

Netcool/OMNIbus Probe for Motorola OMC-R  
(iDEN)  
4.0.4159

*Reference Guide*  
*March 28, 2008*



**Note**

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 applies to version 4.0.4114 of IBM Tivoli Netcool/OMNIBus Probe for Motorola OMC-R (iDEN) (SC23-7885-01) and to all subsequent releases and modifications until otherwise indicated in new editions.

This edition replaces SC23-7885-00.

© **Copyright International Business Machines Corporation 2008.**

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

---

# Contents

**Document control page..... v**

**Chapter 1. Probe for Motorola OMC-R (iDEN).....1**

- Summary..... 1
- Data acquisition..... 2
  - Multi-headed probes..... 2
  - Chat in and chat out scripts.....2
  - Read Timeout.....3
  - Inactivity..... 3
  - Backoff strategy.....3
  - Hosts file format..... 3
  - Peer-to-peer failover functionality.....4
- Properties and command line options..... 5
  - Escape codes..... 8
- Elements..... 8
- Error messages..... 9
- ProbeWatch messages..... 11

**Appendix A. Notices and Trademarks..... 15**

- Notices..... 15
- Trademarks..... 16



## Document control page

---

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

The IBM Tivoli Netcool/OMNIBus Probe for Motorola OMC-R (iDEN) documentation is provided in softcopy format only. To obtain the most recent version, visit the IBM Tivoli Netcool Knowledge Center:

[http://publib.boulder.ibm.com/infocenter/tivihelp/v8r1/index.jsp?topic=/com.ibm.tivoli.namomnibus.doc/welcome\\_ptsm.htm](http://publib.boulder.ibm.com/infocenter/tivihelp/v8r1/index.jsp?topic=/com.ibm.tivoli.namomnibus.doc/welcome_ptsm.htm)

Table 1. Document modification history		
Document version	Publication date	Comment
SC23-7885-00	January 23, 2006	First IBM publication.
SC23-7885-01	March 28, 2008	Summary section updated.



# Chapter 1. Probe for Motorola OMC-R (iDEN)

Motorola Operations & Maintenance Center - Radio (OMC-R) is part of the Motorola NSS advanced Network Management Portfolio and serves as the interface point of the network to other management platforms. The suite is targeted at Wireless operators for management of GSM, analog, CDMA, switching and intelligent networks, with support for products such as iDEN.

The Probe for Motorola OMC-R (iDEN) acquires data from Motorola OMC-R.

Data can also be collected from Motorola OMC-R using a 3GPP CORBA interface. For details of the probe that acquires data in this way, see the *IBM Tivoli Netcool/OMNIBus Probe for Motorola OMC-R (3GPP) Reference Guide*, (SC23-6017).

This guide contains the following sections:

- “Summary” on page 1
- “Data acquisition” on page 2
- “Properties and command line options” on page 5
- “Elements” on page 8
- “Error messages” on page 9
- “ProbeWatch messages” on page 11

## 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	Motorola OMC-R
Probe executable file name	nco_p_motorolaomcr_iden
Patch number	4.0
Probe supported on	For details of supported operating systems, see the following Release Notice on the IBM Software Support Website: <a href="https://www-304.ibm.com/support/docview.wss?uid=swg21414855">https://www-304.ibm.com/support/docview.wss?uid=swg21414855</a>
Properties file	\$OMNIHOME/probes/arch/motorolaomcr_iden.props
Rules file	\$OMNIHOME/probes/arch/motorolaomcr_iden.rules
Requirements	A currently supported version of IBM Tivoli Netcool/OMNIBus. probe-compatibility-3.x (for Tivoli® Netcool/OMNIBus 3.6 only)
Connection method	Telnet

Table 2. Summary (continued)	
Remote connectivity	Available <b>Note :</b> The probe can connect to a remote device using the <b>Host</b> , <b>Port</b> , and <b>HostsFile</b> properties in the properties file.
Multicultural support	Not available
Peer-to-peer failover functionality	Available
IP environment	IPv4 and IPv6 <b>Note :</b> The probe only supports the IPv6 environment on IBM® Tivoli Netcool® OMNIBus v7.x running on Solaris.

## Data acquisition

The probe connects to the Multicasting Buffer System (MCB) using a TCP/IP connection and acquires alert data. The acquired alert data can be an OMC-R event or an MCB MUX event.

## Multi-headed probes

The Probe for Motorola OMC-R (iDEN) is a multi-headed probe that can connect to either a single device or to multiple devices. When the probe connects to a single device, it uses the connection properties. When the probe connects to multiple devices, it references a hosts file; this file contains details of the devices to which the probe connects and the connection properties for each connection.

To connect to a switch, the probe uses either the **Host** and **Port** properties in the probe properties file or a hosts file. If the **HostsFile** property has not been defined, then the probe uses the **Host** and **Port** properties to make a single connection. If, in addition to the **HostsFile** property, either of these properties is undefined, the probe exits.

## Chat in and chat out scripts

Chat in and chat out scripts control probe login and logout. These scripts are on a single line in the expect-send format (for chat in scripts) or send-expect format (for chat out scripts). You can specify chat in and chat out strings using the **ChatinString** and **ChatoutString** properties or the `-chatinstring` and `-chatoutstring` command line options.

The format is:

```
ChatinString : expect send expect send....
```

```
ChatoutString : send expect send expect....
```

**Note :** Each element in the chat strings is separated by white space. In order to send or expect a sequence that includes white space, surround the sequence with single quotes.

A typical chat in script might be:

```
.*login.*:.* anu\r\n .*assword.*:.* anu\r\n
```

The expect text can use any regular expression, while the send text can send any characters, including control characters using the standard UNIX/C escape sequences described in [“Escape codes”](#) on page 8.



## Read Timeout

The **ReadTimeout** property specifies how long the probe waits to read alarm data before timing out. Each time the probe attempts to read an alarm, this is the allotted time that it waits to receive data. If nothing is received, the probe moves on to the next alarm.

## Inactivity

The probe has a timeout facility that allows it to disconnect from the socket if it fails to receive the next alarm data within a predefined amount of time. To specify how long the probe waits before disconnecting, use the **Inactivity** property. After this length of time, the probe disconnects from the switch, sends a ProbeWatch message to the ObjectServer, and tries to reconnect.

You can also use the **InactivityRetry** property to specify the number of consecutive inactivity intervals that the probe allows before disconnecting. If this property is set to 0, the probe disconnects after a single period of inactivity.

## Backoff strategy

If the probe fails to establish a connection or loses an existing connection to the device, the probe reverts to a backoff strategy. The probe tries to reestablish a connection after one second, two seconds, then four seconds, eight seconds, and so on, up to a maximum of 4096 seconds.

After the connection is made to the specified port, the probe tries to log in to the device. If the probe fails to log in, it shuts down and tries to connect again. The backoff strategy remains in place until a successful login occurs. The user can also specify a reconnection interval using the **ReconnectionInterval** property or `-reconnectioninterval` command line option. When this property is enabled, the probe reconnects at the specified time interval instead of using the backoff strategy.

If the remote host terminates the connection, the probe closes the connection on the host machine. The operating system is not allowed to close the connection.

## Hosts file format

The probe needs a hosts file to connect to multiple hosts. The **HostsFile** property specifies from which file the probe gets host information. The probe reads the hosts file and attempts to connect and log in to each host.

If the **HostsFile** property is defined, the probe attempts to open the specified file. This file must contain the required connection information for the probe in the following format:

```
host port : chi, cho, ft, ia, ri, ra, af;
```

The following table describes the hosts file format.

Table 3. Hosts file format	
Item	Description
host	This item specifies the host to which the probe connects.
port	This item specifies the port to which the probe connects.
chi	This item specifies the chat in string for the host/port.
cho	This item specifies the chat out string for the host/port.

Table 3. Hosts file format (continued)	
Item	Description
ft	This item specifies the flush time value for the host/port.
ia	This item specifies the inactivity alarm for the host/port.
ri	This item specifies the reconnection interval.
ra	This item specifies the reconnection attempts.
af	This item specifies the active flag. You must set this to True. Setting this to False deactivates the connection.

When you create a hosts file, note the following information:

- In the hosts file, the end of each entry is indicated by a semicolon.
- Whitespace and commas separate individual elements within a line.
- Whitespace characters are sp and tab.
- Comment lines are preceded #.
- Blank lines are ignored.
- You can force the probe reread the hosts file without restarting the probe by issuing a USR1 signal. To do this, enter the command `kill -USR1 pid`, where *pid* is the process identifier of the probe.

## 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
<b>ChatinString</b> <i>string</i>	-chatinstring <i>string</i>	Use this property to specify the chat in script for connection to the host system.  The default is " ".
<b>ChatoutString</b> <i>string</i>	-chatoutstring <i>string</i>	Use this property to specify the chat out script for disconnection from the host system.  The default is " ".
<b>DateStreamCapture</b> <i>integer</i>	-datestreamcapture <i>integer</i>	Use this property to specify whether the probe appends a date and time to the stream capture file name:  0: The probe does not append the date and time to the stream capture file name.  1: The probe appends the date and time to the stream capture file name.  The default is 0.  <b>Note :</b> Setting the <b>DateStreamCapture</b> property to 1 prevents the stream capture file from being overwritten.
<b>ExpectTimeout</b> <i>integer</i>	-expecttimeout <i>integer</i>	Use this property to specify the time (in seconds) that the probe allows for a response to the expect part of the chat in/out script.  The default is 20.
<b>FlushTime</b> <i>integer</i>	-flushtime <i>integer</i>	Use this property to specify the time (in seconds) that the probe waits before flushing the event to the ObjectServer.  The default is 5.

Table 4. Properties and command line options (continued)

Property name	Command line option	Description
<b>Host</b> <i>string</i>	-host <i>string</i>	Use this property to specify the name of the host to which the probe connects.  The default is localhost.
<b>HostsFile</b> <i>string</i>	-hostsfile <i>string</i>	Use this property to specify the file that the probe uses to connect to multiple hosts.  The default is " " (hosts file not used).
<b>Inactivity</b> <i>integer</i>	-inactivity <i>integer</i>	Use this property to specify the time (in seconds) that the probe allows a port to be inactive before disconnecting.  The default is 0 (probe does not disconnect if the host becomes inactive).
<b>InactivityRetry</b> <i>integer</i>	-inactivityretry <i>integer</i>	Use this property to specify the number of consecutive periods of inactivity the probe allows before attempting to reconnect to the host.  The default is 0 (which disables this feature).
<b>InvalidAlarmLogFile</b> <i>string</i>	-invalidlog <i>string</i>	Use this property to specify the file to which the probe logs invalid alarms. The current date and time is appended to this filename.  The default is \$OMNIHOME/var/motorola_omcr_iden.invalid.
<b>InvalidAlarmLogging</b> <i>integer</i>	-invalidalarmlogging (equivalent to <b>InvalidAlarmLogging</b> with a value of 1)	Use this property to specify whether the probe logs invalid alarms: 0: The probe does not log invalid alarms. 1: The probe logs invalid alarms.  The default is 0.
<b>MaxEvents</b> <i>integer</i>	-maxevents <i>integer</i>	Use this property to specify the maximum number of events the probe attempts to read simultaneously from a given source.  The default is 1.  <b>Note :</b> You can increase this number to increase the event throughput when a large number of events are generated.
<b>MaxInvalidAlarmLog FileSize</b> <i>integer</i>	-maxinvalidlog <i>integer</i>	Use this property to specify the maximum size for the invalid alarm file. When this limit is reached, the probe creates a new file with a new date and time.  The default is 1048576.

Table 4. Properties and command line options (continued)

Property name	Command line option	Description
<b>MaxStreamCapFileSize</b> <i>integer</i>	-maxstreamcapfilesize <i>integer</i>	Use this property to specify the maximum size (in bytes) of the stream capture file. When this limit is reached, the probe creates a new file.  The default is 0.  <b>Note :</b> If the probe reads a stream whose size exceeds the remaining allowable space in the file, it creates a new one, thus storing the whole stream in a single file.
<b>Port</b> <i>integer</i>	-port <i>integer</i>	Use this property to specify the port to which the probe connects.  The default is 23.
<b>ReadTimeout</b> <i>integer</i>	-readtimeout <i>integer</i>	Use this property to adjust the granularity of the timeouts on the reads/writes to and from the device(s) to which the probe is connected. Enter a value in milliseconds.  The default is 100.
<b>ReconnectionAttempts</b> <i>integer</i>	-reconnectionattempts <i>integer</i>	Use this property to specify the maximum number of times that the probe attempts to reconnect to the socket.  The default is 0 (probe makes unlimited attempts to reconnect to the socket).
<b>ReconnectionInterval</b> <i>integer</i>	-reconnectioninterval <i>integer</i>	Use this property to specify the time (in seconds) between successive reconnection attempts.  The default is 0 (probe uses standard backoff strategy).
<b>StreamCaptureFile</b> <i>string</i>	-streamcapturefile <i>string</i>	Use this property to specify the file the probes uses to store the input stream log.  The default is " ".  <b>Note :</b> Leaving this property blank disables the stream capture function. When you no longer require data for debugging, you should disable the stream capture function.

## Escape codes

You can use C-style escape codes in the **ChatinString** and **ChatoutString** properties. This allows you to easily define whether to send escape code sequences after commands.

For example, the following chatin string sends a carriage return character (\r) after the user name and password:

```
ChatinString : ".*: user\r .*: passwd\r .*:"
```

The following table explains the character sequences that are recognized.

Table 5. ChatinString escape codes	
Escape code	Character
\b	This escape code specifies the backspace character.
\f	This escape code specifies the form-feed character.
\n	This escape code specifies the new-line character.
\r	This escape code specifies the carriage return character.
\t	This escape code specifies the tab character.
\\	This escape code specifies the backslash character.
\'	This escape code specifies the single quote character.
\"	This escape code specifies the double quote character.

**Note :** Due to the way in which the above properties are parsed, the escape sequences for backslash, single quote and double quote must be double-escaped. For example, to send a backslash character (\\), use \\\\.

## 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.

Table 6. Elements	
Element name	Element description
\$AdditionalInfo	This element specifies the additional information contained in the alarm.

Table 6. Elements (continued)	
Element name	Element description
\$Class	This element specifies the class of event.
\$ClearanceType	This element specifies the details of why the alarm was raised.
\$Date	This element specifies the date of the alert.
\$DeviceType	This element specifies the type of device.
\$EventType	This element specifies the type of event.
\$FUSeverity	This element specifies the additional severity information. This element is usually null.
\$Host	This element specifies the name of the host.
\$Id	This element specifies the numerical identifier of the path.
\$Instance	This element specifies the class, network element, site and site details information shown in one element.
\$NE	This element specifies the name of network element.
\$Operator	This element specifies the username of the operator.
\$Port	This element specifies the port on which the probe listens for alarms.
\$Severity	This element specifies the severity of the alarm; for example, critical, major, minor, and so forth.
\$Site	This element specifies the name of site.
\$SiteDetails	This element specifies the details of the site.
\$State	This element specifies the current state of the Network Element.
\$Type	This element specifies the type of alarm.
\$Ufmi	This element specifies the urban, fleet and member identifier. This is created when the probe receives an MCB MUX event.

## 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 7. Error messages

Error	Description	Action
<i>Filename</i> File not found	The probe could not find the hosts file.	Check that the host file exists and the permissions are correct.
Error initialising list	This is an internal probe error.	Rerun the probe. If that does not solve the problem, contact IBM Technical Support.
Error Reading HostsFile <i>Filename</i>	The probe could not read the hosts file.	Check that the host file is formatted correctly and that the permissions are correct.
Event from <i>event source</i> did not match the expected format	The event was not in the correct format.	Check that you are connected to the correct port.
Expect timed out after <i>seconds</i> seconds	When the probe tried to log in or out of the host, the expect part of the chat timed out.	Check that the host you are connecting to is running correctly. Check that the login name and password are appropriate for the target host. It may be that the login name was not found and the connection timed out while the target host was trying to validate it.
Failed to read from socket - disconnecting	The probe had successfully logged into the target host, but something interrupted the connection.	Check that the target host is working correctly.
Failed to rollover InvalidAlarmLogFile	The probe was unable to open another alarm log file to write to.	Check that there is sufficient disk space and that the permissions are correct.
Failed to send <i>sendchat</i>	When the probe tried to log in or out of the host, the send part of the chat failed. <i>sendchat</i> indicates the part of the chat message the probe was trying to send when it failed.	Check that the host you are connecting to is running correctly.
Hostfile entry is not valid:	A line in the hosts file was formatted incorrectly. The <i>lineprint</i> part of the error message indicates the line.	Check the hosts file.
Invalid <i>message</i>	The event was not in the correct format.	Check that you are connected to the correct port.
Invalid hostfile entry <i>lineprint</i>	A line in the hosts file was formatted incorrectly. The <i>lineprint</i> part of the error message indicates the line.	Check the hosts file.



Table 7. Error messages (continued)		
Error	Description	Action
Invalid Port Number for Host <i>Hostname</i>	The port number cannot be zero.	Change the <b>Port</b> property to a valid value.
Only managed to write <i>number</i> out of <i>number</i> characters	The probe was unable to write to the alarm log file.	Check that there is sufficient disk space and that the permissions are correct.
Read error during expect	The probe could not process the response from the host during an expect. This could be caused by a problem with the target that caused the connection to time out, or the response from the target could have been corrupted.	If the probe is configured to reconnect, it tries again to establish the connection.
Too many tokens in hostfile entry	A line in the hosts file was formatted incorrectly. There are too many entries on the line.	Check the hosts file.
Unable to open file	The probe was unable to open the hosts file.	Check the hosts file.
Unable to write to file <i>filename</i>	The probe was unable to write to the alarm log file.	Check that there is sufficient disk space and that the permissions are correct.

## 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*.

Table 8. ProbeWatch messages		
ProbeWatch message	Description	Triggers or causes
Connection attempted ( <i>host</i> , <i>port</i> )	The probe is trying to establish a connection to the host.	The probe is trying to establish a connection to the host.
Connection failed ( <i>host</i> , <i>port</i> )	The probe has failed to establish a connection to the host.	General connection failure due either to an incorrect setting of the <b>Host</b> or <b>Port</b> property, or the host refusing a connection request from the probe.
Connection lost ( <i>host</i> , <i>port</i> )	The probe has lost the connection to the host.	The device has dropped the connection to the probe.

Table 8. ProbeWatch messages (continued)

ProbeWatch message	Description	Triggers or causes
Connection succeeded ( <i>host, port</i> )	The probe has successfully established a connection to the host.	The probe has successfully established a connection to the host.
Disconnected from system due to shutdown signal	The probe has disconnected from the host.	A shutdown signal was sent to the probe.
Disconnecting from system due to Inactivity alarm ( <i>host, port</i> )	The probe has disconnected from the device.	The device was silent for the period specified by the <b>Inactivity</b> property. If you want the probe to time out less frequently, increase this value.
Disconnection attempted ( <i>host, port</i> )	The probe is attempting to disconnect from the host.	The probe is shutting down or has lost the connection.
Disconnection failed ( <i>host, port</i> )	The probe has failed to disconnect cleanly from the host.	There has been a corruption of the memory location storing the connection details.
Disconnection succeeded ( <i>host, port</i> )	The probe has disconnected from the host.	The probe shut down or lost the connection.
Event from ( <i>host, port</i> ): did not match the expected format!	The event received by the probe does not match any of the supported event formats.	The probe has received an event that does not match any of the supported event formats.
Failed to get events	A problem occurred while receiving events.	Either there was a problem initializing the connection due to insufficient memory or (if this message was sent after some events had been parsed) there was a connection failure.
Going Down	The probe is shutting down.	The probe is shutting down after performing the shutdown routine.
Have lost all connections - going down	The probe is shutting down because all of its connections have been deactivated.	The number of times that the probe has attempted reconnection for each of its connections has exceeded that specified by the <b>ReconnectionAttempts</b> property.
Login attempted ( <i>host, port</i> )	The probe is trying to log in to the host.	The probe is trying to log in to the host.
Login failed ( <i>host, port</i> )	The probe has failed to log in to the host.	The probe has encountered a problem while logging in. Check that the <b>ChatinString</b> property is specified correctly.

Table 8. ProbeWatch messages (continued)

ProbeWatch message	Description	Triggers or causes
Login succeeded ( <i>host</i> , <i>port</i> )	The probe has logged in to the host.	The probe has logged in to the host.
Logout attempted ( <i>host</i> , <i>port</i> )	The probe is trying to log out from the host.	The probe has received a command to shut down.
Logout failed ( <i>host</i> , <i>port</i> )	The probe has failed to log out from the host.	The probe has encountered a problem while logging out. Check that the <b>ChatoutString</b> property is specified correctly.
Logout succeeded ( <i>host</i> , <i>port</i> )	The probe has logged out from the host.	The probe has logged out from the host.
Running	The probe is running normally.	The probe has just been started.



---

## 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 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, 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.









SC23-7885-01

