Tivoli® Netcool/OMNIbus Probe for Ericsson OSS-RC

IBM

SC23-7664-02

Version 7.0.2

Reference Guide
October 31, 2009

Tivoli® Netcool/OMNIbus Probe for Ericsson OSS-RC

IBM

SC23-7664-02

Version 7.0.2

Reference Guide
October 31, 2009

Edition notice

This edition applies to version 7.0.2 of IBM Tivoli Netcool/OMNIbus Probe for Ericsson OSS-RC (SC23-7664-02) and to all subsequent releases and modifications until otherwise indicated in new editions.

This edition replaces SC23-7664-01.

Contents

iii

Document control page

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

The IBM Tivoli Netcool/OMNIbus Probe for Ericsson OSS-RC documentation is provided in softcopy format only. To obtain the most recent version, visit the IBM Tivoli Netcool Information Center:

http://publib.boulder.ibm.com/infocenter/tivihelp/v8r1/index.jsp?topic=/
com.ibm.tivoli.nam.doc/welcome_ptsm.htm

Table 1. Document modification history

Document version	Publication date	Comments
SC23-6023-00	March 16, 2007	First IBM publication.
SC23-7664-00	June 20, 2007	Patch version and release updated. Resynch property added.
SC23-7664-01	December 14, 2007	Patch version and release updated. Error messages added.
SC23-7664-02	October 31, 2009	"Summary" on page 1 updated. Section added to discuss the process of starting the probe. See "Starting the probe" on page 5. Section added to discuss the properties that need to be configured to enable the probe to receive events. See "Configuring the probe for specific Ericsson devices" on page 5. New properties added to the Properties and command line options table. See "Properties and command line options" on page 10. Description for **FlushBufferInterval** added.

IBM Tivoli Netcool/OMNIbus Probe for Ericsson OSS-RC

The Ericsson Operation and Support System Radio and Core (OSS-RC) is an element manager for the Ericsson 3G core network. The Probe for Ericsson OSS-RC connects to Ericsson OSS-RC versions 2.x, 3.x, 4.x, RANOS, and CN-OSS using the standard CORBA interface.

This probe supports version 3.2.0.

The following topics describe the probe and how it works:

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 summarizes the probe.

Table 2. Summary

Probe target	Ericsson OSS-RC version 2.x, 3.x, 4.x, RANOS, and CNOSS
Probe executable file name	`nco_p_ericsson_oss_rc`
Patch number	7.0
Probe supported on	Solaris, HP-UX PA-RISC For details of the operating system versions on which this probe is supported, see the following page on the IBM Tivoli Netcool Information Center: http://publib.boulder.ibm.com/infocenter/tivihelp/ v8r1/index.jsp?topic=/com.ibm.netcool_OMNIbus.doc/ Supported_Platforms.htm
Properties file	`$OMNIHOME/probes/`*arch*`/ericsson_oss_rc.props`
Rules file	`$OMNIHOME/probes/`*arch*`/ericsson_oss_rc.rules`

Table 2. Summary (continued)

Requirements	IBM Tivoli Netcool/OMNIbus V7.1, 7.2, 7.2.1, or 7.3.
	Note: If you are running IBM Tivoli Netcool/OMNIbus V7.1, you must have Interim Fix 8 (7.1.0.11-IF0008) installed. If you are running IBM Tivoli Netcool/OMNIbus V7.2, you must have Interim Fix 4 (7.2.0.7-IF0004) installed. If you are running IBM Tivoli Netcool/OMNIbus V7.2.1, you must have Interim Fix 3 (7.2.1.5-IF0003) installed. If you are running IBM Tivoli Netcool/OMNIbus V7.3, you need no additional fix packs.
	`Java JRE 1.5 runtime libraries`
	`probe-nonnative-base-7`
	`probe-visibroker-support`
	Note: These patches, with the exception of the VisiBroker patch, are available from the Passport Advantage® site:
	http://www-306.ibm.com/software/howtobuy/ passportadvantage/pao_customers.htm
	To obtain the Visibroker patch, you must contact IBM Software Support.
Connection method	CORBA
Remote connectivity	Yes
Licensing	Electronic licensing was deprecated with the release of IBM Tivoli Netcool V7.2. All IBM Tivoli Netcool V7.2 (and later) products use the IBM software licensing process.
Internationalization	Not available
Peer-to-peer failover functionality	Available
IP environment	IPv4 and IPv6
	The probe is supported on IPv6 when running on IBM Tivoli Netcool/OMNIbus V7.0, 7.1, 7.2, 7.2.1, and 7.3 on Solaris, and on IBM Tivoli Netcool/OMNIbus V7.2, 7.2.1, and 7.3 on all other UNIX plaftorms.
Federal Information Protocol Standards (FIPS)	IBM Tivoli Netcool/OMNIbus V7.2.1 and 7.3 use the FIPS 140-2 approved cryptographic provider: IBM Crypto for C (ICC) certificate 384 for cryptography. This certificate is listed on the NIST web site at http://csrc.nist.gov/cryptval/140-1/1401val2004.htm For details about configuring Netcool/OMNIbus for FIPS 140-2 mode, see *IBM Tivoli Netcool/OMNIbus Installation and Deployment Guide* (SC23-6370).

Installing the probe on Tivoli Netcool/OMNIbus V7.1.0, 7.2.0, and 7.2.1

The process of installing probes on Tivoli Netcool/OMNIbus V7.1.0, 7.2.0, and 7.2.1 consists of downloading the appropriate installation package, and installing each of the patches that the package contains.

Installing the probe on UNIX and Linux operating systems

To install the probe and each of its required patches on UNIX and Linux operating systems, use the following steps:

1. Make a backup of any existing configuration files that you might already have for an earlier version of the probe and that you want to retain.

2. Download the UNIX or Linux installation package for the probe from the Passport Advantage Online Web site:

   http://www-306.ibm.com/software/howtobuy/passportadvantage/
   pao_customers.htm

3. Extract the contents of the package to a temporary location.

4. Navigate to the `patches` directory under the directory containing the `README.txt` file in the extracted package.

   This directory contains the main patches for the probe.

5. Consult the `README.txt` file to ascertain whether you will need to download any further patches from the Passport Advantage Online Web site.

6. Install each of the patches in the `patches` directory, and any additional required patches, in the order shown in the `README.txt` file by running the following command:

   ```
   $NCHOME/omnibus/install/nco_patch -install patch_path
   ```

Where *patch_path* is the path of the required patch.

Note: At any stage, you can see which patches are already installed by running the following command:

```
$NCHOME/omnibus/install/nco_patch -print=id
```

Installing the probe on Tivoli Netcool/OMNIbus V7.3.0 or later

With the introduction of Tivoli Netcool/OMNIbus V7.3.0, all probes are installed using the Tivoli Netcool/OMNIbus installer. You can install the probe using the installation wizard, using a text-based installer (console mode), or using settings predefined in a text file (silent mode).

Installing the probe on UNIX and Linux operating systems

To install the probe on UNIX and Linux operating systems, use the following steps:

1. Make a backup of any existing configuration files that you might already have for an earlier version of the probe and that you want to retain.

2. Download the UNIX or Linux installation package for the probe from the Passport Advantage Online Web site:

   http://www-306.ibm.com/software/howtobuy/passportadvantage/
   pao_customers.htm

3. Extract the contents of the package to a temporary location. The package contains the probe itself and all non-third party dependencies.

4. If you want to install the probe using the installation wizard, use the following steps:

   a. Run the following command:

      ```
      $NCHOME/omnibus/install/nco_install_integration
      ```

b. When the wizard runs, specify the location of the probe to be installed. This location is the directory containing the `README.txt` file in the extracted package.

c. Accept the license conditions.

If you want to install the probe in console mode, use the following steps:

a. Run the following command:

```
$NCHOME/omnibus/install/nco_install_integration -i console
```

b. When the text-based installer runs, specify the location of the probe to be installed. This location is the directory containing the `README.txt` file in the extracted package.

c. Accept the license conditions.

If you want to install the probe in silent mode, use the following steps:

a. Create a text file named `reponse.txt` that contains the following settings:

```
PROBE_OR_GATE_LOCATION=README_directorypath
LICENSE_ACCEPTED=true
```

Where *README_directorypath* is the path to the directory containing the `README.txt` file in the extracted package.

b. Run the following command:

```
$NCHOME/omnibus/install/nco_install_integration -i silent -f
response_path/response.txt
```

Where *response_path* is the full path to the `response.txt` file.

In each case, the probe is installed in the following directory:

```
$NCHOME/omnibus/probes
```

Basic configuration

To configure the probe prior to running, you must update the rules file using probe-specific information.

Updating the rules file

The probe is supplied with a lookup table (`Corba_3gpp_V320.lookup`). This file is installed in the following location: `$OMNIHOME/probes/includes/`

This file is referenced in the rules file by the following command:

```
include "../includes/Corba_3gpp_V320.lookup"
```

Note: `$OMNIHOME` cannot be used in the paths to the lookup files. You must enter the full path to the IBM Tivoli Netcool/OMNIbus installation directory.

Data acquisition

The probe connects to Ericsson OSS-RC through a Common Object Request Broker Architecture CORBA (3GPP) interface. 3GPP is an Object Management Group specification that provides a standard interface definition between objects in a distributed environment; that is, it allows applications to communicate with one another regardless of where they are located or who has designed them.

The probe complies with the following 3GPP standards:

- TS 32.300 v4.1.1 Name Convention for Managed Objects (Release 4)

- TS 32.111-2 v3.2.0 Information Service (Release 99)
- TS 32.111-3 v3.2.0 Corba Solution Set (Release 99)

The probe then connects to the NotificationIRPOperation server and uses the CORBA notification push model to receive new alarms from the server as they are generated.

The probe checks the status of the IRP agent every 60 seconds. You can change this frequency if required using the **Agentheartbeat** property.

The following topics describe how the probe acquires data:
- "Starting the probe"
- "Configuring the probe for specific Ericsson devices"
- "Device connections through the CORBA interface" on page 6
- "Filters for notifications and alarms" on page 7
- "Retrieving objects" on page 7
- "Status checking" on page 7
- "Command line interface" on page 8
- "Peer-to-peer failover functionality" on page 9

Starting the probe

On startup, the probe initializes an ORB and connects to the Ericsson Element Manager's Alarm IRP (OSS-RC, RANOS and CN-OSS) and Event IRP objects. The probe then resynchronizes with the Ericsson Element Manager and acquires the alarms/stateless events currently stored in the Ericsson Element Manager.

The probe then processes the acquired alarms/stateless events, setting most attributes as tokens, and generates an AckAlarmID token. These tokens are sent to the ObjectServer as events. Once the process is complete, the probe subscribes to the online events, processes them, and then forwards them to the ObjectServer.

Configuring the probe for specific Ericsson devices

The Probe for Ericsson OSS-RC connects to the following Ericsson devices:
- Ericsson OSS-RC versions 2.x, 3.x, 4.x.
- Ericsson OSS-RC RANOS.
- Ericsson OSS-RC CN-OSS.

To enable the probe to receive events from the Ericsson device, you must specify the following property values:

For Ericsson OSS-RC version 2.x

Set the **Alarmirp** property to com.ericsson.nms.fm.ALARM_IRP_CIRPAgent1 to specify the appropriate alarm IRP object reference.

Set the **Notificationirp** property to com.ericsson.nms.fm.cif.service.NAConsumer to specify the appropriate notification IRP object.

For Ericsson OSS-RC version 3.x and 4.x

Set the **Alarmirp** property to com.ericsson.nms.fm_cirpagent.AlarmIRP to specify the appropriate alarm IRP object reference.

Set the **Notificationirp** property to com.ericsson.nms.cif.service.NMSNAConsumer to specify the appropriate notification IRP object.

To locate the Naming Service, the probe can use either the **NameServiceHost** and **NSPort** properties to identify the host name and port number of the Naming Service, or the probe can use the IOR file specified by the **NSIorfile** property.

Therefore, the property settings depend on which method the probe will use to locate the Naming Service. Either set the **NameServiceHost** property to *<oss-rc_host>* where the OSS-RC is running and set the **NSPort** property to 49254.

Or set the **NSIorfile** property to http://*<oss-rc_host>* where the OSS-RC is running:80/ior/ExternalNameService.ior

For Ericsson RANOS

Set the **Alarmirp** property to com.ericsson.nms.umts.ranos.AlarmService_R1 to specify the appropriate alarm IRP object reference.

Set the **Notificationirp** property to com.ericsson.umts.ranos.NotificationService_R1 to specify the appropriate notification IRP object.

For Ericsson CN-OSS

Set the **Alarmirp** property to com.ericsson.nms.fm.ALARM_IRP_CIRPAgent1 to specify the appropriate alarm IRP object reference.

Set the **Notificationirp** property to com.ericsson.nms.fm.cif.service.NMSNAConsumer to specify the appropriate notification IRP object.

Device connections through the CORBA interface

The probe uses the CORBA interface to retrieve alerts from Ericsson OSS-RC. The probe can use one of two methods to connect to the device: Interoperable Object Reference (IOR) files or the Naming Service.

IOR files

If using IOR files, the probe retrieves the object reference of the AlarmIRPOperation from the IOR file that is specified by the **AlarmirpFile** property; the probe retrieves the object reference of the NotificationIRPOperation from the IOR file that is specified by the **NotificationirpFile** property. The AlarmIRPOperation and NotificationIRPOperation servers form a part of the IRP agent. The **Eventirp** and **EventirpFile** properties can be used to retrieve stateless events from the Ericsson OSS-RC system. Set the **Eventirp** to property to com.ericsson.nms.fm_cirpagent.EventIRP to specify the Event IRP Object Reference. Set the **EventirpFile** property to specify the path to the Event IRP Object Reference file.

Naming Service

If the **AlarmirpFile** and **NotificationirpFile** properties are not specified, the probe retrieves the object references of the AlarmIRPOperation and NotificationIRPOperation servers from the Naming Service. To locate the Naming

Service, the probe either uses the **NameServiceHost** and **NSPort** properties to identify the host name and port number of the Naming Service, or uses the IOR file specified by the **NSIorfile** property.

The Naming Service uses the values that are specified by the **Alarmirp** and **Notificationirp** properties to retrieve the object references to the IRP objects.

Retrieving objects

The probe initially receives a list of all active alarms from the AlarmIRPOperation server. The probe connects to the Event IRP object of the Ericsson Element Manager and retrieves the stateless events. The probe then connects to the NotificationIRPOperation server and uses the CORBA notification push model to receive new alarms from the server as they are generated.

Status checking

The probe checks the status of the IRP agent every 60 seconds. You can change this frequency if required using the **Agentheartbeat** property.

Filters for notifications and alarms

The **NotificationFilter** and **AlarmFilter** properties allow you to specify what notifications and alarms are sent to the probe. When you use this properties, you must use the actual token names.

For example, the token *h* represents the element NV_PERCEIVED_SEVERITY. So, to specify that the probe is sent only notifications with a perceived severity of 3, you must set the **NotificationFilter** property to $h = = 3.

You can specify more complex filters using AND and OR statements. For example, to specify that the probe is sent notifications with a perceived severity of 3 or 4, you must set the **NotificationFilter** property to $h = = 3 or $h = = 4.

To specify that the probe is only sent notifications for a specific managed element, set the **NotificationFilter** property to Managed_Node_Name~$f where $f represents the element NV_MANAGED_OBJECT_INSTANCE and Managed_Node_Name is the name of the managed object.

For example, if the set of alarms that you require return an NV_MANAGED_OBJECT_INSTANCE of SubNetwork=ONRM_RootMo, SubNetwork=SNMP, ManagedElement=SP1, set the **NotificationFilter** property to SP1'~$'f.

Note: The tilde character (~) is required because spaces cannot be entered in this property. For string comparisons, the first argument is considered to be contained in the second argument; which is why $f is listed second to the literal.

The following table displays the token mappings for use with the **AlarmFilter** and **NotificationFilter** properties.

Table 3. Token mappings

Element	Token
NV_NOTIFICATION_ID	a
NV_CORRELATED_NOTIFICATIONS	b
NV_EVENT_TIME	c

Table 3. Token mappings (continued)

Element	Token
NV_SYSTEM_DN	d
NV_MANAGED_OBJECT_CLASS	e
NV_MANAGED_OBJECT_INSTANCE	f
NV_PROBABLE_CAUSE	g
NV_PERCEIVED_SEVERITY	h
NV_SPECIFIC_PROBLEM	i
NV_ADDITIONAL_TEXT	j
NV_ALARM_ID	k
NV_ACK_USER_ID	l
NV_ACK_TIME	m
NV_ACK_SYSTEM_ID	n
NV_ACK_STATE	o
NV_BACKED_UP_STATUS	p
NV_BACK_UP_OBJECT	q
NV_THRESHOLD_INFO	r
NV_TREND_INDICATION	s
NV_STATE_CHANGE_DEFINITION	t
NV_MONITORED_ATTRIBUTES	u
NV_PROPOSED_REPAIR_ACTIONS	v
NV_REASON	w

Command line interface

The probe has a Command Line Interface (CLI) that you can use to perform commands, for example, to acknowledge alarms or to request a full resynchronization of the CORBA interface.

To use the CLI, you must use the CommandPort property in the properties file to specify a port through which commands are sent. When you want to perform commands, Telnet to this port. The following table describes the commands that you can use with the CLI.

Table 4. CLI commands

Command	Description
acknowledge_alarm	Use this command to acknowledge an alarm in the 3GPP interface. **Note:** This command takes as a parameter the NV_ALARM_ID of the alarm being acknowledged. Only one alarm can be acknowledged at a time. This command also uses the values specified by the **AckSystemId** and **AckUserId** properties in the properties file.
get_event_list	Use this command to display a list of events from Event IRP.
get_event_list_filter	Use this command to perform a get_event_list using this filter.

Table 4. CLI commands (continued)

Command	Description
help	Use this command to display online help about the CLI.
resynch_all	Use this command to perform a full resynchonization with the 3GPP interface.
resynch_filter	Use this command to perform partial resynchronization with the 3GPP interface. **Note:** This command takes as a parameter a filter in the same format as the **AlarmFilter** property.
unacknowledge_alarm	Use this command to unacknowledge an alarm in the 3GPP interface.
userid_acknowledge_alarm	Use this command to acknowledge an alarm in the 3GPP interface by specifying the NV_ALARM_ID of the alarm being acknowledged and the NV_ACK_USER_ID . The format of the alarm is: *ID userID*.
userid_unacknowledge_alarm	Use this command to unacknowledge an alarm in the 3GPP interface by specifying the NV_ALARM_ID of the alarm being acknowledged and the NV_ACK_USER_ID.
version	Use this command to display the version of the probe.

Note: Because the CLI is based upon Telnet connections, you can connect to the probe from anywhere. This means that simple scripts can be set up to allow users to acknowledge selected events from the event list by creating desktop tools to Telnet to the probe, send a command, and then close the connection.

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 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 is running again, the slave 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.

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   : 5555 # [communication port between master and slave probes]
Mode       : "master"
```

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   : 5555 # [communication port between master and slave probes]
Mode       : "slave"
```

Note: The properties file also contains all other properties required to configure the probe.

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 properties and command line options, see the *IBM Tivoli Netcool/OMNIbus Probe and Gateway Guide*, (SC23-6373).

Table 5. Properties and command line options

Property name	Command line option	Description
AckSystemId *string*	-acksystemid *string*	Use this property to specify the processing system on which the IRP Manager runs. This is used by the acknowledge_alarm CLI function. The default is "".
AckUserId *string*	-ackuserid *string*	Use this property to specify the name of the user acknowledging the alarm. This is used by the acknowledge_alarm CLI function. The default is "".
AgentHeartbeat *integer*	-agentheartbeat *integer*	Use this property to specify the frequency (in seconds) with which the probe checks the status of the IRP agent. The default is 60.
AlarmFilter *string*	-alarmfilter *string*	Use this property to specify the filter the alarm IRP uses to limit the alarms sent to the probe. The default is "".

Table 5. Properties and command line options (continued)

Property name	Command line option	Description
Alarmirp *string*	-alarmirp *string*	Use this property to specify the alarm IRP object reference. The default is com.ericsson.nms.fm_cirpagent. AlarmIRP. **Note:** The value required for this property is version-specific. To confirm the appropriate value for the version of the device that you are running, you must contact your technical support team or your system administrator. If necessary, a tool can be used that queries the Naming Service. Such tools are available on the Internet.
AlarmirpFile *string*	-alarmirpfile *string*	Use this property to specify the path to the Alarm IRP object reference. The default is "".
CommandPort *integer*	-commandport *integer*	Use this property to specify the port to which users can Telnet to communicate with the 3GPP interface using the Command Line Interface (CLI) supplied with the probe. The default is 6790.
CommandPortLimit *integer*	-commandportlimit *integer*	Use this property to specify the maximum number of Telnet connections that can be made to the probe. The default is 10.
DiscardBlankAdd Text *string*	-discardblankadd text *string*	Use this property to specify whether the probe discards blank additional line elements. The default is true.
EventFilter *string*	-eventfilter *string*	Use this property to specify the filter used by the EventIrp. The default is "".
Eventirp *string*	-eventirp *string*	Use this property to specify the Event IRP Object Reference. The default is com.ericsson.nms.fm_cirpagent. EventIRP.
EventirpFile *string*	-eventirpfile *string*	Use this property to specify the full path to the Event IRP Ior file. The default is "".

Table 5. Properties and command line options (continued)

Property name	Command line option	Description
FlushBufferInterval *integer*	-flushbufferinterval *integer*	Use this property to specify how often (in seconds) the probe flushes all alerts in the buffer to the ObjectServer. The default is 0 (which instructs the probe to never flush the alerts to the ObjectServer).
GetEventList *string*	-geteventlist *string*	Use this property to specify whether events are received from Event IRP. The default is false: Events are not received from Event IRP.
GetEventListOnStart *string*	-geteventlistonstart *string*	Use this property to specify whether events are received from Event IRP during startup. The default is false: Events are not received from Event IRP during startup.
GetFMError Messages *string*	-getfmerror messages *string*	Use this property to specify whether the probe subscribes to Fault Management Error Messages. The default is false.
NameServiceHost *string*	-nameservicehost *string*	Use this property to specify the host on which the naming service is running. The default is "".
Notification Filter *string*	-notificationfilter *string*	Use this property to specify the filter the notification IRP uses to limit the notifications sent to the probe. The default is "".
Notificationirp *string*	-notificationirp *string*	Use this property to specify the Notification IRP object. The default is com.ericsson.nms.cif.service. NMSNAConsumer. **Note:** The value required for this property is version-specific. To confirm the appropriate value for the version of the device that you are running, you must contact your technical support team or your system administrator. If necessary, a tool can be used that queries the Naming Service. Such tools are available on the Internet.
Notificationirp File *string*	-notificationirpfile *string*	Use this property to specify the path of the notification IRP IOR file. The default is "".
NSIorfile *string*	-nsiorfile *string*	Use this property to specify the Naming Service object reference file. The default is "".

Table 5. Properties and command line options (continued)

Property name	Command line option	Description
NSPort *integer*	-nsport *integer*	Use this property to specify the port on which the Naming Service is running. The default is 0.
ORBLocalPort *integer*	-orblocalport *integer*	Use this property to specify the port number for the ORB to listen on. The default is 0.
Resynch *string*	-resynch *string*	Use this property to specify whether the probe performs a resynchronization at startup. The possible values are: false - probe does not perform a resynchronization true - probe performs a resynchronization The default is false.
Retry *string*	-retry *string*	Use this property to specify whether the probe attempts to reconnect to the system following a timeout. The possible values are: false: The probe does not attempt to reconnect to the system true: The probe attempts to reconnect to the system The default is false.
Timeout *integer*	-timeout *integer*	Use this property to specify the time (in seconds) that the probe allows the port to be silent before disconnecting. The default is 0 (probe never disconnects).
TimeTick *integer*	-timetick *integer*	Use this property to specify the time (in seconds) that Ericsson OSS-RC sessions are kept open. The default is 15.

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:

Table 6. Elements

Element name	Element description
ATTRIBUTE_VALUE(*element_name*)	This element specifies the value of an element that is being monitored. This element is generated dynamically and its content is dependent on the IRPAgent. **Note:** The *element_name* part of this element and the next element can be the name of any of the other elements in this table.
ATTRIBUTE_VALUE_CHANGE (*element_name*)	This element specifies the managed object attributes whose value changes are being monitored.
DOMAIN_NAME	This element specifies the domain name from which the notification originated.
EVENT_NAME	This element specifies the extended event type for this IRP.
EVENT_TYPE	This element specifies the event type of the notification.
NV_ACK_STATE	This element specifies the acknowledgement state of the alarm.
NV_ACK_SYSTEM_ID	This element specifies the system ID of the IRP Manager processing the notification.
NV_ACK_TIME	This element specifies the time at which the user acknowledged the alarm.
NV_ACK_USER_ID	This element specifies the last user who has changed the acknowledgement state.
NV_ADDITIONAL_TEXT	This element specifies information about the network element from which the alarm originated.
NV_ALARM_ID	This element specifies the identification information of the alarm as it appears in the alarm list.
NV_BACK_UP_OBJECT	This element specifies the distinguished Name (DN) of the backup object.
NV_BACKED_UP_STATUS	This element specifies whether the object has been backed up.
NV_CORRELATED_NOTIFICATIONS_ *notif_ID_Set*	This element specifies the set of notifications to which this notification is considered to be correlated. This element is generated dynamically and its content is dependent on the IRPAgent.
NV_CORRELATED_NOTIFICATIONS_ SOURCE	This element specifies the source of the notification set.
NV_EVENT_TIME	This element specifies the time at which the event occurred.
NV_MANAGED_OBJECT_INSTANCE	This element specifies the managed object instance of the network resource.
NV_NOTIFICATION_ID	This element specifies the identification information of the notification.

Table 6. Elements (continued)

Element name	Element description
NV_PERCEIVED_SEVERITY	This element specifies the relative level of urgency for operator attention.
NV_PROBABLE_CAUSE	This element specifies further information about the probable cause of the alarm.
NV_PROPOSED_REPAIR_ACTIONS	This element specifies the proposed repair actions associated with the notification.
NV_SPECIFIC_PROBLEM	This element specifies further information about the problem to which the notification relates.
NV_SYSTEM_DN	This element specifies the distinguished name (DN) used to identify the system.
NV_THRESHOLD_INFO	This element specifies information about a threshold that has been crossed.
NV_TREND_INDICATION	This element specifies how an observed condition has changed.
ResynchEvent	This element specifies whether the event was obtained by running the resynchronization function getalarm_list(). Possible values are: • false: The event was obtained after the probe subscribed to the notification service for real-time alarms • true: The event was obtained by running the resynchronization function

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 error messages, see the *IBM Tivoli Netcool/OMNIbus Probe and Gateway Guide*, (SC23-6373).

Table 7. Error messages

Error	Description	Action
Command Port Error occurred *Command Port* Failed to get CommandPortLimit property - using IOR *Command Port* Failed to get property CommandPort *Command Port* Failed to open listening socket, shutting down Thread! *Command Port* Failed to send probewatch message *Command Port* host Failed to close command socket *Command Port* host Failed to get socket IO *Command Port* host Failed to read command *Command Port* Thread shutting down due to error! NetcoolIRPManager: Failed to acknowledge_alarms() NetcoolIRPManager: Failed to Unacknowledge_alarms()	A problem occurred with the command port functionality.	Check that you have specified the command port correctly. Check the connection between the probe and the command port.
CLASSPATH ERROR,cannot find the VisiBroker/Netcool classes	The CLASSPATH environment variable is set incorrectly. This variable is set automatically during installation to include the paths in which the classes have been installed. This error message displays if the files are subsequently moved.	Update the CLASSPATH environment variable to include the correct location of the patches listed in Installation Requirements.
BAD_PARAM Exception i.e one or more of the in/out parameter is null InvalidParameter Exception NetcoolIRPManager: Exception occurred. Stack trace to stderr OperationNotSupported Exception Unexpected CORBA Exception	There is a problem with the command port functionality.	Refer to your CORBA documentation.

Table 7. Error messages (continued)

Error	Description	Action
COMMUNICATION FAILURE Exception i.e Server is dead GetSubscriptionStatus Exception NetcoolIRPManager:Both NameServiceHost and NSPort property needs to be set NetcoolIRPManager:'Alarmirp' property needs to be set NetcoolIRPManager: 'Notificationirp' property needs to be set NetcoolIRPManager: Failed to perform resynch	The probe cannot connect to the server.	Check that the Ericsson OSS-RC server is running correctly. Check that you have specified the parameters correctly in the properties file.
Error Failed to parse event completely Name is null, cannot create Element NetcoolIRPManager: Category Value NOT supported: Discarding Alarm NetcoolIRPManager: Error when parsing event NetcoolIRPManager: Failed to send event	The probe cannot parse the alarm; this is probably because the alarm is not in a format that the probe can understand.	Check that the Ericsson OSS-RC server is running correctly.
Failed to get properties	The probe is unable to open the properties file.	Check the properties file or the command line and ensure the parameter is pointing to the correct destination.
Failed to get timeout property value, defaulting to 0	This is message generated by the IRP Manager.	This message is intended for use when debugging.
NetcoolIRPManager: Stack Trace to stderr:	The **Timeout** property has not been set.	Check the value for the **Timeout** property; change this value if necessary.

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 raw ProbeWatch error messages that the probe generates. For information about generic error messages, see the *IBM Tivoli Netcool/OMNIbus Probe and Gateway Guide*, (SC23-6373).

Table 8. ProbeWatch messages

ProbeWatch message	Description	Triggers or causes
Will listen for commands on port number *command_port value*	The probe is listening for commands on the specified port.	The specified port is ready to receive commands.
[Command Port] Failed to send probewatch message	The probe was unable to send the alert, command, or ProbeWatch message specified to the ObjectServer.	The ObjectServer is not available, or there is a problem witht the connection with the ObjectServer.
Failed to listen for commands on port number *command_port value* : + e.toString()	The probe could not open the socket specified by the port to listen for commands.	The specified port is in use for another process.
START SYNCHRONIZATION	The probe is synchronizing the events.	The probe has started receiving alarms from the alarm list.
SYNCHRONIZATION ERROR - + e.toString()	The probe could not get new alarms.	The probe failed to get the alarm list, or failed to perform resynchronization of alarms.
END SYNCHRONIZATION	The probe is closing the synchronization process.	The probe has finished receiving alarms from the alarm list.

Appendix. 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 wish 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.

Other company, product, or service names may be trademarks or service marks of others.

IBM®

Printed in USA