port between master and slave probe]
Mode       : "master"
PidFile    : "master_pid_file"

```

The following example shows the peer-to-peer settings from the properties file of the corresponding slave probe:

```

Server      : "NCOMS"
RulesFile   : "slave_rules_file"
MessageLog  : "slave_log_file"
PeerHost    : "master_hostname"
PeerPort    : 6789 # [communication port between master and slave probe]
Mode        : "slave"
PidFile     : "slave_pid_file"

```

Properties and command line options

You use properties to specify how the probe interacts with the device. You can override the default values by using the properties file or the command line options.

The following table describes the properties and command line options specific to this probe. For more information about generic Netcool/OMNIbus properties and command line options, see the *IBM Tivoli Netcool/OMNIbus Probe and Gateway Guide*.

Table 4. Properties and command line options		
Property name	Command line option	Description
Inactivity <i>integer</i>	<i>-inactivity integer</i>	Use this property to specify the number of seconds that the probe waits to receive events before sending a ProbeWatch inactivity message. The default is 60.
Interface <i>string</i>	<i>-interface string</i>	Use this property to specify the local host used by the server-side ORB to place the server's host name into the IOR of a remote object. The default is 127.0.0.1.
NamingContextIOR <i>string</i>	<i>-nsior string</i>	Use this property to specify the location of the IOR file that contains the root context of the Naming Service. The default is "".
NetworkR1IOR <i>string</i>	<i>-emsior string</i>	Use this property to specify the location of the NetworkR1 IOR file. The probe uses this object to retrieve all resynchronization alarms on start up. The default is "".

Table 4. Properties and command line options (continued)

Property name	Command line option	Description
NetworkR1NS <i>string</i>	<code>-emsns string</code>	<p>Use this property to specify the location of the NetworkR1 object in the Naming Service.</p> <p>The default is S2\\\.1\\\.0-Version.Lucent/SNMS_EML-NetworkR1.</p> <p>Note : Back slashes (\) are treated as escape characters, so double back slashes (\\) are used to create a readable path.</p>
ObservableIOR <i>string</i>	<code>-observableior string</code>	<p>Use this property to specify the location of the Observable object IOR file. The probe uses this object to retrieve all new alarms as they occur.</p> <p>The default is " " .</p>
ObservableNS <i>string</i>	<code>-observablens string</code>	<p>Use this property to specify the location of the Observable object in the Naming Service.</p> <p>The default is S2\\\.1\\\.0-Version.Lucent/SNMS_EML-Observable.</p> <p>Note : Back slashes (\) are treated as escape characters, so double back slashes (\\) are used to create a readable path.</p>
ORBCharEncoding <i>string</i>	<code>-orbcharencoding string</code>	<p>Use this property to specify the native character encoding set used by the Object Request Broker (ORB) for character data.</p> <p>The default is UTF8.</p>
ORBInitialHost <i>string</i>	<code>-orbinitialhost string</code>	<p>Use this property to specify the name of the Naming Service host.</p> <p>The default is snmshost.</p>
ORBInitialPort <i>integer</i>	<code>-orbinitialport integer</code>	<p>Use this property to specify the port number through which the probe connects to the Naming Service.</p> <p>The default is 1570.</p>
ORBLocalPort <i>integer</i>	<code>-orblocalport integer</code>	<p>Use this property to specify the local port to which the ORB listens.</p> <p>The default is 0 (which instructs the ORB to select an available port at random).</p>

Table 4. Properties and command line options (continued)

Property name	Command line option	Description
ORBCharDefault <i>string</i>	<code>-orbwchardefault <i>string</i></code>	Use this property to specify what wide character (wchar) set the IBM ORB uses when communicating with other ORBs that do not publish a wchar set. The default is UTF16.
Resynch <i>string</i>	<code>-resynch <i>string</i></code>	Use this property to specify whether the probe requests all active alarms from Alcatel-Lucent Wavestar SNMS before connecting to the Notification Service for new alarms. This property takes the following values: <code>false</code> : the probe does not perform resynchronization. <code>true</code> : the probe performs resynchronization. The default is <code>false</code> .
ResynchInterval <i>integer</i>	<code>-resynchinterval <i>integer</i></code>	Use this property to specify the interval (in seconds) at which the probe repeats the resynchronization process. The default is 0 (which instructs the probe to resynchronize once). Note : To instruct the probe to perform repeated resynchronization, you must set the Resynch property to <code>true</code> and set the ResynchInterval property to a value greater than or equal to 1.
Timeout <i>integer</i>	<code>-timeout <i>integer</i></code>	Use this property to specify the number of seconds that the probe waits to receive events before disconnecting from the Notification Service and shutting down. The default is 300. Note : If you specify a value of 0 for this property, the probe waits indefinitely.

Elements

The probe breaks event data down into tokens and parses them into elements. Elements are used to assign values to ObjectServer fields; the field values contain the event details in a form that the ObjectServer understands.

The following table describes the elements that the probe generates. Not all the elements described are generated for each event. The elements that the probe generates depend on the event type.

<i>Table 5. Elements</i>	
Element name	Element description
\$additionalInfo	This element displays additional information about the alarm.
\$additionalText	This element displays a brief description of the problem being reported by the alarm.
\$CTP	This element displays the Connection Termination Point.
\$domain_name	This element displays the Domain name from which the event was sent.
\$edgePoint	This element indicates whether the alarm relates to a PTP that is not the end point of an inner topological link.
\$emsTime	This element displays the time at which the event was reported by the EMS system.
\$Equipment	This element displays the layer to which the alarm applies.
\$EquipmentHolder	This element displays the name of the object reporting the alarm as given in the EMS user interface.
\$event_name	This element displays the probable cause as given in the EMS user interface.
\$event_type	This element displays the time at which the error occurred in the network element.
\$isClearable	Indicates whether the event is clearable.
\$layer	This element displays the layer to which this alarm is relevant.
\$ManageElement	This element indicates the object that managed the system for which the alarm was generated.
\$nativeEMSName	This element displays the name used by the EMS interface for the object reporting an alarm.
\$neTime	This element displays the time of the alarm provided by the network element.
\$NativeProbableCause	This element displays the probable cause for the alarm as defined by the EMS interface.
\$notificationId	This element displays a unique identifier of the alarm. This is derived from the serial number of the alarm as used by the EMS.

Table 5. Elements (continued)	
Element name	Element description
\$objectType	This element displays the type of object reporting the alarm.
\$perceivedSeverity	This element displays the perceived severity of the alarm.
\$probableCause	This element displays the probable cause of the alarm.
\$probableCauseQualifier	This element displays the qualifier used to classify the alarm type.
\$ProtectionGroup	This element indicates whether the alarm has affected the service.
\$PTP	This element displays the Physical Termination Point.
\$RoutingLink	This element indicates the routing link of the equipment.
\$serviceAffecting	This element indicates whether the alarm has affected the notification service.
\$Subnetwork	This element displays the subnetwork where the equipment is located.
\$Subnetworkconnection	This element displays the connection point of the equipment on the subnetwork.
\$TopologicalLink	This element indicates whether the alarm relates to a connection between PTPs.
\$TPPool	This element displays a list of TPs affected by an equipment failure.
\$TrafficDescriptor	This element displays a Descriptor of the network traffic.

Error messages

Error messages provide information about problems that occur while running the probe. You can use the information that they contain to resolve such problems.

The following table describes the error messages specific to this probe. For information about generic Netcool/OMNIbus error messages, see the *IBM Tivoli Netcool/OMNIbus Probe and Gateway Guide*.

Table 6. Error messages		
Error	Description	Action
ResynchInterval will not take -ve values	A negative value was specified for the ResynchInterval property.	Make the value of the ResynchInterval property greater than or equal to 0.

Table 6. Error messages (continued)

Error	Description	Action
CorbaUtils: Cannot get object reference, IOR and NS values were empty	The probe cannot get the required object references. No values have been specified for the IOR file or the Naming Service.	Depending on whether you are using an IOR file or a Naming Service to acquire event data, you must specify values for the required properties as described in “Connecting to the CORBA interface” on page 3.
CorbaUtils: Failed to find file, please check file exists	The probe could not find an IOR file.	Depending on whether you are using an IOR file or a Naming Service to acquire event data, you must specify values for the required properties as described in “Connecting to the CORBA interface” on page 3.
CorbaUtils: CORBA.BAD_PARAM Exception. The IOR is not the expected Object. Please check IOR files are valid and up to date	The IOR files are not pointing to the correct CORBA objects.	Verify that you have specified the correct values for the NetworkR1IOR and ObservableIOR properties, and that the IOR files are valid and up-to-date.
CorbaUtils: CORBA.OBJECT_NOT_EXIST Exception. The CORBA service is not running on the expected Host/Port/IOR	The CORBA service is not running on the expected host, port, or IOR.	Verify that you have specified the correct values for the NetworkR1NS , ObservableNS properties.
CorbaUtils: Exception raised when locating server	The probe cannot find a Naming Service object.	Verify that you have specified the correct values for the NetworkR1IOR and ObservableIOR properties, and that the IOR files are valid and up-to-date.
CorbaUtils: Check Naming Service and CORBA interface is running	There is a problem with the Naming Service or the CORBA interface.	Verify that the Naming Service and the CORBA interface are running.
CorbaUtils: Unexpected Exception occurred	An unexpected problem occurred with the CORBA interface.	Contact IBM Software Support.

Table 6. Error messages (continued)

Error	Description	Action
CorbaUtils: Failed to get the object reference	The probe cannot get the required object references.	Depending on whether you are using an IOR file or a Naming Service to acquire event data, check the values that you specified for the required properties described in “Connecting to the CORBA interface” on page 3.
Observer: Failed to get object reference for Observable object	The probe cannot get the object reference for the Observable CORBA object.	Depending on whether you are using an IOR file or a Naming Service to acquire event data, check the values that you specified for the required properties described in “Connecting to the CORBA interface” on page 3.
Observer: Failed to narrow Observable reference	The Observable CORBA object could not be narrowed to the correct type.	Contact IBM Software Support.
Observer: Check that IOR files are correct and up to date	The IOR files are not pointing to the correct Observable CORBA object.	Verify that you have specified the correct value for the ObservableIOR property, and that the IOR file is valid and up-to-date.
Servant appears to already be active when activating consumer	A consumer object is already registered on the ORB.	Contact IBM Software Support.
Observer: Failed to connect to the CORBA interface due to exception	The gateway failed to connect to the CORBA interface due to an exception.	Check the probe message log file <code>lucent_snms.log</code> for more information. The log file is located in the following directory: \$OMNIHOME/log If you cannot identify the problem, contact IBM Software Support.
Observer: Please check that the interface is running and the probe is using up to date IOR files	There is a problem with the CORBA interface or the IOR file for the Observable CORBA object.	Verify that the CORBA interface is running. Verify that you have specified the correct value for the ObservableIOR property, and that the IOR file is valid and up-to-date.
Parser: Cannot parse type <i>value_type: name</i>	The probe cannot parse the event value described in the error message.	Contact IBM Software Support.

Table 6. Error messages (continued)

Error	Description	Action
Parser: Error parsing <i>name</i>	The probe cannot parse the event value described in the error message.	Contact IBM Software Support.
Resynch: Failed to narrow networkR1 reference	The NetworkR1 CORBA object could not be narrowed to the correct type.	Contact IBM Software Support.
Resynch: Failed to connect to the CORBA Interface due to exception	Resynchronization failed due to a CORBA interface exception.	Verify that the CORBA interface is running correctly.
Resynch: Please check that the interface is running and the probe is using up to date IOR files	Resynchronization failed due to a problem with the CORBA interface or the IOR files.	Verify that the CORBA interface is running correctly. Depending on whether you are using an IOR file or a Naming Service to acquire event data, check the values that you specified for the required properties described in “Connecting to the CORBA interface” on page 3.
Resynch: Failed to get interface version/name information	Resynchronization failed because the probe could not get the required interface information.	Verify that the CORBA interface is running correctly.
Resynch: Check that the CORBA service is running	Resynchronization failed due to a problem with the CORBA interface.	Verify that the CORBA interface is running correctly.

ProbeWatch messages

During normal operations, the probe generates ProbeWatch messages and sends them to the ObjectServer. These messages tell the ObjectServer how the probe is running.

The following table describes the ProbeWatch messages that the probe generates. For information about generic Netcool/OMNIbus ProbeWatch messages, see the *IBM Tivoli Netcool/OMNIbus Probe and Gateway Guide*.

Note : The content of some ProbeWatch messages generated by the probe has changed in this release. This might require you to make changes to custom tools or automations that depend on ProbeWatch messages.

Table 7. ProbeWatch messages

ProbeWatch message	Description	Triggers/causes
No alarms received for inactivity period (<i>inactivity_period</i> seconds)	The probe received no alarms for the period of time indicated by the <i>inactivity_period</i> .	The probe has received no alarms for the period of time specified by the Inactivity property.
No alarms received for timeout period (<i>timeout_period</i> seconds)	The probe received no alarms for the period of time indicated by the <i>timeout_period</i> .	The probe has received no alarms for the period of time specified by the Timeout property.
Observer client started	The probe is connecting to the server.	This message is sent when the probe initially connects to the server.
Observer client finished	The probe is disconnecting from the server.	This message is sent when the probe is disconnecting from the server, usually on shut down.
Resynch client started	The probe started a resynchronization.	The probe started a resynchronization.
Resynch complete	The probe completed a resynchronization.	The probe completed a resynchronization.

Running the probe

You run the probe from the command line.

To start the probe, use the following command:

```
$OMNIHOME/probes/nco_p_lucent_snms
```

Appendix A. Notices and Trademarks

This appendix contains the following sections:

- Notices
- Trademarks

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing 2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who want to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Software Interoperability Coordinator, Department 49XA

3605 Highway 52 N
Rochester, MN 55901
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

IBM, the IBM logo, ibm.com, AIX, Tivoli, zSeries, and Netcool are trademarks of International Business Machines Corporation in the United States, other countries, or both.

Adobe, Acrobat, Portable Document Format (PDF), PostScript, and all Adobe-based trademarks are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, other countries, or both.

Intel, Intel Inside (logos), MMX, and Pentium are trademarks of Intel Corporation in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java™ and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.



SC23-7800-01

