



@server

iSeries

REXEC server

Version 5 Release 3





@server

iSeries

REXEC server

Version 5 Release 3

Note

Before using this information and the product it supports, be sure to read the information in "Notices," on page 11.

Fourth Edition (August 2005)

| This edition applies to version 5, release 3, modification 0 of IBM Operating system/400 (product number 5722-SS1)
| and to all subsequent releases and modifications until otherwise indicated in new editions. This version does not
| run on all reduced instruction set computer (RISC) models nor does it run on CISC models.

© Copyright International Business Machines Corporation 2000, 2005. All rights reserved.

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

Contents

REXEC	1		RXCS0100	5
Print this topic.	1		Determining Problems with REXEC	7
Changing Attributes	1		Materials Required for Reporting REXEC Problems	8
REXEC Command Considerations	2		Getting a Copy of a REXEC Server Job Log	9
Selecting a Command Processor	2		Tracing the REXEC Server	9
REXEC Connection Usage	2			
For iSeries CL command processing	2			
For Qshell and spawned path command processing	3			
Spooled Output Considerations	3		Appendix. Notices	11
Client Considerations	3		Programming Interface Information	12
REXEC Server Jobs and Job Names	3		Trademarks	12
Creating REXEC Server Spooled Job Logs.	4		Terms and conditions for downloading and printing publications	13
Exit Points for Controlling REXEC Server.	4		Code disclaimer information.	13

REXEC

The Remote Execution (REXEC) server is a Transmission Control Protocol/Internet Protocol (TCP/IP) application that allows a client user to submit system commands to a remote server system. The user's client program sends the user identifier, password, and command to run to the server. The server validates the user, runs the requested command, and returns the results of the command to the client.

Commands submitted to the iSeries host fall into three categories:

iSeries command processor

You run iSeries command processor commands by specifying QCAPCMD as the target of the client REXEC.

Qshell command interpreter (OS/400® option 30)

You can use the Qshell interpreter by specifying qsh as the target of client REXEC.

"Spawned paths"

You can run any iSeries program in a "child" (spawned) job by specifying the complete path to the program or shell script as the target of the REXEC command.

You can work with REXEC server properties through iSeries Navigator, the graphical user interface (GUI) for OS/400.


Use Print this topic to print out the REXEC articles.

Print this topic

To view or download the PDF version, select REXEC (about 117 KB or 32 pages).

To save a PDF on your workstation for viewing or printing:

1. Open the PDF in your browser (click the link above).
2. In the menu of your browser, click **File**.
3. Click **Save As...**
4. Navigate to the directory in which you would like to save the PDF.
5. Click **Save**.

If you need Adobe Acrobat Reader to view or print these PDFs, you can download a copy from the Adobe Web site (www.adobe.com/prodindex/acrobat/readstep.html) .

Changing Attributes

The Change REXEC Attributes (CHGRXCA) command changes the REXEC server attributes. The following are two ways to get to this command prompt:

- Specify the CHGRXCA command.
- Select Option 17 on the Configure TCP/IP Applications (CFGTCPAPP) display.

Note: You must have *IOSYSCFG special authority to make changes to the REXEC attributes with the CHGRXCA command.

Figure 1. Change REXEC Attributes (CHGRXCA)

Change REXEC Attributes (CHGRXCA)

Type choices, press Enter.

Autostart server	*YES	*YES, *NO, *SAME
Number of initial servers . . .	2	1-20, *SAME, *DFT
Inactivity timeout	300	1-2147483647, *SAME, *DFT
Coded character set identifier	00437	1-65533, *SAME, *DFT

REXEC Command Considerations

The REXEC server is restricted to running commands that are allowed in batch jobs. The command must have *BATCH as one of the *Where allowed to run* values.

The maximum length of a command that the REXEC server can process is 4000 bytes. Some REXEC clients limit the command to a smaller length.

For spawned paths, the program that runs in the child process must be either a program object in the QSYS.LIB file system (*PGM object) or a shell script. You must specify the path with the proper syntax for the file system in which the file resides.

For Qshell commands, you can put the same commands that you would enter at an interactive command line into a non-interactive shell script.

Selecting a Command Processor

You can use the REXEC server command processing selection exit program (QIBM_QTMX_SVR_SELECT) to select which command processor the REXEC server uses to run the submitted command. (If you do not use an exit program, the REXEC server uses the Control Language (QCAPCMD) processor.) The allowed command processors are:

- Control Language (QCAPCMD)
- Qshell interpreter
- Spawned path (a shell script or program object)

Because data conversion is optional for the Qshell and spawn options, the exit program also selects whether the REXEC server performs ASCII-EBCDIC conversions on the stdin, stdout, and stderr streams.

REXEC Connection Usage

The REXEC protocol allows a REXEC client to specify whether to use one or two connections for returning data.

For iSeries CL command processing

If you choose iSeries CL command processing and two connections, normal output returns on the first connection, and error output returns on the second connection. The REXEC server returns all spooled data that is written to the default printer file (*PRTF). This includes data that is written to the screen if the command is run in an interactive job. Any messages written to the job log return to the client on the second connection.

If the client specifies that all data returns on a single connection, the job log messages are returned first, followed by any spooled output.


For Qshell and spawned path command processing

For Qshell or spawned path command processing, the REXEC server by default returns normal output on the first connection and error output on the second connection. (The REXEC stdin, stdout, and stderr streams are mapped to file descriptors 0, 1, and 2 respectively, and the QIBM_USE_DESCRIPTOR_STDIO environment variable is set to Y.) These options allow you to redirect input and output.

Choosing the Qshell command processor sets these environment variables:

- `TERMINAL_TYPE=REMOTE`
- `PATH=/usr/bin`
- `LOGNAME= user`, where *user* is the user profile
- `HOME=homedir`, where *homedir* is the user's home directory

The child job inherits any other environment variables that the exit program sets.

Spawned child processes are batch jobs or prestart jobs. They cannot do interactive I/O. See WebSphere® Development Studio: ILE C/C++ Programmer's Guide  for complete details on this support.

Spooled Output Considerations

Note: This section applies only to iSeries CL commands.

The REXEC server overrides the default printer file (*PRTF) to capture spooled output. Any resulting spool files are tagged with the user data field set to REXECSVR. After the REXEC server runs the specified command, each spooled file with this user data tag is retrieved, returned to the client, and then deleted. If more than one spool file is created, the files are processed in the order created, as determined by the spool file number.

If the command or program run through REXEC performs its own print file override and changes the user data, the REXEC server is unable to capture and return the resulting spooled data.

Client Considerations

The iSeries REXEC client (RUNRMTCMD) uses a single connection for returned data, which is written to a spooled file on the client system.

The UNIX®, OS/2®, Windows® 95, and Windows NT® REXEC clients all use two connections, returning the normal output to the stdout stream and the error output to the stderr stream.

The VM REXEC client uses a single connection for the returned data, which is written to the console of the user.

REXEC Server Jobs and Job Names

REXEC server jobs start when you run the STRTCP command and set the REXEC AUTOSTART parameter to *YES. You can also start REXEC server jobs by running the STRTCP SVR command with a SERVER parameter of *REXEC or *ALL. These jobs run in the QSYSWRK subsystem. Their purpose is monitoring and processing requests from REXEC client users. The format for the names of these jobs is QTRXCnnnnn, where nnnn is a 5-digit decimal number.

To work with jobs in the QSYSWRK subsystem, including REXEC server jobs, specify the following command:

```
WRKSBSJOB SBS(QSYSWRK)
```

If you choose to have commands processed by the Qshell command interpreter, you start Qshell is by using the spawn() application program interface (API) to create a child job.

If you choose to have commands interpreted as spawn path names, the REXEC server treats command strings as path names and passes them to the spawn() API. Spawned child processes are batch jobs or prestart jobs. Shell scripts are allowed for the child process. If you specify a shell script, the appropriate shell interpreter program is called. The shell script must be a text file and must contain this format on the first line of the file: `#!/interpreter_path <options>`.

Creating REXEC Server Spooled Job Logs

The REXEC server automatically writes a server job log to a spooled file when it ends with an error.

To have a spooled job log produced at the end of each REXEC session and each time the REXEC server ends, use the CHGJOB command, as follows:

```
CHGJOB JOB(QTCP/QTMXRCS) LOG(4 00 *SECLVL)
```

To obtain a spooled job log only when a server ends, use the CHGJOB command, as follows:

```
CHGJOB JOB(QTCP/QTMXRCS) LOG(4 00 *NOLIST)
```

Exit Points for Controlling REXEC Server

With the use of exit programs, the experienced programmer can create customized processing during an application. If the REXEC server finds a program registered to one of the exit points for the server, it calls that program using parameters that are defined by the exit point.

An exit point is a specific point in the REXEC program where control may pass to an exit program. An exit program is a program to which the exit point passes control.

For each exit point, there is an associated programming interface, called an exit point interface. The exit point uses this interface to pass information between the REXEC application and the exit program. Each exit point has a unique name. Each exit point interface has an exit point format name that defines how information is passed between the REXEC application and the customer-written exit program.

Different exit points may share the same exit point interface. When this is the case, multiple exit points can call a single exit program. For more information about using exit programs, see Using server exit programs.

Exit point performance

The following table lists exit points that give you additional control over the REXEC server.

Table 1.

TCP/IP Exit Points	Application	VLRQ0100	TCPL0100	TCPL0300	"RXCS0100" on page 5	Brief description
QIBM_QTMX_SERVER_REQ	REXEC	X				The TCP/IP request validation exit point provides additional control for restricting an operation.

Table 1. (continued)

QIBM_QTMX_SVR_LOGON ¹	REXEC		X	X ²		The TCP/IP server logon exit point () provides additional control over authenticating a user and setting up the user's environment for the REXEC server.
QIBM_QTMX_SVR_SELECT	REXEC				X	The REXEC server command processing selection exit point allows you to specify which command processor the REXEC server uses for interpreting and running your commands.

¹ - An exit point may have more than one format, but an exit program can only be registered for one of the exit point formats. Examine each of these formats, then choose the one most appropriate for your system.

² - This format is available starting with V5R1.

Note:

- The same interface format is used for request validation for the FTP client, FTP server, REXEC server, and TFTP server. This allows the use of one exit program for request validation of any combination of these applications.
- The same interface format is used for server log-on processing for the FTP server and REXEC server applications. This allows the use of one exit program to process log-on requests for both of these applications.

RXCS0100

The REXEC server command processing selection (RXCS0100) exit program enables you to select:

- Which command processor runs the command that the REXEC client user provides
- Whether the REXEC server converts data between ASCII and EBCDIC (for Qshell commands or spawn path names)

Table 2. Required parameter group:

Parameter	Description	Input or Output	Type and length
1	User profile	Input	Char(10)
2	Remote IP address	Input	Char(*)
3	Length of remote IP address	Input	Binary(4)
4	Command string	Input	Char(*)
5	Length of command string	Input	Binary(4)
6	Command processor identifier	Output	Binary(4)

Table 2. Required parameter group: (continued)

7	Character conversion option	Output	Binary(4)
---	-----------------------------	--------	-----------

Exit point format name: RXCS0100

Exit point name: QIBM_QTMX_SVR_SELECT

Note: Character data passes to the exit program in the coded character set identifier (CCSID) of the job. If the job CCSID is 65535, the server uses the default CCSID of the job.

Required Parameter Group

User profile

INPUT; CHAR(10) The user profile under which the requested operations is run.

Remote IP address

INPUT; CHAR(*) The Internet protocol (IP) address of the REXEC client system. This string is in dotted decimal format, left justified.

Length of remote IP address

INPUT; BINARY(4) Indicates the length (in bytes) of the remote IP address.

Command string

INPUT; CHAR(*) The command to be run as specified by the REXEC client.

Length of command string

INPUT; BINARY(4) Indicates the length (in bytes) of the command string.

Command processor identifier

OUTPUT; BINARY(4) Indicates the command processor that you want the server to use for interpreting and running the command. The following values are valid:

Table 3. Valid values

Value	Operation
0	iSeries™ control language — The server processes the command as an iSeries control language (CL) command. This is the default value.
1	Qshell command — The Qshell command interpreter processes the command. The server uses the spawn() application program interface (API) to call QShell as a child job.
2	Spawn path name — The server treats the command name as a path name and passes it to the spawn() application program interface (API), which runs as the child job.

Character conversion option

OUTPUT; BINARY(4) Indicates whether the REXEC server performs ASCII-EBCDIC character conversion for data that is passed on the stdin, stdout, and stderr streams. These values are valid:

Table 4. Valid values

Value	Operation
0	Do not convert data. The server transfers all data on the stdin, stdout, and stderr streams without converting it.

Table 4. Valid values (continued)

1	<p>Convert data. This is the default.</p> <ul style="list-style-type: none"> • The server converts data in the stdin stream from the ASCII CCSID that the CHGRXCA command specifies to the job CCSID. If the job CCSID is 65535, the server uses the default CCSID of the job. • The server converts data in the stdout and stderr streams from the job CCSID to the ASCII CCSID that the CHGRXCA command specifies. If the job CCSID is 65535, the server uses the default CCSID of the job.
---	---

Usage Note:

- If you add exit programs to both the QIBM_QTMX_SERVER_REQ and QIBM_QTMX_SVR_SELECT exit points, REXEC server first calls the exit program that you add to the QIBM_QTMX_SERVER_REQ exit point. If this program allows the operation, the server then calls the exit program that you add to the QIBM_QTMX_SVR_SELECT exit point.
- When you set the Command processor identifier parameter to 0 (iSeries Control Language command), the conversion option is ignored. The server always performs character conversion for the CL commands.
- When you set the command processor identifier to 1 (QShell Command), the server sets these environment variables:
 - TERMINAL_TYPE=REMOTE
 - PATH= /usr/bin:
 - LOGNAME= *user* (where *user* is the user profile)
 - HOME= *homedir* (where *homedir* is the user's home directory)

Determining Problems with REXEC

If you detect a problem when using the REXEC server, you can use the following flow chart and cause lists to identify potential problems. However, you may be having problems with general TCP/IP connectivity. In that case, it may be beneficial to first use the TCP/IP troubleshooting topic to identify basic TCP/IP problems before using the REXEC flow chart for more localized problems

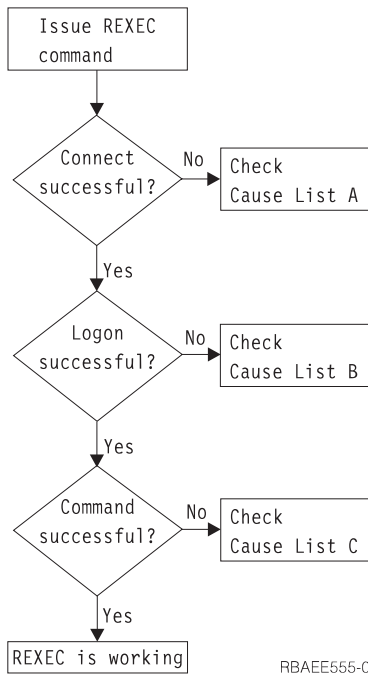


Figure 2. REXEC Server Problem Analysis

Cause List A

1. Check to see that the REXEC server is running. If not, start it with the STRTCPSVR SERVER(*REXEC) command.
2. If the message "Connection refused" is returned to the REXEC client, check the exit program associated with the exit point QIBM_QTMX_SERVER_REQ. This exit program has either specified that the connection should be rejected, returned a value that is not correct for the Allow Operation parameter, or ended abnormally. Examine the REXEC server job log for messages. Resolve any problems with the exit program and install the corrected version.

Cause List B

1. Check your user ID and password by logging on to the system. If you are unable to do so, contact the system administrator to verify that your user ID and password are correct.
2. Check the exit program associated with exit point QIBM_QTMX_SERVER_LOGON (if any). This exit program has either specified that the connection should be rejected, returned a value this is not correct for the Allow Operation parameter, or ended abnormally. Examine the REXEC server job log for messages.

Cause List C

1. Check for any job log messages returned to the REXEC client. Resolve any indicated problems and try the command again.
2. If the message "Command Rejected" is returned to the REXEC client, check the exit program associated with the exit point QIBM_QTMX_SERVER_REQ. This exit program may be specifying that the command should be rejected, returning a value this is not correct for the Allow Operation parameter, or ending abnormally. Examine the REXEC server job log for messages. Resolve any problems with the exit program and install the corrected version.
3. Verify that the correct ASCII CCSID is configured for the REXEC server. If not, set the correct CCSID with the CHGRXCA command.

Materials Required for Reporting REXEC Problems

Any REXEC problem reported to IBM® should include the following:

- A communications trace from the time of the failure (Request TCP/IP data only) formatted for ASCII. If you are not familiar with the procedure for collecting a communications trace, see Communications trace.
- If the REXEC server has logged software error data, submit this information.

Note: The system value QSFWERRLOG must be set to *LOG for software error logging to take place. If an error occurs while QSFWERRLOG is set to *NOLOG, change the value to *LOG, try to recreate the error, and submit the logged software error data. If logged software error data is submitted, there is no need to perform a trace of the REXEC server.

- The QTCPIP and any REXEC server job logs.

Getting a Copy of a REXEC Server Job Log

To have the REXEC server save job logs, see “Creating REXEC Server Spooled Job Logs” on page 4.

Tracing the REXEC Server

The REXEC server can be traced by creating a data area. Note that running the REXEC server with trace running may cause a significant performance impact.

To trace the REXEC server:

1. Create the data area using the following command: CRTDTAARA DTAARA(QUSRSYS/QTMXRXCDBG) TYPE(*LGL) LEN(1)
2. Perform the REXEC operation that you want to trace.
3. Delete the data area using the following command: DLTDTAARA DTAARA(QUSRSYS/QTMXRXCDBG)
4. Enter the following command to find the output queue:

```
DSPPSYSVAL QPRTDEV
```

For example, the following display appears:

```

                                Display System Value
System value . . . . . : QPRTDEV
Description . . . . . : Printer device description
Printer device . . . . : PRT01      Name
```

Figure 3. Display System Value Display

The printer device is also the name of the default system output queue.

5. Record the name of the printer device. In this example, PRT01 is the printer device.
6. Press F12 (Cancel) to return to the display where you entered the DSPPSYSVAL command.
7. Type the following command:

```
WRKOUTQ OUTQ(printer-device)
```

Replace printer-device with the printer device recorded in the previous display. PRT01 is the output queue in this example. For example, the following display appears:

```

                                Work with Output Queue
Queue: PRT01      Library: QGPL      Status: RLS
Type options, press Enter.
 1=Send  2=Change  3=Hold  4=Delete  5=Display  6=Release  7=Messages
 8=Attributes  9=Work with printing status
Opt  File      User      User Data  Sts  Pages  Copies  Form Type  Pty
-   QTCPPRT    QTCP      QTMSMTP    HLD   46    1      *STD      5
-   QPSRVTRC   QSECOFR           HLD   44    1      *STD      5
```

Figure 4. Work with Output Queue Display

8. Press F18 (Bottom) to get to the bottom of the spooled file list if More... appears on the display.

9. Find the last file named QPSRVTRC with the same user as the user who was logged on the REXEC server when the trace was created.
10. Press F11 (View 2) to view the date and time of the file you want to work with.
11. Verify that you are working with the most recent spooled file, QPSRVTRC.
12. Indicate in the problem report that the trace was tried and it failed. Send whatever trace information there is with the problem report.

Appendix. 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 give 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,
| IBM License Agreement for Machine Code, 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.

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.

Programming Interface Information

| This REXEC server topic documents intended Programming Interfaces that allow the customer to write
| programs to obtain the services of 5722-SS1.

Trademarks

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM
iSeries
OS/2
OS/400
Websphere
400

Microsoft®, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation 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, and service names may be trademarks or service marks of others.

Terms and conditions for downloading and printing publications

| Permissions for the use of the information you have selected for download are granted subject to the following terms and conditions and your indication of acceptance thereof.

| **Personal Use:** You may reproduce this information for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative works of this information, or any portion thereof, without the express consent of IBM.

| **Commercial Use:** You may reproduce, distribute and display this information solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of this information, or reproduce, distribute or display this information or any portion thereof outside your enterprise, without the express consent of IBM.

| Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the information or any data, software or other intellectual property contained therein.

| IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the information is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

| You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations. IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THIS INFORMATION. THE INFORMATION IS PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

All material copyrighted by IBM Corporation.

| By downloading or printing information from this site, you have indicated your agreement with these terms and conditions.

Code disclaimer information

IBM grants you a nonexclusive copyright license to use all programming code examples from which you can generate similar function tailored to your own specific needs.

| SUBJECT TO ANY STATUTORY WARRANTIES WHICH CANNOT BE EXCLUDED, IBM, ITS PROGRAM DEVELOPERS AND SUPPLIERS MAKE NO WARRANTIES OR CONDITIONS EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OR CONDITIONS OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT, REGARDING THE PROGRAM OR TECHNICAL SUPPORT, IF ANY.

| UNDER NO CIRCUMSTANCES IS IBM, ITS PROGRAM DEVELOPERS OR SUPPLIERS LIABLE FOR ANY OF THE FOLLOWING, EVEN IF INFORMED OF THEIR POSSIBILITY:

- | 1. LOSS OF, OR DAMAGE TO, DATA;
- | 2. SPECIAL, INCIDENTAL, OR INDIRECT DAMAGES, OR FOR ANY ECONOMIC CONSEQUENTIAL DAMAGES; OR
- | 3. LOST PROFITS, BUSINESS, REVENUE, GOODWILL, OR ANTICIPATED SAVINGS.

| SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO SOME OR ALL OF THE ABOVE LIMITATIONS OR EXCLUSIONS MAY NOT APPLY TO YOU.



Printed in USA