

IBM Rational Developer for System z
Version 9.0.1

Host Configuration Quick Start Guide



IBM Rational Developer for System z
Version 9.0.1

Host Configuration Quick Start Guide



Note

Before using this information, be sure to read the general information under “Documentation notices for IBM Rational Developer for System z” on page 55.

Ninth edition (December 2013)

This edition applies to IBM Rational Developer for System z Version 9.0.1 (program number 5724-T07) and to all subsequent releases and modifications until otherwise indicated in new editions.

Order publications by phone or fax. IBM Software Manufacturing Solutions takes publication orders between 8:30 a.m. and 7:00 p.m. eastern standard time (EST). The phone number is (800) 879-2755. The fax number is (800) 445-9269. Faxes should be sent Attn: Publications, 3rd floor.

You can also order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address below.

IBM welcomes your comments. You can send your comments by mail to the following address:

IBM Corporation
Attn: Information Development Department 53NA
Building 501 P.O. Box 12195
Research Triangle Park NC 27709-2195
USA

You can fax your comments to: 1-800-227-5088 (US and Canada)

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Note to U.S. Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

© Copyright IBM Corporation 2000, 2013.

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

Contents

Figures	v
--------------------------	----------

Tables	vii
-------------------------	------------

About this document	ix
Who should read this book	ix

IBM Rational Developer for System z Host Configuration Quick Start Guide	1
---	----------

Chapter 1. Planning.	3
Requisites	3
Requisite products	3
Required resources	3
Client user ID	4
Servers	4
Configuration method	4
Deployment and upgrade	5

Chapter 2. Basic customization.	7
Customization setup.	7
PARMLIB changes	7
Setting the z/OS UNIX limits in BPXPRMxx.	7
Adding the started tasks to COMMNDxx.	8
Setting the APF authorizations in PROGxx	8
Requisite LINKLIST and LPA definitions	8
PROCLIB changes	9
JMON, JES Job Monitor started task	9
RSED, RSE daemon started task	10
ELAXF* remote build procedures	10
Creating the security definitions	12
FEJJCNFG, JES Job Monitor configuration file	12
rsed.envvars, RSE configuration file	13
ISPF.conf, ISPF's TSO/ISPF Client Gateway configuration file	17

Chapter 3. Installation verification	19
Verifying the started tasks	19
JMON, JES Job Monitor	19
RSED, RSE daemon.	19
IVP operator commands	19
PassTicket reusability	19
RSE daemon connection	19
ISPF Client Gateway	19

Chapter 4. Security definitions	21
Activate the security settings and classes	21
Define an OMVS segment for Developer for System z users	23
Define the Developer for System z started tasks	23
Define RSE as a secure z/OS UNIX server	23

Define the MVS program controlled libraries for RSE	24
Define PassTicket support for RSE.	24
Define application protection for RSE.	25
Define JES command security	25
Define data set profiles	25
Define the z/OS UNIX program controlled files for RSE	26
Verify the security settings	26

Chapter 5. Migration considerations	29
Version 9.0 migration notes	29
IBM Rational Developer for System z, FMID HHOP900	29
IBM Rational Developer for System z Host Utilities, FMID HAKG900	30
Migrate from version 8.5 to version 9.0	30
IBM Rational Developer for System z, FMID HHOP900	30
IBM Rational Developer for System z Host Utilities, FMID HAKG900	32

Chapter 6. Operator commands	33
Modify (F).	33
JES Job Monitor	33
RSE daemon	35

Chapter 7. Optional customization	45
(Optional) Common Access Repository Manager (CARMA)	45
(Optional) SCLM Developer Toolkit	45
(Optional) Application Deployment Manager (deprecated)	46
(Optional) Host-based code analysis	46
(Optional) pushtoclient.properties, Host-based client control	46
(Optional) ssl.properties, RSE SSL encryption	47
(Optional) rsecomm.properties, RSE tracing.	47
(Optional) include.conf, Forced includes for C/C++ content assist.	47
(Optional) z/OS UNIX subprojects	47
(Optional) Include preprocessor support.	48
(Optional) xUnit support for Enterprise COBOL and PL/I.	48
(Optional) CICS bidirectional language support	48
(Optional) Diagnostic IRZ messages for generated code.	49
(Optional) Integrated debugger.	49
(Optional) DB2 and IMS debug support	49
(Optional) File Manager support	49
(Optional) WORKAREA and /tmp cleanup.	50

Chapter 8. Host Configuration

Reference 51

Understanding Developer for System z	51
Security considerations	51
TCP/IP considerations.	51
WLM considerations	51
Tuning considerations	51
Performance considerations	51
Push-to-client considerations	52
CICSTS considerations.	52
User exit considerations	52
Customizing the TSO environment	52

Running multiple instances	52
Troubleshooting configuration problems	52
Setting up SSL and X.509 authentication.	53
Setting up TCP/IP	53

Documentation notices for IBM

Rational Developer for System z. 55

Copyright license	58
Trademark acknowledgments	58

Index 59

Figures

1.	JMON - JES Job Monitor started task	9	6.	ISPF.conf - ISPF configuration file	18
2.	RSED - RSE daemon started task	10	7.	MODIFY JMON operator command	33
3.	FEJJCNFG - JES Job Monitor configuration file	13	8.	MODIFY RSED operator command	35
4.	rsed.envvars - RSE configuration file	14			
5.	rsed.envvars - RSE configuration file (continued).	15			

Tables

1.	Required resources	3	5.	ELAXF*.	12
2.	Administrators needed for required tasks	4	6.	Security setup variables	21
3.	Sample ELAXF* procedures	10	7.	Thread pool error status	37
4.	ELAXF* high-level qualifier checklist	11			

About this document

This document discusses the configuration of the IBM® Rational® Developer for System z® functions. It includes brief instructions on how to configure IBM Rational Developer for System z Version 9.0.1 on the z/OS® host system. For complete details on the configuration of this product, see *Rational Developer for System z Host Configuration Guide* (SC23-7658).

The following names are used in this manual:

- *IBM Rational Developer for System z* is called *Developer for System z*.
- *IBM Rational Developer for System z Integrated Debugger* is called *Integrated Debugger*.
- *Common Access Repository Manager* is abbreviated to *CARMA*.
- *Software Configuration and Library Manager Developer Toolkit* is called *SCLM Developer Toolkit*, abbreviated to *SCLMDT*.
- *z/OS UNIX System Services* is called *z/OS UNIX*.
- *Customer Information Control System Transaction Server* is called *CICSTS*, abbreviated to *CICS®*.

This document is part of a set of documents that describe Developer for System z host configuration. Each of these documents has a specific target audience. You do not have to read all of these documents to complete the Developer for System z configuration.

- *Rational Developer for System z Host Configuration Guide* (SC23-7658) describes in detail all of the planning tasks, configuration tasks, and options (including optional ones), and provides alternative scenarios.
- *Rational Developer for System z Host Configuration Reference* (SC14-7290) describes Developer for System z design and gives background information for various configuration tasks of Developer for System z, z/OS components, and other products (such as WLM and CICS) related to Developer for System z.
- *Rational Developer for System z Host Configuration Quick Start Guide* (GI11-9201) describes a minimal setup of Developer for System z.
- *Rational Developer for System z Host Configuration Utility* (SC14-7282) describes the Host Configuration Utility, an ISPF panel application that guides you through basic and common optional customization steps for Developer for System z.

The information in this document applies to all of the Rational Developer for System z Version 9.0 packages.

Who should read this book

This document is intended for system programmers who are installing and configuring IBM Rational Developer for System z Version 9.0.1.

It lists and briefly documents the different steps that are needed to do a basic setup of the products. For more details on the listed actions and non-default settings, see *Rational Developer for System z Host Configuration Guide* (SC23-7658).

To use this book, you must be familiar with the z/OS UNIX System Services and MVS™ host systems.

IBM Rational Developer for System z Host Configuration Quick Start Guide

Chapter 1. Planning

This section summarizes the host component installation and configuration information in *Rational Developer for System z Host Configuration Guide* (SC23-7658). For more details on these subjects, see that publication:

- Migration considerations
- Planning considerations
- Preinstallation considerations
- Preconfiguration considerations
- Predeployment considerations
- Client checklist

Requisites

Requisite products

Developer for System z has a list of prerequisite software that must be installed and operational before the product will work. There is also a list of corequisite software to support specific features of Developer for System z. These requisites must be installed and operational at runtime for the corresponding feature to work as designed.

Refer to *Rational Developer for System z Prerequisites* (SC23-7659) to get a complete list of prerequisites and corequisites for your version of Developer for System z. The key requisites for a basic setup are the following:

- z/OS 1.8 or higher
- ISPF APAR OA43014 (TSO/ISPF Client Gateway)
- Java™ 6.0 or higher (31 or 64 bit)

Required resources

A basic setup for Developer for System z requires the allocation of the systems resources listed in Table 1.

Table 1. Required resources

Resource	Default value
APF-authorized data set	FEK.SFEKAUTH
started task	JMON and RSED
port for host-confined communication	6715
port for client-host communication	4035
port range for client-host communication	any available port is used
z/OS UNIX server security definition	UPDATE permission to BPX.SERVER for RSED started task
PassTicket security definitions	no default
MVS build procedures	ELAXF*

Table 2 lists the administrators who are needed for basic customization tasks.

Table 2. Administrators needed for required tasks

Administrator	Task
System	Typical system programmer actions are required for all of the customization tasks
Security	<ul style="list-style-type: none">• Define OMVS segment for Developer for System z users• Define data set profiles• Define started tasks• Define operator command security• Define z/OS UNIX server profiles• Define application security• Define PassTicket support• Define program controlled data sets• Define program controlled z/OS UNIX files
TCP/IP	Define new TCP/IP ports
WLM	Assign started task goals to the servers and their child processes

Client user ID

The user ID of a Developer for System z user must have the following attributes:

- TSO access, with a normal region size
- An OMVS segment defined in the security software for both user ID and default group, with these features:
 - A valid z/OS UNIX user ID. UID 0 is not required.
 - HOME directory with read, write, and execute access.
 - PROGRAM pointing to a valid z/OS UNIX shell, such as /bin/sh.
 - Blank ASSIZEMAX. Use the system defaults.
- Read and execute access to Developer for System z directories and files
- Read access to Developer for System z data sets

Servers

Developer for System z consists of the following permanently active servers, which can be started tasks or user jobs. These servers provide the requested services themselves, or start other servers such as z/OS UNIX threads or user jobs, to provide the service:

- JES Job Monitor (JMON) server provides all JES-related services.
- Remote Systems Explorer (RSE) server provides core services such as connecting the client to the host and starting other servers for specific services.

Configuration method

Developer for System z provides alternative methods to configure the host side of the product. These are the alternative methods:

- Using the ISPF panel application, which is designed to guide you through the required customization steps and selected optional customization steps. For more information, see the *Host Configuration Utility* (SC14-7282).
- Using the *Host Configuration Quick Start Guide* (GI11-9201), which is written to guide you through the required customization steps. The scope of this guide is limited to a basic setup.
- Using the *Host Configuration Guide* (SC23-7658), which is written to guide you through the required customization steps and all optional customization steps. All configurable options are covered in this guide, including some non-default scenarios.

Deployment and upgrade

Developer for System z is designed to support the installing of products once, followed by the copying of a minimal set of necessary data sets and directories to other systems for deployment. Running multiple copies, whether same or different version, of the products on the same system is also supported.

During an upgrade, before installing the product, make backups of all of the configuration files because the installation process might overwrite the files.

Chapter 2. Basic customization

This chapter summarizes the basic customization information in *Rational Developer for System z Host Configuration Guide* (SC23-7658). For more details, see that publication.

Customization setup

Developer for System z comes with several sample configuration files and sample JCL. To avoid overwriting your customizations when applying maintenance, you should copy all these members and z/OS UNIX files to a different location and to customize the copy.

Some functions of Developer for System z also require the existence of certain directories in z/OS UNIX, which must be created during the customization of the product. To ease the installation effort, a sample job, FEKSETUP, is provided to create the copies and the required directories.

Note: The *Rational Developer for System z Host Configuration Utility Guide* (SC14-7282) describes the host configuration using the Host Configuration Utility. The FEKSETUP job and the utility do some of the same tasks, with no way of checking to see if those tasks have already been performed. Therefore it is possible to undo changes that have already been made. For this reason, you should not use both methods for a single installation.

Customize and submit sample member FEKSETUP in data set FEK.SFEKSAMP to create customizable copies of configuration files and configuration JCL, and to create required z/OS UNIX directories. The required customization steps are described within the member.

This job performs the following tasks:

- Create FEK.#CUST.PARMLIB and populate it with sample configuration files.
- Create FEK.#CUST.PROCLIB and populate it with sample SYS1.PROCLIB members.
- Create FEK.#CUST.JCL and populate it with sample configuration JCL.
- Create FEK.#CUST.CNTL and populate it with sample server startup scripts.
- Create FEK.#CUST.ASM and populate it with sample assembler source code.
- Create FEK.#CUST.COBOL and populate it with sample COBOL source code.
- Create FEK.#CUST.SQL and populate it with sample SQL command files.
- Create /etc/rdz/* and populate it with sample configuration files.
- Create /var/rdz/* as work directories for various Developer for System z functions, and populate it with sample files.

PARMLIB changes

Note that the listed PARMLIB changes are for a basic setup only. More changes are needed if you select to use certain optional functions.

Setting the z/OS UNIX limits in BPXPRMxx

MAXASSIZE specifies the maximum address space (process) region size. Set MAXASSIZE in SYS1.PARMLIB(BPXPRMxx) to 2G, which is the maximum value allowed.

MAXTHREADS specifies the maximum number of active threads for a single process. Set MAXTHREADS in SYS1.PARMLIB(BPXPRMxx) to 1500 or higher.

MAXTHREADTASKS specifies the maximum number of active MVS tasks for a single process. Set MAXTHREADTASKS in SYS1.PARMLIB(BPXPRMxx) to 1500 or higher.

MAXPROCUSER specifies the maximum number of processes that a single z/OS UNIX user ID can have concurrently active. Set MAXPROCUSER in SYS1.PARMLIB(BPXPRMxx) to 50 or higher.

These values can be checked and set dynamically until the next IPL with the following console commands:

- DISPLAY OMVS,0
- SETOMVS MAXASSIZE=2G
- SETOMVS MAXTHREADS=1500
- SETOMVS MAXTHREADTASKS=1500
- SETOMVS MAXPROCUSER=50

Adding the started tasks to COMMNDxx

To start the Developer for System z RSE and JMON servers automatically at next system IPL, add the start commands to SYS1.PARMLIB(COMMANDxx).

After the servers are defined and configured, they can be started dynamically with the following console commands:

- S RSED
- S JMON

Setting the APF authorizations in PROGxx

For JES Job Monitor to access JES spool files, the FEJJMON module in the FEK.SFEKAUTH load library and the Language Environment® (LE) runtime libraries, namelyCEE.SCEERUN*, must be APF-authorized.

For ISPF to create the TSO/ISPF Client Gateway, the ISPZTS0 module in SYS1.LINKLIB must be APF-authorized.

By default, APF authorizations are defined in SYS1.PARMLIB(PROGxx).

Set the APF authorizations dynamically until the next IPL with the following console commands, where volser is the volume on which the data set resides if it is not SMS managed:

- SETPROG APF,ADD,DSN=FEK.SFEKAUTH,SMS
- SETPROG APF,ADD,DSN=CEE.SCEERUN,VOL=volser
- SETPROG APF,ADD,DSN=CEE.SCEERUN2,VOL=volser
- SETPROG APF,ADD,DSN=SYS1.LINKLIB,VOL=volser

Requisite LINKLIST and LPA definitions

The RSE server is a z/OS UNIX process that requires access to MVS load libraries. The following (prerequisite) libraries must be made available, either through STEPLIB or LINKLIST/LPALIB:

- System load library
 - SYS1.LINKLIB

- Language Environment's runtime
 - CEE.SCEERUN
 - CEE.SCEERUN2
- C++'s DLL class library
 - CBC.SCLBDLL
- ISPF's TSO/ISPF Client Gateway
 - ISP.SISPLoad
 - ISP.SISPLPA

LINKLIST data sets are defined in SYS1.PARMLIB(PROGxx), if your site followed IBM recommendations. LPA data sets are defined in SYS1.PARMLIB(LPALSTxx).

If you opt to use STEPLIB, you must define the libraries not available through LINKLIST/LPALIB in the STEPLIB directive of rsed.envvars, the RSE configuration file. Be aware, however, that:

- Using STEPLIB in z/OS UNIX has a negative performance impact.
- If one STEPLIB library is APF authorized, then all must be authorized. Libraries lose their APF authorization when they are mixed with non-authorized libraries in STEPLIB.
- Libraries added to the STEPLIB DD in a JCL are not propagated to the z/OS UNIX processes started by the JCL.

PROCLIB changes

The started task and remote build procedures listed below must reside in a system procedure library defined to your JES subsystem. In the instructions below, the IBM default procedure library, SYS1.PROCLIB, is used.

JMON, JES Job Monitor started task

Customize the sample started task member FEK.#CUST.PROCLIB(JMON), as described within the member, and copy it to SYS1.PROCLIB. As shown in the code sample below, you have to provide the following:

- The high-level qualifier of the load library, default FEK
- The JES Job Monitor configuration file, default FEK.#CUST.PARMLIB(FEJJCNFG)

```

/**
/** JES JOB MONITOR
/**
/**JMON      PROC PRM=,                * PRM='-TV' TO START TRACING
/**          LEPRM='RPTOPTS(ON)',
/**          HLQ=FEK,
/**          CFG=FEK.#CUST.PARMLIB(FEJJCNFG)
/**
/**JMON      EXEC PGM=FEJJMON,REGION=0M,TIME=NOLIMIT,
/**          PARM=('&LEPRM,ENVAR("_CEE_ENVFILE_S=DD:ENVIRON")/&PRM')
/**STEPLIB DD DISP=SHR,DSN=&HLQ..SFKEAUTH
/**ENVIRON DD DISP=SHR,DSN=&CFG
/**SYSPRINT DD SYSOUT=*
/**SYSOUT DD SYSOUT=*
/**          PEND
/**

```

Figure 1. JMON - JES Job Monitor started task

RSED, RSE daemon started task

Customize the sample started task member FEK.#CUST.PROCLIB(RSED), as described within the member, and copy it to SYS1.PROCLIB. As shown in the code sample below, you have to provide the following:

- The home directory where Developer for System z is installed, default /usr/lpp/rdz
- The location of the configuration files, default /etc/rdz

```
/*  
/* RSE DAEMON  
/*  
//RSED      PROC IVP=,                * 'IVP' to do an IVP test  
//          PORT=,  
//          CNFG='/etc/rdz',  
//          HOME='/usr/lpp/rdz'  
/*  
//RSED      EXEC PGM=BPXBATSL,REGION=0M,TIME=NOLIMIT,  
// PARM='PGM &HOME./bin/rsed.sh &IVP -C&CNFG -P&PORT'  
//STDOUT    DD SYSOUT=*  
//STDERR    DD SYSOUT=*  
//          PEND  
/*
```

Figure 2. RSED - RSE daemon started task

ELAXF* remote build procedures

Developer for System z provides sample JCL procedures that can be used for the JCL generation, remote project builds, and remote syntax check features of CICS BMS maps, IMS™ MFS screens, and COBOL, PL/I, Assembler and C/C++ programs. These procedures allow installations to apply their own standards, and ensure that developers use the same procedures with the same compiler options and compiler levels.

The sample procedures and their function are listed in Table 3.

Table 3. Sample ELAXF* procedures

Member	Purpose
ELAXFADT	Sample procedure for assembling and debugging High Level assembler programs
ELAXFASM	Sample procedure for assembling High Level assembler programs
ELAXFBMS	Sample procedure for creating CICS BMS object and corresponding copy, dsect, or include member
ELAXFCOC	Sample procedure for doing COBOL Compiles, Integrated CICS translate and integrated DB2® translate
ELAXFCOP	Sample procedure for doing DB2 preprocess of EXEC SQL statements embedded in COBOL programs
ELAXFCOT	Sample procedure for doing CICS translation for EXEC CICS statements embedded in COBOL programs
ELAXFCPC	Sample procedure for doing C compiles
ELAXFCPP	Sample procedure for doing C++ compiles
ELAXFCP1	Sample procedure for COBOL compiles with SCM preprocessor statements -INC and ++INCLUDE
ELAXFDCL	Sample procedure for running a program in TSO mode

Table 3. Sample ELAXF* procedures (continued)

Member	Purpose
ELAXFGO	Sample procedure for the GO step
ELAXFLNK	Sample procedure for linking C/C++, COBOL, PLI and High Level Assembler programs
ELAXFMFS	Sample procedure for creating IMS MFS screens
ELAXFPLP	Sample procedure for doing DB2 preprocess of EXEC SQL statements embedded in PLI programs
ELAXFPLT	Sample procedure for doing CICS translation of EXEC CICS statements embedded in PLI programs
ELAXFPL1	Sample procedure for doing PL/I compiles, integrated CICS translate, and integrated DB2 translate
ELAXFPP1	Sample procedure for PL/I compiles with SCM preprocessor statements -INC and ++INCLUDE
ELAXFSP	Sample procedure to register a stored procedure to DB2.
ELAXFSQL	Sample procedure to invoke SQL.
ELAXFTSO	Sample procedure for running and debugging generated DB2 code in TSO mode
ELAXFUOP	Sample procedure for generating the UOPT step when building programs that run in CICS or IMS subsystems

Customize the FEK.#CUST.PROCLIB(ELAXF*) sample build procedure members, as described within the members, and copy them to SYS1.PROCLIB. Provide the correct high-level qualifiers for different product libraries, as described in Table 4.

Table 4. ELAXF* high-level qualifier checklist

Product	Default HLQ	Value
Rational Developer for System z	FEK	
CICS	CICSTS42.CICS	
DB2	DSNA10	
IMS	IMS	
COBOL	IGY.V4R2M0	
PL/I	IBMZ.V4R2M0	
C/C++	CBC	
LE	CEE	
system LINKLIB	SYS1	
system MACLIB	SYS1	

Some ELAXF* procedures reference data set names that do not have fixed low level qualifiers. An example is the DB2 run library, which holds DB2 utilities that are compiled by your DB2 administrator. Use Table 5 on page 12 to map the default data set names to the names used at your site.

Table 5. ELAXF*. fully qualified data set checklist

Product	Default DSN	Value
Developer for System z- SQL samples	FEK.#CUST.SQL	
DB2 run library	DSNA10.RUNLIB.LOAD	

If the ELAXF* procedures cannot be copied into a system procedure library, ask the Developer for System z users to add a JCLLIB card immediately after the JOB card to the job properties on the client.

```
//MYJOB    JOB <job parameters>
//PROCS    JCLLIB ORDER=(FEK.#CUST.PROCLIB)
```

Creating the security definitions

To create the security definitions for Developer for System z, customize and submit the FEKRACF sample member in the FEK.#CUST.JCL data set. The user submitting this job must have security administrator privileges, such as being RACF[®] SPECIAL. For more details, see Chapter 4, “Security definitions,” on page 21.

Note:

- For those sites that use CA ACF2 TM for z/OS or CA Top Secret[®] for z/OS, see the product page on the CA support site (<https://support.ca.com>) and check for the related Developer for System z Knowledge Document. This Knowledge Document has details on the security commands that are necessary to properly configure Developer for System z.
- The sample FEKRACF job holds more than just RACF commands. The last step of the security definitions consists of making a z/OS UNIX file program controlled. Depending on the policies at your site, this step might be a task for the system programmer and not the security administrator.

FEJJC�FG, JES Job Monitor configuration file

JES Job Monitor (JMON) provides all JES-related services. The behavior of JES Job Monitor can be controlled with the definitions in FEJJC�FG.

FEJJC�FG is located in FEK.#CUST.PARMLIB, unless a different location was specified when the FEK.SFEKSAMP (FEKSETUP) job was customized and submitted. For more details, see “Customization setup” on page 7.

Customize the FEJJC�FG sample JES Job Monitor configuration member, as shown in the following code sample:


```

SERV_PORT=6715
TZ=EST5EDT
#APPLID=FEKAPPL
#AUTHMETHOD=SAF
#CODEPAGE=UTF-8
#CONCHAR=$
#CONSOLE_NAME=JMON
#GEN_CONSOLE_NAME=OFF
#HOST_CODEPAGE=IBM-1047
#LIMIT_COMMANDS=NOLIMIT
#LIMIT_CONSOLE=LIMITED
#LIMIT_VIEW=USERID
#LISTEN_QUEUE_LENGTH=5
#LOOPBACK_ONLY=ON
#MAX_DATASETS=32
#MAX_THREADS=200
#TIMEOUT=3600
#TIMEOUT_INTERVAL=1200
#TRACE_STORAGE=OFF
#SEARCHALL=OFF
#SUBMIT_TIMEOUT=30
#SUBMITMETHOD=TSO
#TSO_TEMPLATE=FEK.#CUST.CNTL(FEJTS0)

```

Figure 3. FEJJCNFG - JES Job Monitor configuration file

SERV_PORT

The port number for JES Job Monitor host server. The default port is 6715, which can be changed. This value must match the port number that is set for JES Job Monitor in the `rsed.envvars` configuration file. If these values differ, RSE cannot connect the client to JES Job Monitor.

TZ Time zone selector. The default is EST5EDT. The default time zone is UTC +5 hours (Eastern Standard Time (EST) Eastern Daylight Savings Time (EDT)). Change this value to represent your time zone.

rsed.envvars, RSE configuration file

The RSE server processes (RSE daemon, RSE thread pool, and RSE server) use the definitions in `rsed.envvars`. Optional Developer for System z and third-party services can use this configuration file also to define environment variables for their use.

`rsed.envvars` is located in `/etc/rdz/`, unless you specified a different location when you customized and submitted job `FEK.SFEKSAMP(FEKSETUP)`. See “Customization setup” on page 7 for more details. You can edit the file with the TSO **OEDIT** command.

See the following sample `rsed.envvars` file, which must be customized to match your system environment.

```

#####
# (1) required definitions
JAVA_HOME=/usr/lpp/java/J6.0
RSE_HOME=/usr/lpp/rdz
_RSE_RSED_PORT=4035
_RSE_JMON_PORT=6715
RSE_HLQ=FEK
_RSE_HOST_CODEPAGE=IBM-1047
TZ=EST5EDT
LANG=C
PATH=/bin:/usr/sbin
_CEE_DMPTARG=/tmp
STEPLIB=NONE
_RSE_JAVAOPTS=""
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Xms128m -Xmx512m"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Ddaemon.log=/var/rdz/logs"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Duser.log=/var/rdz/logs"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DDSTORE_LOG_DIRECTORY="
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Dlog.retention.period=5"
#####
# (2) required definitions for TSO/ISPF Client Gateway
CGI_ISPHOME=/usr/lpp/ispf
CGI_ISPCONF=/etc/rdz
CGI_ISPWORK=/var/rdz
#####
# (3) required definitions for SCLM Developer Toolkit
#####
# (4) optional definitions
# _RSE_PORTRANGE=8108-8118
#####

```

Figure 4. rsed.envvars - RSE configuration file

```

# (5) do not change unless directed by IBM support center
_RSE_SAF_CLASS=/usr/include/java_classes/IRRRacf.jar
_CEE_RUNOPTS="ALL31(ON) HEAP(32M,32K,ANYWHERE,KEEP,,) TRAP(ON)"
_BPX_SHAREAS=YES
_BPX_SPAWN_SCRIPT=YES
_EDC_ADD_ERRNO2=1
JAVA_PROPAGATE=NO
RSE_DSN_SFELKLOAD=$RSE_HLQ.SFELKLOAD
RSE_LIB=$RSE_HOME/lib
PATH=.:$JAVA_HOME/bin:$RSE_HOME/bin:$CGI_ISPHOME/bin:$PATH
LIBPATH=$JAVA_HOME/bin:$JAVA_HOME/bin/classic:$RSE_LIB:$RSE_LIB/icuc
LIBPATH=.:usr/lib:$LIBPATH
CLASSPATH=$RSE_LIB:$RSE_LIB/dstore_core.jar:$RSE_LIB/clientserver.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/dstore_extra_server.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/zosserver.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/dstore_miners.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/universalm miners.jar:$RSE_LIB/mvsminers.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/carma.jar:$RSE_LIB/luceneminer.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/mvsluceneminer.jar:$RSE_LIB/cdzminer.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/mvscdzminer.jar:$RSE_LIB/jesminers.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/mvsutil.jar:$RSE_LIB/jesutils.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/lucene-core-2.3.2.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/cdtparser.jar:$RSE_LIB/wdzBidi.jar
CLASSPATH=$CLASSPATH:$RSE_SAF_CLASS
CLASSPATH=.:$CLASSPATH
_RSE_PTC=$RSE_LDAP_PTC_GROUP_SUFFIX
_RSE_ISPF_OPTS="&SESSION=SPAWN$RSE_ISPF_OPTS"
_RSE_JAVAOPTS="$RSE_JAVAOPTS -Dldap.server.address=$RSE_LDAP_SERVER"
_RSE_JAVAOPTS="$RSE_JAVAOPTS -Dldap.server.port=$RSE_LDAP_PORT"
_RSE_JAVAOPTS="$RSE_JAVAOPTS -Dldap.ptc.group.name.suffix=$RSE_PTC"
_RSE_JAVAOPTS="$RSE_JAVAOPTS -DISPF_OPTS='$RSE_ISPF_OPTS'"
_RSE_JAVAOPTS="$RSE_JAVAOPTS -DA_PLUGIN_PATH=$RSE_LIB"
_RSE_JAVAOPTS="$RSE_JAVAOPTS -Xbootclasspath/p:$RSE_LIB/bidiTools.jar"
_RSE_JAVAOPTS="$RSE_JAVAOPTS -Dfile.encoding=$RSE_HOST_CODEPAGE"
_RSE_JAVAOPTS="$RSE_JAVAOPTS -Dconsole.encoding=$RSE_HOST_CODEPAGE"
_RSE_JAVAOPTS="$RSE_JAVAOPTS -DDSTORE_INITIAL_SIZE=0"
_RSE_JAVAOPTS="$RSE_JAVAOPTS -DDSTORE_MAX_FREE=0"
_RSE_JAVAOPTS="$RSE_JAVAOPTS -DDSTORE_SPIRIT_ON=false"
_RSE_JAVAOPTS="$RSE_JAVAOPTS -DSPIRIT_EXPIRY_TIME=90"
_RSE_JAVAOPTS="$RSE_JAVAOPTS -DSPIRIT_INTERVAL_TIME=6"
_RSE_JAVAOPTS="$RSE_JAVAOPTS -Dcom.ibm.cacheLocalHost=true"
_RSE_JAVAOPTS="$RSE_JAVAOPTS -Duser.home=$HOME"
_RSE_JAVAOPTS="$RSE_JAVAOPTS -Dclient.username=$RSE_USER_ID"
_RSE_JAVAOPTS="$RSE_JAVAOPTS -Dlow.heap.usage.ratio=15"
_RSE_JAVAOPTS="$RSE_JAVAOPTS -Dmaximum.heap.usage.ratio=40"
_RSE_JAVAOPTS="$RSE_JAVAOPTS -DDSTORE_KEEPA_LIVE_ENABLED=true"
_RSE_JAVAOPTS="$RSE_JAVAOPTS -DDSTORE_KEEPA_LIVE_RESPONSE_TIMEOUT=60000"
_RSE_JAVAOPTS="$RSE_JAVAOPTS -DDSTORE_IO_SOCKET_READ_TIMEOUT=180000"
_RSE_JAVAOPTS="$RSE_JAVAOPTS -DRSECOMM_LOGFILE_MAX=0"
_RSE_JAVAOPTS="$RSE_JAVAOPTS -Djob.monitor.port=$RSE_JMON_PORT"
_RSE_JAVAOPTS="$RSE_JAVAOPTS -Dlock.info.timeout=10000"
_RSE_JAVAOPTS="$RSE_JAVAOPTS -showversion"
_RSE_SERVER_CLASS=org.eclipse.dstore.core.server.Server
_RSE_DAEMON_CLASS=com.ibm.etools.zos.server.RseDaemon
_RSE_POOL_SERVER_CLASS=com.ibm.etools.zos.server.ThreadPoolProcess
_RSE_SERVER_TIMEOUT=120000
_SCLMDT_BASE_HOME=$RSE_HOME
_SCLMDT_WORK_HOME=$CMDSE_RV_WORK_HOME
CGI_DTWORK=$_SCLMDT_WORK_HOME
CMDSE_RV_BASE_HOME=$CGI_ISPHOME
CMDSE_RV_CONF_HOME=$CGI_ISPCONF
CMDSE_RV_WORK_HOME=$CGI_ISPWORK
# =====
# (6) additional environment variables

```

Figure 5. rsed.envvars - RSE configuration file (continued)

JAVA_HOME

Java home directory. The default is /usr/lpp/java/J6.0. Change to match your Java installation.

RSE_HOME

RSE home directory. The default is /usr/lpp/rdz. Change to match your Developer for System z installation.

_RSE_RSED_PORT

RSE daemon port number. The default is 4035. Can be changed if desired.

_RSE_JMON_PORT

JES Job Monitor port number. The default is 6715. Can be changed if desired. This value must match the port number set for JES Job Monitor in the FEJJCNG configuration file. If these values differ, RSE cannot connect the client to JES Job Monitor.

_RSE_HOST_CODEPAGE

The host codepage. The default is IBM-1047. Change to match your host codepage.

TZ Time zone selector. The default is EST5EDT. The default time zone is UTC +5 hours (Eastern Standard Time (EST) Eastern Daylight Savings Time (EDT)). Change to match your time zone.

LANG

Specifies the name of the default locale. The default is C. C specifies the POSIX locale and (for example) Ja_JP specifies the Japanese locale. Change to match your locale.

STEPLIB

Access MVS data sets not in LINKLIST/LPALIB. The default is NONE.

You can bypass the need of having (prerequisite) libraries in LINKLIST/LPALIB by uncommenting and customizing one or more of the following STEPLIB directives:

```
STEPLIB=$STEPLIB:CEE.SCEERUN:CEE.SCEERUN2:CBC.SCLBDLL
STEPLIB=$STEPLIB:ISP.SISPLoad:ISP.SISPLPA:SYS1.LINKLIB
```

Note:

- Using STEPLIB in z/OS UNIX has a negative performance impact.
- If one STEPLIB library is APF authorized, then all must be authorized. Libraries lose their APF authorization when they are mixed with non-authorized libraries in STEPLIB.
- Libraries that are designed for LPA placement might require additional program control or APF authorizations if they are accessed through LINKLIST or STEPLIB.
- Coding a STEPLIB DD statement in the server JCL does not set the requested STEPLIB concatenation.

_RSE_JVAOPTS="\$_RSE_JVAOPTS -Ddaemon.log=/var/rdz/logs"

Directory holding the RSE daemon and server logging and RSE audit data. The default is /var/rdz/logs. Change to enforce the desired location. If this directive is commented out, the home directory of the user ID assigned to RSE daemon will be used. The home directory is defined in the OMVS security segment of the user ID.

_RSE_JVAOPTS="\$_RSE_JVAOPTS -Duser.log=/var/rdz/logs"

Directory leading to the user-specific logs. The default is /var/rdz/logs. Change to enforce the desired location. If this directive is commented out,

the home directory of the client user ID will be used. The directory path is defined in the OMVS security segment of the user ID.

Note: The complete path to the user logs is `userlog/dstorelog/$LOGNAME/`, where `userlog` is the value of the `user.log` directive, `dstorelog` is the value of the `DSTORE_LOG_DIRECTORY` directive and `$LOGNAME` is the client's user ID in uppercase.

_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Dlog.retention.period=5"

Number of days daemon and user logs are kept. The default is 5.

Customize this directive to delete the logs after a given number of days.

Specify 0 to set no limit. The maximum value is 365. Note that daemon log cleanup happens at the next action that requires daemon activity. User logs are cleaned up the next time the user connects.

CGI_ISPHOME

Home directory for the ISPF code that provides the TSO/ISPF Client Gateway service. The default is `/usr/lpp/ispf`. Change to match your ISPF installation.

CGI_ISPCONF

ISPF base configuration directory. The default is `/etc/rdz`. Change to match the location of `ISPF.conf`, the TSO/ISPF Client Gateway customization file.

CGI_ISPWORK

ISPF base work directory. The default is `/var/rdz`. Change to match the location of the `WORKAREA` directory used by the TSO/ISPF Client Gateway.

_RSE_PORTRANGE

Specifies the port range that the RSE server can open for communication with a client. Any port can be used by default. This is an optional directive.

ISPF.conf, ISPF's TSO/ISPF Client Gateway configuration file

ISPF's TSO/ISPF Client Gateway uses the definitions in `ISPF.conf` to create a valid environment to execute batch TSO and ISPF commands. Developer for System z uses this environment to run some MVS based services.

`ISPF.conf` is located in `/etc/rdz/`, unless you specified a different location when you customized and submitted job `FEK.SFEKSAMP(FEKSETUP)`. See "Customization setup" on page 7 for more details. You can edit the file with the TSO **OEDIT** command.

Comment lines start with an asterisk (*). When concatenating data set names, add them on the same line and separate the names with a comma (,).

Besides providing the correct names for the ISPF data sets, you must also add the TSO Commands service data set name, `FEK.SFEKPROC`, to the `SYSPROC` or `SYSEXEC` statement, as shown in the following code sample.

```
* REQUIRED:
sysproc=ISP.SISPCLIB,FEK.SFEKPROC
ispmlib=ISP.SISPMENU
isptlib=ISP.SISPTENU
ispplib=ISP.SISPPENU
ispslib=ISP.SISPSLIB
ispllib=ISP.SISPLoad

* OPTIONAL:
*allocjob = ISP.SISPSAMP(ISPZISP2)
*ISPF_timeout = 900
```

Figure 6. ISPF.conf - ISPF configuration file

Note: You can add your own DD-like statements and data set concatenations to customize the TSO environment, thus mimicking a TSO logon procedure.

Chapter 3. Installation verification

This chapter summarizes the installation verification information in *Rational Developer for System z Host Configuration Guide* (SC23-7658). For more details, see that publication.

Verifying the started tasks

JMON, JES Job Monitor

Start the JMON started task or user job. The startup information in DD STDOUT should end with the following message:

```
FEJ211I Server ready to accept connections.
```

If the job ends with return code 66, FEK.SFEKAUTH is not APF authorized.

RSED, RSE daemon

Start the RSED started task or user job with the IVP=IVP parameter. With this parameter, the server ends after doing some installation verification tests. The output of these tests is available in DD STDOUT. In case of certain errors, data also is available in DD STDERR.

Note: Start the RSE daemon, without the IVP parameter, before continuing with the other IVP tests. RSE daemon issues the following console message upon successful startup:

```
FEK002I RseDaemon started. (port=4035)
```

IVP operator commands

PassTicket reusability

Developer for System z requires that the PassTickets it generates are reusable, because PassTicket generation is limited to one for each user every second. Verify the PassTicket reusability by executing the following operator command. Replace userid with a valid TSO user ID.

```
MODIFY RSED,APPL=IVP PASSTICKET,userid
```

RSE daemon connection

Verify the RSE daemon connection by executing the following command. Replace userid with a valid TSO user ID.

```
MODIFY RSED,APPL=IVP DAEMON,userid
```

ISPF Client Gateway

Verify the ISPF Client Gateway connection by executing the following command. Replace userid with a valid TSO user ID.

```
MODIFY RSED,APPL=IVP ISPF,userid
```

Chapter 4. Security definitions

Customize and submit the FEKRACF sample member in the FEK.#CUST.JCL data set, which has sample RACF and z/OS UNIX commands to create the basic security definitions for Developer for System z.

Note: For those sites that use CA ACF2™ for z/OS or CA Top Secret® for z/OS, see the product page on the CA support site (<https://support.ca.com>) and check for the related Developer for System z Knowledge Document. This Knowledge Document has details on the security commands that are necessary to properly configure Developer for System z.

To complete the security setup, the security administrator must know the values listed in Table 6. These values were defined during the previous steps of the installation and customization of Developer for System z.

Table 6. Security setup variables

Description	<ul style="list-style-type: none">• Default value• Where to find the answer	Value
Developer for System z product high-level qualifier	<ul style="list-style-type: none">• FEK• SMP/E installation	
Developer for System z customization high-level qualifier	<ul style="list-style-type: none">• FEK.#CUST• FEK.SFEKSAMP(FEKSETUP), as described in “Customization setup” on page 7	
JES Job Monitor started task name	<ul style="list-style-type: none">• JMON• FEK.#CUST.PROCLIB(JMON), as described in “PROCLIB changes” on page 9	
RSE daemon started task name	<ul style="list-style-type: none">• RSED• FEK.#CUST.PROCLIB(RSED), as described in “PROCLIB changes” on page 9	

Activate the security settings and classes

Developer for System z utilizes a variety of security mechanisms to ensure a secure and controlled host environment for the client. To do so, several classes and security settings must be active, as shown with the following sample RACF commands:

- Display current settings
 - SETROPTS LIST
- Activate facility class for z/OS UNIX and digital certificate profiles
 - SETROPTS GENERIC(FACILITY)

```

-
  SETROPTS CLASSACT(FACILITY) RACLIST(FACILITY)
• Activate started task definitions
-
  SETROPTS GENERIC(STARTED)
-
  RDEFINE STARTED ** STDATA(USER(=MEMBER) GROUP(STCGROUP) TRACE(YES))
-
  SETROPTS CLASSACT(STARTED) RACLIST(STARTED)
• Activate console security for JES Job Monitor
-
  SETROPTS GENERIC(CONSOLE)
-
  SETROPTS CLASSACT(CONSOLE) RACLIST(CONSOLE)
• Activate operator command protection for JES Job Monitor
-
  SETROPTS GENERIC(OPERCMDS)
-
  SETROPTS CLASSACT(OPERCMDS) RACLIST(OPERCMDS)
• Activate application protection for RSE
-
  SETROPTS GENERIC(APPL)
-
  SETROPTS CLASSACT(APPL) RACLIST(APPL)
• Activate secured signon using PassTickets for RSE
-
  SETROPTS GENERIC(PTKTDATA)
-
  SETROPTS CLASSACT(PTKTDATA) RACLIST(PTKTDATA)
• Activate program control to ensure that only trusted code can be loaded by RSE
-
  RDEFINE PROGRAM ** ADDMEM('SYS1.CMDLIB'//NOPADCHK) UACC(READ)
-
  SETROPTS WHEN(PROGRAM)

```

Note: Do not create the ** profile if you already have a * profile in the PROGRAM class. The ** profile obscures and complicates the search path used by your security software. If a * profile exists, merge the existing * and the new ** definitions. IBM recommends to use the ** profile, as documented in *Security Server RACF Security Administrator's Guide (SA22-7683)*.

Attention: Some products, such as FTP, require to be program controlled if "WHEN PROGRAM" is active. Test this before activating it on a production system.

Define an OMVS segment for Developer for System z users

An RACF OMVS segment (or equivalent) that specifies a valid non-zero uid, home directory, and shell command must be defined for each user of Developer for System z. Their default group also requires an OMVS segment with a group ID.

- ```
ALTUSER #userid
OMVS(UID(#user-identifier) HOME(/u/#userid) PROGRAM(/bin/sh) NOASSIZEMAX)
```
- ```
ALTGROUP #group-name OMVS(GID(#group-identifier))
```

Define the Developer for System z started tasks

The following sample RACF commands create the JMON and RSED started tasks, with the STCJMON and STCRSE protected user IDs, and the STCGROUP group assigned to them. Replace the #group-id and #user-id-* variables with valid OMVS IDs.

- ```
ADDGROUP STCGROUP OMVS(GID(#group-id))
DATA('GROUP WITH OMVS SEGMENT FOR STARTED TASKS')
```
- ```
ADDUSER STCJMON DFLTGRP(STCGROUP) NOPASSWORD NAME('RDZ - JES JOBMONITOR')  
OMVS(UID(#user-id-jmon) HOME(/tmp) PROGRAM(/bin/sh)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
```
- ```
ADDUSER STCRSE DFLTGRP(STCGROUP) NOPASSWORD NAME('RDZ - RSE
DAEMON')
OMVS(UID(#user-id-rse) HOME(/tmp) PROGRAM(/bin/sh)
ASSIZEMAX(2147483647)
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
```
- ```
RDEFINE STARTED JMON.* DATA('RDZ - JES JOBMONITOR')  
STDATA(USER(STCJMON) GROUP(STCGROUP) TRUSTED(NO))
```
- ```
RDEFINE STARTED RSED.* DATA('RDZ - RSE DAEMON')
STDATA(USER(STCRSE) GROUP(STCGROUP) TRUSTED(NO))
```
- ```
SETROPTS RACLIST(STARTED) REFRESH
```

Note:

- Ensure that the started tasks user IDs are protected by specifying the NOPASSWORD keyword.
- Ensure that RSE server has a unique OMVS uid, which is required due to the z/OS UNIX related privileges granted to this uid.

Define RSE as a secure z/OS UNIX server

RSE requires UPDATE access to the BPX.SERVER profile to create and delete the security environment for the client's thread. If this profile is not defined, UID(0) is required for RSE.

- ```
RDEFINE FACILITY BPX.SERVER UACC(NONE)
```
- ```
PERMIT BPX.SERVER CLASS(FACILITY) ACCESS(UPDATE) ID(STCRSE)
```
- ```
SETROPTS RACLIST(FACILITY) REFRESH
```

**Attention:** Defining the BPX.SERVER profile makes z/OS UNIX as a whole switch from UNIX level security to z/OS UNIX level security, which is more secure. This switch might impact other z/OS UNIX applications and operations. Test the security before activating the profile on a production system.

---

## Define the MVS program controlled libraries for RSE

Servers with authority to BPX.SERVER must run in a clean, program-controlled environment. This requirement implies that all programs called by the RSE server must also be program controlled. For MVS load libraries, program control is managed by your security software.

- RALTER PROGRAM \*\* UACC(READ) ADDMEM('SYS1.LINKLIB'//NOPADCHK)
- RALTER PROGRAM \*\* UACC(READ) ADDMEM('CEE.SCEERUN'//NOPADCHK)
- RALTER PROGRAM \*\* UACC(READ) ADDMEM('CEE.SCEERUN2'//NOPADCHK)
- RALTER PROGRAM \*\* UACC(READ) ADDMEM('ISP.SISPLOAD'//NOPADCHK)
- SETROPTS WHEN(PROGRAM) REFRESH

**Note:** Do not use the \*\* profile if you already have a \* profile in the PROGRAM class. A \* profile obscures and complicates the search path used by your security software. If a \* profile exists, merge the existing \* and the new \*\* definitions. IBM recommends using the \*\* profile, as documented in *Security Server RACF Security Administrator's Guide* (SA22-7683).

---

## Define PassTicket support for RSE

The client's password or other means of identification, such as an X.509 certificate is used only to verify the identity upon connection. After a connection is established, PassTickets are used to maintain thread security. PassTickets are system generated passwords with a lifespan of about 10 minutes. The generated PassTickets are based upon a secret key. This key is a 64-bit number (16 hex characters). Replace in the sample RACF commands below the key16 variable with a user-supplied 16 character hex string (characters 0-9 and A-F).

**Note:** If the PTKTDATA class is already defined, before creating the listed profiles, verify that it is defined as a generic class. The support for generic characters in the PTKTDATA class was introduced with the z/OS release 1.7, with the introduction of a Java interface to PassTickets.

- RDEFINE PTKTDATA FEKAPPL UACC(NONE) SSIGNON(KEYMASKED(key16))  
APPLDATA('NO REPLAY PROTECTION – DO NOT CHANGE') DATA('RATIONAL DEVELOPER  
FOR SYSTEM Z')
- RDEFINE PTKTDATA IRRPTAUTH.FEKAPPL.\* UACC(NONE) DATA('RATIONAL DEVELOPER  
FOR SYSTEM Z')
- PERMIT IRRPTAUTH.FEKAPPL.\* CLASS(PTKTDATA) ACCESS(UPDATE) ID(STCRSE)
- SETROPTS RACLIST(PTKTDATA) REFRESH

**Attention:** The client connection request fails if PassTickets are not set up correctly.

---

## Define application protection for RSE

During client logon, RSE daemon verifies that a user is allowed to use the application.

- ```
RDEFINE APPL FEKAPPL UACC(READ)
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
```
- ```
SETROPTS RACLIST(APPL) REFRESH
```

**Note:** The client connection request fails only if the application ID is defined and the user lacks READ access to the profile.

---

## Define JES command security

JES Job Monitor issues all JES operator commands requested by a user through an extended MCS (EMCS) console, whose name is controlled with the `CONSOLE_NAME` directive, as documented in “FEJJCNFG, JES Job Monitor configuration file” on page 12.

The following sample RACF commands give Developer for System z users conditional access to a limited set of JES commands: Hold, Release, Cancel, and Purge. Users have only execution permission if they issue the commands through JES Job monitor. Replace the `#console` variable with the actual console name.

- ```
RDEFINE OPERCMDS MVS.MCSOPER.#console UACC(READ)
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
```
- ```
RDEFINE OPERCMDS JES%.** UACC(NONE)
```
- ```
PERMIT JES%.** CLASS(OPERCMDS) ACCESS(UPDATE) WHEN(CONSOLE(JMON)) ID(*)
```
- ```
SETROPTS RACLIST(OPERCMDS) REFRESH
```

**Note:**

- Usage of the console is permitted if no `MVS.MCSOPER.#console` profile is defined.
- The `CONSOLE` class must be active for `WHEN(CONSOLE(JMON))` to work, but there is no actual profile check in the `CONSOLE` class for EMCS consoles.
- Do not replace `JMON` with the actual console name in the `WHEN(CONSOLE(JMON))` clause. The `JMON` keyword represents the point-of-entry application, not the console name.

**Attention:** Defining JES commands with universal access `NONE` in your security software might impact other applications and operations. Test the impact before activating the universal access on a production system.

---

## Define data set profiles

READ access for users and ALTER for system programmers is sufficient for most Developer for System z data sets.

Protect `FEK.SFEKAUTH` and `FEK.SFEKLPA` against updates because these data sets are APF authorized.

- ADDGROUP (FEK) OWNER(IBMUSER) SUPGROUP(SYS1)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - HLQ STUB')
- ADDSD 'FEK.\*.\*\*\*' UACC(READ)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
- PERMIT 'FEK.\*.\*\*\*' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
- SETROPTS GENERIC(DATASET) REFRESH

---

## Define the z/OS UNIX program controlled files for RSE

Servers with authority to BPX.SERVER must run in a clean, program-controlled environment. This requirement implies that all programs called by the RSE server must also be program controlled. For z/OS UNIX files, program control is managed by the **extattr** command. To execute this command, you need READ access to BPX.FILEATTR.PROGCTL in the FACILITY class, or be UID(0).

- extattr +p /usr/lib/libIRRRacf\*.so

### Note:

- For z/OS 1.9 and later, /usr/lib/libIRRRacf\*.so is installed program controlled during SMP/E RACF install.
- For z/OS 1.10 and later, /usr/lib/libIRRRacf\*.so is part of SAF, which is included with base z/OS, so it is also available to non-RACF customers.
- The setup might be different if you use a security product other than RACF. For more information, consult the documentation of your security product.
- The SMP/E install of Developer for System z sets the program control bit for internal RSE server programs.
- Use the **ls -Eog** z/OS UNIX command to display the current status of the program control bit. The file is program controlled if the second string contains the letter *p*.

```
$ ls -Eog /usr/lib/libIRRRacf*.so
-rwxr-xr-x aps- 2 69632 Oct 5 2007 /usr/lib/libIRRRacf.so
-rwxr-xr-x aps- 2 69632 Oct 5 2007 /usr/lib/libIRRRacf64.so
```

---

## Verify the security settings

Use the following sample commands to display the results of your security-related customizations.

- Security settings and classes
  - SETROPTS LIST
- OMVS segment for users
  - LISTUSER #userid NORACF OMVS
  - LISTGRP #group-name NORACF OMVS
- Started tasks
  - LISTGRP STCGROUP OMVS
  - LISTUSER STCJMON OMVS
  - LISTUSER STCRSE OMVS
  - RLIST STARTED JMON.\* ALL STDATA
  - RLIST STARTED RSED.\* ALL STDATA

- RSE as a secure z/OS UNIX server
  - RLIST FACILITY BPX.SERVER ALL
- MVS program controlled libraries for RSE
  - RLIST PROGRAM \*\* ALL
- PassTicket support for RSE
  - RLIST PTKTDATA FEKAPPL ALL SSIGNON
  - RLIST PTKTDATA IRRPTAUTH.FEKAPPL.\* ALL
- Application protection for RSE
  - RLIST APPL FEKAPPL ALL
- JES command security
  - RLIST CONSOLE JMON ALL
  - RLIST OPERCMDS MVS.MCSOPER.JMON ALL
  - RLIST OPERCMDS JES%.\*\* ALL
- Data set profiles
  - LISTGRP FEK ALL
  - LISTDSD PREFIX(FEK) ALL
- z/OS UNIX program controlled files for RSE
  - ls -E /usr/lib/libIRRRacf\*.so





---

## Chapter 5. Migration considerations

This appendix summarizes the migration information in *Rational Developer for System z Host Configuration Guide* (SC23-7658). For more details, see that publication.

---

### Version 9.0 migration notes

The following migration notes are specific to IBM Rational Developer for System z version 9.0. These notes are valid for migration from IBM Rational Developer for System z version 9.0.0 to version 9.0.1, and they are additions to the existing version 9.0.0 migration notes.

All of the listed changes are valid since version 9.0.1.

#### IBM Rational Developer for System z, FMID HHOP900

- CARMA: The CRADEF VSAM file for the CA Endeavor<sup>®</sup> SCM RAM has been updated.
- CARMA: The CRASTART load module, which resides in LPA, has been updated, requiring an LPA update.
- CARMA: Added support to execute a user exit during CARMA startup.
- CARMA: Added support for RAMs processing startup arguments.
- CARMA: New customizable members have been added:
  - CRAEXIT: Sample CARMA user exit.
  - CRAALLOC: Allocation exec for custom RAM CARMA invocations.
  - CRACFG: CA Endeavor<sup>®</sup> SCM RAM usage configuration file.
- CARMA: The following customizable members have changed:
  - CRASRV.properties
  - crastart.conf
  - crastart.endevor.conf
  - CRASUBMT
  - CRASUBCA
  - CRANDVRA
- CARMA: Additional DD statements are added for the CA Endeavor<sup>®</sup> SCM RAM in crastart.endevor.conf and CRASUBCA:
  - CRAPARM, which is allocated by CRANDVRA
  - CRACFG
- CARMA: Additional DD statements are added for the non-“CA Endeavor<sup>®</sup> SCM RAM” in crastart.conf and CRASUBMT:
  - CRAPARM, which is allocated by CRAALLOC
- Customization: The FEKSETUP JCL now processes the new members:
  - CRACFG: copied to FEK.#CUST.PARMLIB(CRACFG)
  - AQESTC: copied to FEK.#CUST.PROCLIB(DBGMGR)
  - AQECSD: copied to FEK.#CUST.JCL(AQECSD)
- Integrated Debugger: New optional service

- IEASVCxx, LPALSTxx and PROGxx (APF and LINKLIST) parmlib updates
- DBGMGR: started task JCL
- AQECSD: sample JCL to update CICS CSD
- AQERACF: sample JCL to do security setup for just Integrated Debugger
- RSE: Updated PROCLIB members
  - ELAXFGO
- RSE: New optional directives have been added to rsecomm.properties:
  - USER
- RSE: New operator commands
  - F rsed,APPL=TRACE {USER | SERVER | CLEAR}
- RSE: New optional directives have been added to rsed.envvars:
  - (\_RSE\_JAVAOPTS) -Dsearch.server.limit.timeout
  - (\_RSE\_JAVAOPTS) -Dkeep.all.logs
  - (\_RSE\_JAVAOPTS) -Daudit.users
  - RSE\_UBLD\_DD
  - RSE\_UBLD\_STEPLIB
- zUnit: New optional startup arguments have been added:
  - CLOCALE / -l

**Note:** To simplify migration from an existing Developer for System z setup without the Integrated Debugger, sample JCL FEK.SFEKSAMP(AQERACF) with RACF commands is provided to define just the Integrated Debugger related security definitions.

## IBM Rational Developer for System z Host Utilities, FMID HAKG900

There are no version 9.0 specific migration notes for this product.

---

### Migrate from version 8.5 to version 9.0

These notes are for a migration from a base version 8.5 to version 9.0. It includes changes that are already documented as part of version 8.5 maintenance. The changes that are part of the maintenance stream, and thus possibly already implemented, are marked with the release where they were introduced.

## IBM Rational Developer for System z, FMID HHOP900

- The default SMP/E install location for MVS and z/OS UNIX components did not change and thus remain FEK.\* and /usr/lpp/rdz/.\*.
- CARMA: The CRADEF and CRASTRS VSAM files for the CA Endeavor® SCM RAM must be updated to use the new support for customizable CA Endeavor® SCM batch-actions (since version 8.5.1).
- CARMA: Added support to disable a RAM during CRADEF VSAM creation (since version 8.5.1).
- CARMA: Added support for non-absolute file references in CRASRV.properties (since version 8.5.1).
- CARMA: New sample members have been added:
  - CRABJOBC: Default JOB card for CA Endeavor® SCM batch-actions (since version 8.5.1).

- CARMA: The following customizable members have changed:
  - CRASRV.properties (since version 8.5.1)
  - carma.startup.rex (since version 8.5.1)
  - CRA\$VCAD (since version 8.5.1)
  - CRA\$VDEF (since version 8.5.1)
  - CRABATCA (since version 8.5.1)
  - CRABCFG (since version 8.5.1)
  - CRANDVRA (since version 8.5.1)
- CARMA: Additional DD statements are added for the CA Endevor® SCM RAM in crastart.endevor.conf and CRASUBCA:
  - CRABJCLO, which is allocated by CRANDVRA (since version 8.5.1)
  - ENHCEDIT, which is allocated by CRANDVRA (since version 8.5.1)
- Customization: The FEKSETUP JCL now processes the new members:
  - CRABJOBC: copied to FEK.#CUST.CNTL(CRABJOBC) (since version 8.5.1)
  - ELAXFSP: copied to FEK.#CUST.PROCLIB(ELAXFSP) (since version 9.0)
  - ELAXFSQL: copied to FEK.#CUST.PROCLIB(ELAXFSQL) (since version 9.0)
  - FEKTEP2: copied to FEK.#CUST.SQL(FEKTEP2) (since version 9.0)
  - FEKTIAD: copied to FEK.#CUST.SQL(FEKTEP2) (since version 9.0)
- Fault Analyzer Integration: support for FAI has been discontinued. This change is incompatible with older clients still using FAI.
- JES Job Monitor - New operator commands have been added to the JMON started task:
  - MODIFY USERS (since version 8.5.1)
  - MODIFY -T{N | E | I | V} (since version 8.5.1)
  - MODIFY -M{N | E | W | I | V} (since version 8.5.1)
  - MODIFY TRACE {N | E | I | V} (since version 9.0)
  - MODIFY MESSAGE {N | E | W | I | V} (since version 9.0)
- JES Job Monitor - New optional directives have been added to FEJJCENFG:
  - LOOPBACK\_ONLY (since version 9.0)
- JES Job Monitor - Optional directives have been removed from FEJJCENFG:
  - \_BPXK\_SETIBMOPT\_TRANSPORT (since version 9.0)
- Problem determination: The FEKLOGS JCL now supports specifying multiple user IDs for gathering user logs (since version 8.5.1).
- Problem determination: The FEKLOGS JCL now uses DD REFORMAT to collect logs reformatted for quicker problem determination (since version 8.5.1).
- Problem determination : The following customizable members have changed:
  - FEKLOGS (since version 8.5.1)
- RSE - New operator commands have been added to the RSED started task:
  - MODIFY DISPLAY OWNER,DATASET=dataset (since version 9.0)
  - MODIFY DEBUG GC,PID=pid (since version 9.0)
- RSE: New non-customizable directives have been added to rsed.envvars:
  - \_CMDSERV\_BASE\_HOME (since version 8.5.1)
  - \_CMDSERV\_CONF\_HOME (since version 8.5.1)
  - \_CMDSERV\_WORK\_HOME (since version 8.5.1)
  - RSE\_DSN\_SFELLOAD (since version 9.0)
  - (\_RSE\_JAVAOPTS) -Dlock.info.timeout (since version 9.0)

- (\_RSE\_JAVAOPTS) -DDSTORE\_INITIAL\_SIZE (since version 9.0)
- (\_RSE\_JAVAOPTS) -DDSTORE\_MAX\_FREE (since version 9.0)
- RSE: New required directives have been added to `rsed.envvars`:
  - RSE\_HLQ (since version 9.0)
- RSE: New optional directives have been added to `rsed.envvars`:
  - (\_RSE\_JAVAOPTS) -DRSE\_DSICALL (since version 8.5.1)
  - (\_RSE\_JAVAOPTS) -DDISABLE\_REMOTE\_INDEX\_SEARCH (since version 8.5.1)
  - (\_RSE\_JAVAOPTS) -DDISABLE\_TEXT\_SEARCH (since version 9.0)
  - (\_RSE\_JAVAOPTS) -Dsearch.server.limit.hits (since version 9.0)
  - (\_RSE\_JAVAOPTS) -Dsearch.server.limit.datasets (since version 9.0)
  - (\_RSE\_JAVAOPTS) -Dsearch.server.limit.lines (since version 9.0)
  - (\_RSE\_JAVAOPTS) -DDSTORE\_SSL\_ALGORITHM (since version 9.0)
- RSE: The default value for non-customizable directives in `rsed.envvars` has changed:
  - (\_RSE\_JAVAOPTS) -DSPIRIT\_EXPIRY\_TIME (since version 9.0)
- RSE: The default value for optional directives in `rsed.envvars` has changed:
  - (\_RSE\_JAVAOPTS) -Xms (since version 8.5.1)
  - (\_RSE\_JAVAOPTS) -Xmx (since version 8.5.1)
  - (\_RSE\_JAVAOPTS) -Dmaximum.clients (since version 8.5.1)
  - (\_RSE\_JAVAOPTS) -Dmaximum.threads (since version 8.5.1)
  - CGI\_ISPPREF (since version 9.0)
- Security: Support for new security profiles has been added:
  - FEK.USR.\*\* (since version 8.5.1)

## IBM Rational Developer for System z Host Utilities, FMID HAKG900

There are no migration notes, because there is no equivalent function in version 8.5.

## Chapter 6. Operator commands

This chapter provides an overview of the available operator (or console) commands for Developer for System z.

### Modify (F)

The **MODIFY** command can be used to dynamically query and change the characteristics of an active task. The abbreviated version of the command is the letter F.

## JES Job Monitor

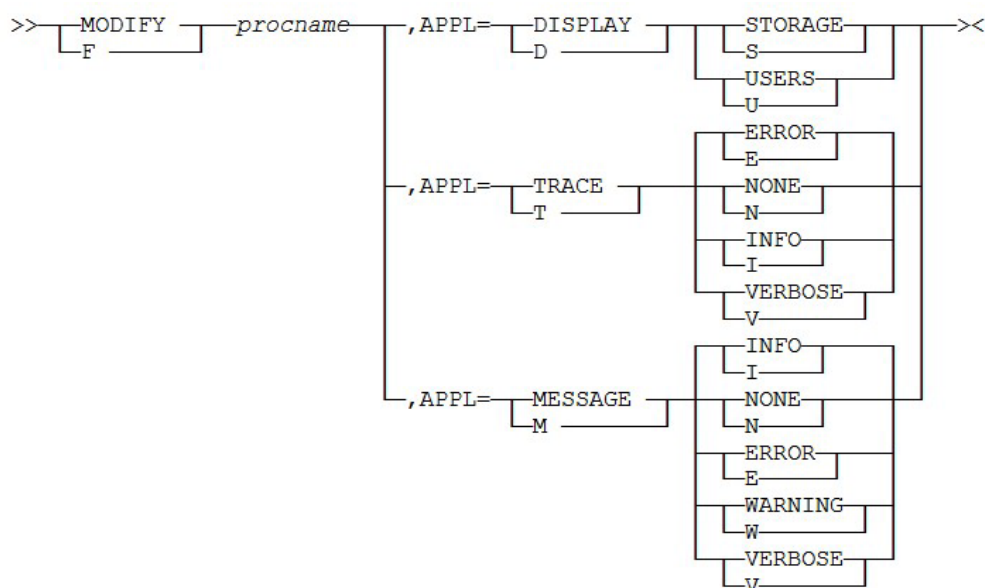


Figure 7. MODIFY JMON operator command

**procname**

The name of the member in a procedure library that was used to start the server. The default name used during the host system configuration is JMON.

## DISPLAY STORAGE

Write a storage usage report to DD SYSOUT. A message "JMON storage information written to SYSOUT" is written to the console with message ID BPXM023I. The storage usage report shows various storage-related fields with sizes in bytes, kilobytes, and megabytes.

```
>>>STORAGE TRACE (console request)<<<
LDAREGRQ 00000000000 00000000K 00000M requested region size
 below 16M line
LDASIZA 00006266880 00006120K 00005M maximum region size
LDALIMIT 00006266880 00006120K 00005M limit
LDAVVRG 00006266880 00006120K 00005M getmain limit
LDALOAL 00000006144 000000060K 00000M in use
LDAHIAL 000000266240 00000260K 00000M LSQA/SWA/private subpools
GAP 00000000000 00000000K 00000M gaps in allocation
```

```

_AVAIL 00005939200 00005800K 00005M available (including gaps)
_MAX 00006000640 00005860K 00005M current limit
 above 16M line
LDAESIZA 01905262592 01860608K 01817M maximum region size
LDAELIM 01905262592 01860608K 01817M limit
LDAEVVRG 01905262592 01860608K 01817M getmain limit
LDAELOAL 00000937984 00000916K 00000M in use
LDAEHIAL 00012754944 00012456K 00012M ELSQA/ESWA/private subpools
_EGAP 00000000000 00000000K 00000M gaps in allocation
_EAVAIL 01891569664 01847236K 01803M available (including gaps)
_EMAX 01892507648 01848152K 01804M current limit

```

## DISPLAY USERS

Write a list of active users to DD SYSOUT. A message "JMON user list written to SYSOUT" is written to the console with message ID BPXM023I. The user list shows various user-related data, including CPU usage.

```

S0 userid USER 4:04(elapsed) 4:04(idle)
Users: 1

```

## TRACE {NONE | ERROR | INFO | VERBOSE}

Control the detail level of the JES Job Monitor trace log (DD SYSOUT). The default is E (Error). A message "JMON TRACE LEVEL:{NONE | ERROR | INFO | VERBOSE}" is written to the console with message ID BPXM023I.

|              |                                                     |
|--------------|-----------------------------------------------------|
| N or NONE    | Startup messages only                               |
| E or ERROR   | Startup and Error messages only (default)           |
| I or INFO    | Startup, Error, and Informational messages          |
| V or VERBOSE | Startup, Error, Informational, and Verbose messages |

Detailed tracing will cause performance degradations and should only be done under the direction of the IBM support center.

## MESSAGE {NONE | ERROR | WARNING | INFO | VERBOSE}

Control the detail level of the JES Job Monitor message log (DD SYSPRINT). The default is I (Informational). A message "JMON MESSAGE LEVEL:{NONE | ERROR | WARNING | INFO | VERBOSE}" is written to the console with message ID BPXM023I.

|              |                                                      |
|--------------|------------------------------------------------------|
| N or NONE    | No messages.                                         |
| E or ERROR   | Error messages only                                  |
| W or WARNING | Error and Warning messages                           |
| I or INFO    | Error, Warning, and Informational messages (default) |
| V or VERBOSE | Error Warning, Informational, and Verbose messages   |

Detailed tracing will cause performance degradations and should only be done under the direction of the IBM support center.

## RSE daemon

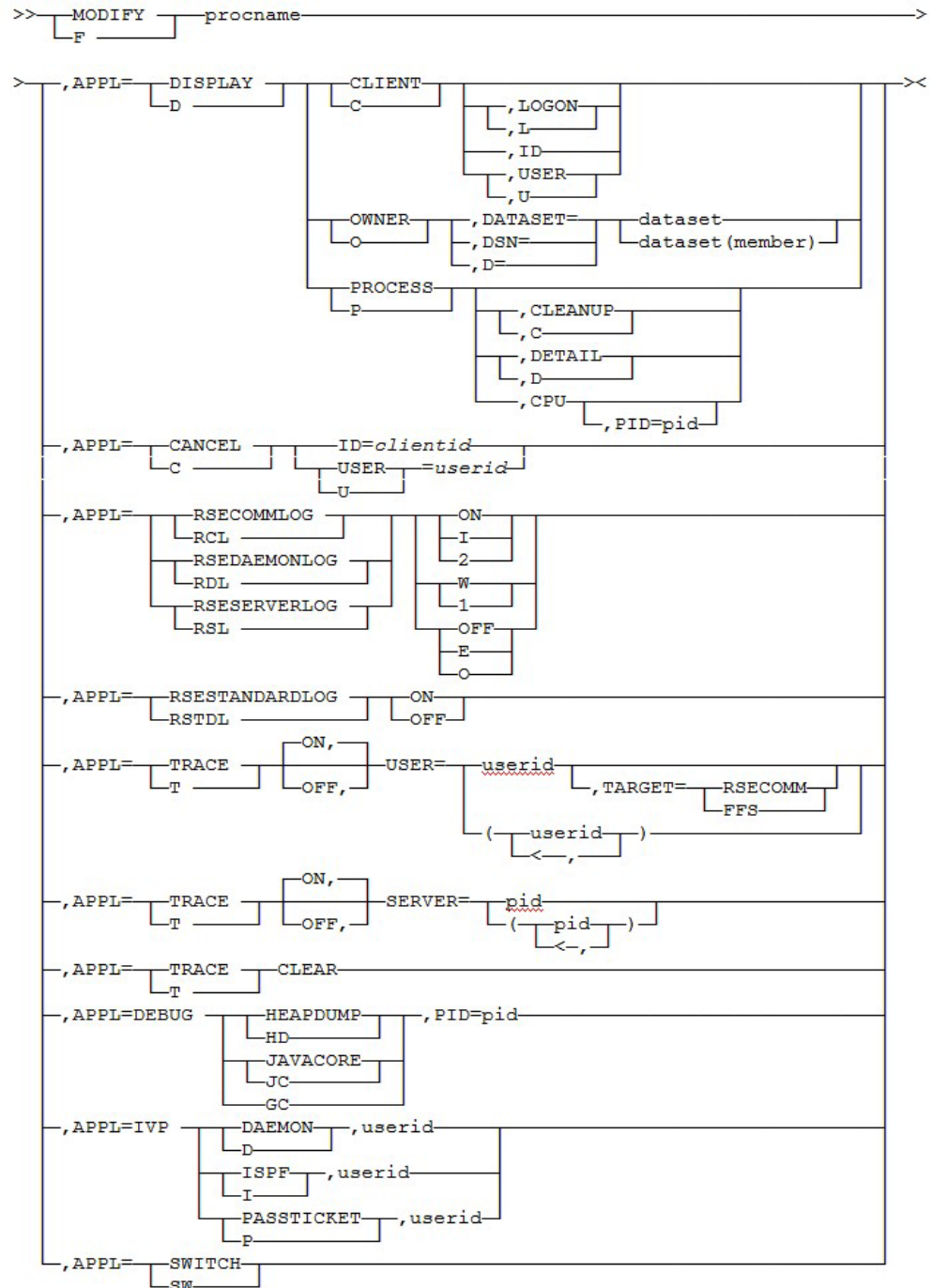


Figure 8. MODIFY RSED operator command

### procname

The name of the member in a procedure library that was used to start the server. The default name used during the host system configuration is RSED.

### DISPLAY CLIENT[{,LOGON | ,ID | ,USER}]

Display the active clients in a single BPXM023I message. The result layout



depends on the command option that was used. You can change the sorting order with the optional command arguments.

- No command option: Clients are grouped by the thread pool process that serves them.

```
ProcessId(<processid>) ASId(<asid>) JobName(<jobname>)
Clients(<local>/<total>) Order(<startup order>)
<clientid><userid><connected since>
```

- LOGON command option: Clients are ordered by logon time.

```
LOGON TIME----- ID----- USERID--
<connected since> <clientid> <userid>
```

- ID command option: Clients are ordered by client ID.

```
ID----- USERID-- LOGON TIME-----
<clientid> <userid> <connected since>
```

- USER command option: Clients are ordered by user ID.

```
USERID-- ID----- LOGON TIME-----
<userid> <clientid> <connected since>
```

#### **DISPLAY OWNER,DATASET={dataset | dataset(member)}**

Display the data set enqueue owner in a single BPXM023I message.

```
FEK217I <dataset[(member)]> is locked by <userid>
FEK218I <dataset[(member)]> is not locked
FEK219E Failed to determine lock owner for <dataset[(member)]>
```

- The server also reports the locks held by other products, such as ISPF.
- The **D GRS,RES=(\*,dataset)** operator command is unable to tell which Developer for System z user is the actual enqueue holder, all it can tell you is the threadpool in which the user is active.

#### **DISPLAY PROCESS[{,CLEANUP | ,CPU [,PID=pid] | ,DETAIL}]**

Display the RSE thread pool processes in one or more BPXM023I messages. There can be multiple processes, which are used for load balancing the connected users.

```
ProcessId(<processid>) Memory Usage(<java heap usage>%)
Clients(<number of clients>) Order(<startup order>) <error status>
```

#### **Note:**

- <processid> can be used in process-specific z/OS UNIX operator commands.
- Each process has its own Java heap, whose size can be set in `rsed.envvars`. The reported Java heap usage includes storage that is released by Developer for System z, but which is not yet freed by the Java garbage collection process.
- <startup order> is a sequential number that indicates the order that the thread pools were started. The number corresponds to the number used in the filename of the `stderr.*.log` and `stdout.*.log` files.

In normal situations, <error status> is blank. Table 7 on page 37 documents the possible non-blank values for <error status>.



Table 7. Thread pool error status

| Status           | Description                                                                                                                                                                                                                                                                                                 |
|------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| *severe error*   | The thread pool process encountered an unrecoverable error and halted operations. The other status fields show the last known values. To remove this entry from the table, use the CLEANUP option of the <b>DISPLAY PROCESS</b> modify command.                                                             |
| *killed process* | The thread pool process was killed by Java, z/OS UNIX or an operator command. The other status fields show the last known values. To remove this entry from the table, use the CLEANUP option of the <b>DISPLAY PROCESS</b> modify command.                                                                 |
| *timeout*        | The thread pool process did not respond in a timely manner to RSE daemon during a client connect request. The other status fields show the current values. The thread pool is excluded for future client connect requests. The *timeout* status is reset when a client served by this thread pool logs off. |

More information is provided when the DETAIL option of the **DISPLAY PROCESS** modify command is used:

```
ProcessId(33555087) ASId(002E) JobName(RSED8) Order(1)
PROCESS LIMITS: CURRENT HIGHWATER LIMIT
JAVA HEAP USAGE(%) 10 56 100
CLIENTS 0 25 30
MAXFILEPROC 83 103 64000
MAXPROCUSER 97 99 200
MAXTHREADS 9 14 1500
MAXTHREADTASKS 9 14 1500
```

The ASId field is the address space ID, in hexadecimal notation. The process limits table shows the current resource usage, the high-water mark for the resource usage, and the resource limit. Due to other limiting factors, the defined limit might never be reached.

The CPU option of the **DISPLAY PROCESS** modify command shows the accumulated CPU usage, in milliseconds, of each thread in a thread pool. Every thread pool has a BPXM023I message. By default, all thread pools report the CPU usage, but you can limit the scope to a single thread pool by specifying PID=pid on the operator command, where pid is the process ID of the target thread pool.

```
ProcessId(421) ASId(007D) JobName(RSED8) Order(1)
USERID THREAD-ID TCB@ ACC_TIME TAG
STCRSE 0EDE54000000000 005E6B60 822 1/ThreadPoolProcess
STCRSE 0EDE87000000000 005E69C8 001
STCRSE 0EDE98000000000 005E6518 1814
STCRSE 0EDEBA000000000 005E66B0 2305
STCRSE 0EDECB000000000 005E62F8 001
STCRSE 0EEDC0000000000 005E60D8 001
STCRSE 0EDF86000000000 005C2BF8 628 6/ThreadPoolMonitor$Memory
UsageMonitor
STCRSE 0EDF97000000000 005C2D90 003 7/ThreadPoolMonitor
STCRSE 0EDFDB000000000 005C29D8 001
STCRSE 0EE22E000000000 005C1BE0 070
IBMUSER 0EE0EB000000000 005C22B8 276 20/ServerReceiver
IBMUSER 0EE250000000000 005C19C0 137 16/ServerUpdateHandler
```

```

IBMUSER 0EE2610000000013 005C17A0 509 15/ServerCommandHandler
IBMUSER 0EE1840000000014 005C1E00 065 21/ZosSystemMiner
STCRSE 0EE1510000000016 005C2098 078
STCRSE 0EE1950000000017 005C1580 001
IBMUSER 0EE23F0000000018 005C1360 021 26/UniversalFileSystemMine
r
IBMUSER 0EE2A5000000001C 005C0CF0 003 27/EnvironmentMiner
IBMUSER 0EE283000000001D 005C1140 002 31/CommandMiner
IBMUSER 0EE272000000001E 005C0E88 081 32/MVSFileSystemMiner
IBMUSER 0EE294000000001F 005C0AD0 002 33/MVSByteStreamHandler$Op
enCloseThread
STCRSE 0EE2E90000000023 005C0470 001
IBMUSER 0EE2C70000000024 005C08B0 050 38/JESMiner
IBMUSER 0EE2B60000000026 005C0690 004 40/FAMiner
IBMUSER 0EE30B0000000027 005C0250 002 41/LuceneMiner
IBMUSER 0EE31C0000000028 005C0030 002 42/CDTParserMiner
IBMUSER 0EE32D0000000029 005BDE00 002 43/MVSLuceneMiner
IBMUSER 0EE33E000000002A 005BDBE0 002 44/CDTMVSParserMiner

```

If the output size exceeds the maximum number of lines for a console message, the output is split over multiple BPXM023I messages. These additional messages have the same header as the first message, but with the CONTINUATION keyword added to the first line.

```

ProcessId(421) ASId(007D) JobName(RSED8) Order(1) CONTINUATION
USERID THREAD-ID TCB# ACC_TIME TAG

```

The output is limited to the first 4000 threads for each thread pool.

#### **CANCEL ID=clientid**

Cancel a client connection based on the client ID, which is shown in the **DISPLAY CLIENT** modify command.

When a client connection is cancelled, the host system threads go through normal termination processing to clean up resources used by them. This action implies that some threads can take a few minutes before they end; for example, because they are waiting on the keep-alive mechanism to time out.

#### **CANCEL USER=userid**

Cancel a client connection based on the client's user ID, which is shown in the **DISPLAY CLIENT** modify command.

When a client connection is cancelled, the host system threads go through normal termination processing to clean up resources used by them. This action implies that some threads can take a few minutes before they end; for example, because they are waiting on the keep-alive mechanism to time out.

#### **RSECOMMLOG {ON | OFF | I | W | E | 2 | 1 | 0}**

Control the trace detail level for RSE server (rsecomm.log) and the MVS data set services (lock.log and ffs\*.log). The startup default is defined in rsecomm.properties. Three detail levels are available:

|               |                                                                                |
|---------------|--------------------------------------------------------------------------------|
| E or 0 or OFF | Error messages only.                                                           |
| W or 1        | Error and warning messages. This is the default setting in rsecomm.properties. |
| I or 2 or ON  | Error, warning, and informational messages.                                    |

Detailed tracing will cause performance degradations and should only be done under the direction of the IBM support center.

### **RSEDAEMONLOG {ON | OFF | I | W | E | 2 | 1 | 0}**

Control the trace detail level for RSE daemon (rsedaemon.log). The startup default is defined in rsecomm.properties. There are three detail levels available:

|               |                                                                                |
|---------------|--------------------------------------------------------------------------------|
| E or 0 or OFF | Error messages only.                                                           |
| W or 1        | Error and warning messages. This is the default setting in rsecomm.properties. |
| I or 2 or ON  | Error, warning, and informational messages.                                    |

Detailed tracing will cause performance degradations and should only be done under the direction of the IBM support center.

### **RSESERVERLOG {ON | OFF | I | W | E | 2 | 1 | 0}**

Control the trace detail level for RSE thread pools (rseserver.log). The startup default is defined in rsecomm.properties. Three detail levels are available:

|               |                                                                                |
|---------------|--------------------------------------------------------------------------------|
| E or 0 or OFF | Error messages only.                                                           |
| W or 1        | Error and warning messages. This is the default setting in rsecomm.properties. |
| I or 2 or ON  | Error, warning, and informational messages.                                    |

Detailed tracing will cause performance degradations and should only be done under the direction of the IBM support center.

### **RSESTANDARDLOG {ON | OFF}**

Disable (OFF) or enable (ON) the updating of the log files that hold the stdout and stderr streams of the stdout\*.log and stderr\*.log thread pools. The startup default is defined by the enable.standard.log directive in rsed.envvars.

Detailed tracing will cause performance degradations and should only be done under the direction of the IBM support center.

### **TRACE [{ON, | OFF,}]USER=userid[,TARGET={FFS | RSECOMM}]**

Enable (ON) or disable (OFF) tracing for the specified user IDs. The default is ON. This setting overrides the default setting controlled by the **MODIFY RSECOMMLOG** operator command. Two detail levels are available:

|              |                                             |
|--------------|---------------------------------------------|
| OFF          | Error messages only                         |
| ON (default) | Error, warning, and informational messages. |

The command alters the trace detail level for RSE server (rsecomm.log) and the MVS data set services (lock.log and ffs\*.log). This can be limited with the TARGET keyword, which accepts two values:

|         |                                                                                    |
|---------|------------------------------------------------------------------------------------|
| FFS     | Set the specified log level only for MVS data set services (lock.log and ffs*.log) |
| RSECOMM | Set the specified log level only for RSE server (rsecomm.log)                      |

The command can be issued for users that are currently not logged on. The setting remains active when a user logs off and will be used again when the user logs on.

Use the **USER** directive in `rsecomm.properties` to simulate issuing the **MODIFY TRACE USER** command at server startup. Existing settings from previous **MODIFY TRACE USER** or **MODIFY TRACE SERVER** operator commands or the **USER** directive in `rsecomm.properties` will be replaced by the setting of this command.

Detailed tracing will cause performance degradations and should only be done under the direction of the IBM support center.

#### **TRACE [{ON, | OFF,}]USER=(userid,userid,...)**

Enable (ON) or disable (OFF) tracing for the specified user IDs. The default is ON. This setting overrides the default setting controlled by the **MODIFY RSECOMMLOG** operator command. Two detail levels are available:

|              |                                             |
|--------------|---------------------------------------------|
| OFF          | Error messages only.                        |
| ON (default) | Error, warning, and informational messages. |

The command alters the trace detail level for RSE server (`rsecomm.log`) and the MVS data set services (`lock.log` and `ffs*.log`). The command can be issued for users that are currently not logged on. The setting remains active when a user logs off and will be used again when the user logs on. Use the **USER** directive in `rsecomm.properties` to simulate issuing the **MODIFY TRACE USER** command at server startup. Existing settings from previous **MODIFY TRACE USER** or **MODIFY TRACE SERVER** operator commands or the **USER** directive in `rsecomm.properties` will be replaced by the setting of this command.

Detailed tracing will cause performance degradations and should only be done under the direction of the IBM support center.

#### **TRACE [{ON, | OFF,}] SERVER={pid | (pid,pid,...)}**

Enable (ON) or disable (OFF) tracing for all users in the specified thread pool where `pid` is the process ID of an RSE thread pool. The default is ON. This setting overrides the default setting controlled by the **MODIFY RSECOMMLOG** operator command. Two detail levels are available:

|              |                                             |
|--------------|---------------------------------------------|
| OFF          | Error messages only.                        |
| ON (default) | Error, warning, and informational messages. |

The command alters the trace detail level for RSE server (`rsecomm.log`) and the MVS data set services (`lock.log` and `ffs*.log`). Existing settings from previous **MODIFY TRACE USER** or **MODIFY TRACE SERVER** operator commands or the **USER** directive in `rsecomm.properties` will be replaced by the setting of this command.

Detailed tracing will cause performance degradations and should only be done under the direction of the IBM support center.

#### **TRACE CLEAR**

Remove all trace overrides set by the **MODIFY TRACE USER** and **MODIFY TRACE SERVER** operator commands and the **USER** directive in `rsecomm.properties`.

#### **DEBUG HEAPDUMP,PID=pid**

Request a Java Heap dump for a specified thread pool, where `pid` is the process ID of an RSE thread pool. The dump is written to the directory specified by `_CEE_DUMPTARG` in `rsed.envvars`, where the default value is `/tmp`. Results are shown in a single BPXM023I console message.

JVMDUMP034I User requested Heap dump using '/tmp/heapdump.20120223.211'  
430.16777590.0001.phd' through JVMRI

#### **DEBUG JAVACORE,PID=pid**

Request a Java Core dump for a specified thread pool, where pid is the process ID of an RSE thread pool. The dump is written to the directory specified by \_CEE\_DUMPTARG in rsed.envvars, where the default value is /tmp. Results are shown in a single BPXM023I console message.

JVMDUMP034I User requested Java dump using '/tmp/javacore.20120223.214  
244.16777590.0002.phd' through JVMRI

#### **DEBUG GC,PID=pid**

Request a Java Garbage Collection for a specified thread pool, where pid is the process ID of an RSE thread pool.

#### **IVP DAEMON,userid**

Log user ID userid on to RSE daemon to do a connection test. Results are shown with one or more FEK900I console messages. The return code is shown with console message FEK901I.

```
+FEK900I DAEMON IVP: SSL is disabled
+FEK900I DAEMON IVP: connected
+FEK900I DAEMON IVP: 1977
+FEK900I DAEMON IVP: 6902918
+FEK900I DAEMON IVP: Success
+FEK901I DAEMON IVP Exit code = 0
```

#### **Note:**

- The function is similar to what the fekfivpd IVP (Installation Verification Program) does.
- RSE daemon generates a PassTicket which is used as password for the IVP, so there is no Write To Operator with Reply (WTOR) requesting a password.

#### **IVP ISPF,userid**

Invoke ISPF's Client Gateway as user ID userid. Results are shown with one or more FEK900I console messages. The return code is shown with console message FEK901I.

```
+FEK900I ISPF IVP: executed on CDFMVS08 -- Tue Sep 13 22:29:28 EDT 2011
+FEK900I ISPF IVP: executed by uid=1(IBMUSER) gid=0(SYS1)
+FEK900I ISPF IVP: using /etc/rdz/rsed.envvars
+FEK900I ISPF IVP: current address space size limit is 2147483647
(2048.0 MB)
+FEK900I ISPF IVP: maximum address space size limit is 2147483647
(2048.0 MB)
+FEK900I ISPF IVP: -----

+FEK900I ISPF IVP: /etc/rdz/ISPF.conf content:
+FEK900I ISPF IVP: -----

+FEK900I ISPF IVP: ispllib=ISP.SISPLOAD
+FEK900I ISPF IVP: ispmllib=ISP.SISPMENU
+FEK900I ISPF IVP: isptlib=ISP.SISPTENU
+FEK900I ISPF IVP: isplib=ISP.SISPPENU
+FEK900I ISPF IVP: ispslib=ISP.SISPSLIB
+FEK900I ISPF IVP: sysproc=ISP.SISPCLIB,FEK.SFEKPROC
+FEK900I ISPF IVP: -----

+FEK900I ISPF IVP: Host install verification for RSE
+FEK900I ISPF IVP: Review IVP log messages from HOST below :
+FEK900I ISPF IVP: -----

+FEK900I ISPF IVP: Service level 22Feb2011
```

```

+FEK900I ISPF IVP: RSE connection and base TSO/ISPF session initializati
on check only
+FEK900I ISPF IVP: *** CHECK : ENVIRONMENT VARIABLES - key variables
displayed below :
+FEK900I ISPF IVP: Server PATH = ./usr/lpp/java/J6.0/bin:/usr/l
pp/rdz/bin:/usr/lpp/ispf/bin:/bin:/usr/sbin
+FEK900I ISPF IVP: STEPLIB = NONE
+FEK900I ISPF IVP: Temporary directory = /tmp
+FEK900I ISPF IVP: CGI_ISPHOME = /usr/lpp/ispf
+FEK900I ISPF IVP: CGI_ISPCONF = /etc/rdz
+FEK900I ISPF IVP: CGI_ISPWORK = /var/rdz
+FEK900I ISPF IVP: -----
+FEK900I ISPF IVP: *** CHECK : USS MODULES
+FEK900I ISPF IVP: Checking ISPF Directory : /usr/lpp/ispf
+FEK900I ISPF IVP: Checking modules in /usr/lpp/ispf/bin directory
+FEK900I ISPF IVP: Checking for ISPF configuration file ISPF.conf
+FEK900I ISPF IVP: RC=0
+FEK900I ISPF IVP: MSG: SUCCESSFUL
+FEK900I ISPF IVP: -----
+FEK900I ISPF IVP: *** CHECK : TSO/ISPF INITIALIZATION
+FEK900I ISPF IVP: (TSO/ISPF session will be initialized)
+FEK900I ISPF IVP: RC=0
+FEK900I ISPF IVP: MSG: SUCCESSFUL
+FEK900I ISPF IVP: -----
+FEK900I ISPF IVP: *** CHECK: Shutting down TSO/ISPF IVP session
+FEK900I ISPF IVP: RC=0
+FEK900I ISPF IVP: MSG: SUCCESSFUL
+FEK900I ISPF IVP: -----
+FEK900I ISPF IVP: Host installation verification completed successfully
+FEK900I ISPF IVP: -----
+FEK901I ISPF IVP Exit code = 0

```

**Note:**

- The function is similar to what the fekfivpi IVP (Installation Verification Program) does.
- RSE daemon generates a PassTicket which is used as password for the IVP, so there is no Write To Operator with Reply (WTOR) requesting a password.

**IVP PASSTICKET,userid**

Test the reusability of a PassTicket generated for user ID userid. Results are shown with one or more FEK900I console messages. The return code is shown with console message FEK901I.

```

+FEK900I PASSTICKET IVP: the default applid=FEKAPPL
+FEK900I PASSTICKET IVP: Success, PassTicket IVP finished normally
+FEK901I PASSTICKET IVP Exit code = 0

```

**Note:**

- When using RACF as security product, reusable PassTickets require the “NO REPLAY PROTECTION” keyword in the security definitions.
- There is no equivalent IVP (Installation Verification Program) for this test. Starting the RSE daemon with the IVP=IVP argument calls a PassTicket IVP that tests PassTicket generation, but it cannot test PassTicket reusability.

- RSE daemon generates a PassTicket which is used as password for the IVP, so there is no Write To Operator with Reply (WTOR) requesting a password.

**SWITCH**

Switch to a new audit log file.





---

## Chapter 7. Optional customization

This section summarizes the CARMA, Application Deployment Manager, SCLM Developer Toolkit, and other customization task information in *Rational Developer for System z Host Configuration Guide* (SC23-7658). For more details, see that publication.

---

### (Optional) Common Access Repository Manager (CARMA)

You will need the assistance of a security administrator and a TCP/IP administrator to complete this customization task, which requires the following resources or special customization tasks:

- (Optional) TCP/IP port range for internal communication
- (Optional) Security rule to allow developers update to CARMA VSAM files
- (Optional) Security rule to allow users to submit CRA\* jobs
- (Optional) LPA update

Common Access Repository Manager (CARMA) is a server platform for Repository Access Managers (RAMs). A RAM is an Application Programming Interface (API) for a z/OS based Software Configuration Manager (SCM). By wrapping the SCM functionality in a RAM, a single API is available for a client to access any supported SCM.

Developer for System z provides multiple pre-built RAMs, as well as source code examples for creating your own RAM.

The IBM Rational Developer for System z Interface for CA Endeavor® Software Configuration Manager gives Developer for System z clients direct access to CA Endeavor® SCM.

---

### (Optional) SCLM Developer Toolkit

You will need assistance of an SCLM administrator and optionally a security administrator to complete this customization task, which requires the following resources and/or special customization tasks:

- APF and LINKLIST updates
- Define SCLM language translators for Java EE support
- Define SCLM types for Java EE support
- (Optional) Security rule to allow users update to an SCLM VSAM
- (Optional) Installation of Ant

SCLM Developer Toolkit provides the tools needed to extend the capabilities of SCLM to the client. SCLM (Software Configuration and Library Manager) itself is a host-based source code manager that is shipped as part of ISPF.

The SCLM Developer Toolkit has an Eclipse-based plug-in that interfaces to SCLM and provides for access to all SCLM processes for legacy code development as well as support for full Java and Java EE development on the workstation with synchronization to SCLM on the mainframe including building, assembling, and deployment of the Java EE code from the mainframe.

---

## **(Optional) Application Deployment Manager (deprecated)**

You will need assistance of a CICS administrator, a TCP/IP administrator and a security administrator to complete this customization task, which requires the following resources or special customization tasks:

- TCP/IP port for external communication
- Update CICS region JCL
- Update CICS region CSD
- Define group to CICS region
- Security rule to allow administrators update to an Application Deployment Manager VSAM
- CICS TS security setup
- (Optional) Define CICS transaction names
- (Optional) Security rule to allow users update to an Application Deployment Manager VSAM

Developer for System z uses certain functions of Application Deployment Manager as a common deployment approach for various components. Optional customization enables more features of Application Deployment Manager and can add the following services to Developer for System z:

- IBM CICS Explorer provides an Eclipse-based infrastructure to view and manage CICS resources and enables greater integration between CICS tools.
- CICS Resource Definition (CRD) client and server provide the following functions:
  - CICS Resource Definition editor
  - Allow application developers to define CICS resources in a limited, controlled, and secure fashion.
  - Prevent CICS development access to unauthorized or incorrect VSAM data sets by providing the CICS administrator control over the physical data set name attribute in File definitions.
  - Miscellaneous CICS development aids
  - Miscellaneous CICS Web Service development aids

---

## **(Optional) Host-based code analysis**

Similar to the Developer for System z client, the Developer for System z host supports running code analysis tools, which are provided as a separate product, Rational Developer for System z Host Utilities. A benefit of doing code analysis on the host is that it can be integrated in your daily batch processing.

The following code analysis tools are available on the host:

- Code review: Using rules with different severity levels, code review scans source code and reports rule violations.
- Code coverage: Analyze a running program and generate a report of lines that are executed, compared to the total number of executable lines.

---

## **(Optional) pushtoclient.properties, Host-based client control**

This customization task does not require assistance, special resources, or special customization tasks.

Developer for System z clients version 8.0.1 and later pull client configuration files and upgrade information from the host when they connect, ensuring that all clients have common settings and that they are up-to-date.

z/OS Projects can be defined individually through the z/OS Projects perspective on the client or can be defined centrally on the host and propagated to the client of a user, as needed. These "host-based projects" look and function exactly like projects that are defined on the client except that their structure, members, and properties cannot be modified by the client, and they are accessible only when connected to the host.

---

## **(Optional) ssl.properties, RSE SSL encryption**

You will need assistance of a security administrator to complete this customization task, which requires the following resources or special customization tasks:

- LINKLIST update
- Security rule to add program controlled data sets
- (Optional) Security rule to add certificate for SSL

External (client-host) communication can be encrypted using SSL. This feature is disabled by default and is controlled by the settings in `ssl.properties`.

---

## **(Optional) rsecomm.properties, RSE tracing**

This customization task does not require assistance, special resources, or special customization tasks.

Developer for System z supports different levels of tracing the internal program flow for problem solving purposes. RSE, and some of the services called by RSE, use the settings in `rsecomm.properties` to know the desired detail level in the output logs.

---

## **(Optional) include.conf, Forced includes for C/C++ content assist**

This customization task does not require assistance, special resources, or special customization tasks.

Content assist for C/C++ can use the definitions in `include.conf` to do forced includes of specified files or members. A forced include consists of a file or directory, data set, or data set member which are parsed when a content assist operation is performed, regardless of whether that file or member was included in the source code using a preprocessor directive.

---

## **(Optional) z/OS UNIX subprojects**

This customization task does not require assistance, special resources, or special customization tasks.

Remote Execution (REXEC) is a TCP/IP service to let clients execute a command on the host. Secure Shell (SSH) is a similar service, but here all communication is encrypted using Secure Socket Layer (SSL). Developer for System z uses either service for doing remote, host-based actions in z/OS UNIX subprojects.

---

## **(Optional) Include preprocessor support**

This customization task does not require assistance, special resources, or special customization tasks.

Developer for System z supports the interpreting and expanding COBOL and PL/I include statements, including select third-party include statements. Developer for System z also provides a sample REXX exec, FEKRNPLI, that can be called by the Developer for System z client to expand PL/I source by starting the PL/I compiler.

---

## **(Optional) xUnit support for Enterprise COBOL and PL/I**

This customization task does not require assistance, but does require the following resources or special customization tasks: LINKLIST update.

Frameworks that assist developers in writing code to perform repeatable, self-checking unit tests are collectively known as xUnit. Developer for System z provides such a framework for unit testing of Enterprise COBOL and PL/I code, called zUnit.

---

## **(Optional) CICS bidirectional language support**

You will need the assistance of a CICS administrator to complete this customization task, which requires the following resources or special customization tasks:

- Update CICS region JCL
- Define a program to CICS

The Developer for System z Enterprise Service Tools (EST) component supports different formats of Arabic and Hebrew interface messages, as well as bidirectional data presentation and editing in all editors and views. In terminal applications, both left-to-right and right-to-left screens are supported, as well as numeric fields and fields with opposite-to-screen orientation.

Additional bidirectional features and functionality include the following:

- The EST service requestor dynamically specifies bidirectional attributes of interface messages.
- Bidirectional data processing in service flows is based on bidirectional attributes (text type, text orientation, numeric swapping, and symmetric swapping). These attributes can be specified in different stages of flow creation for both interface and terminal flows.
- EST-generated runtime code includes conversion of data between fields in messages that have different bidirectional attributes.

Additionally, EST-generated code can support bidi transformation in environments other than CICS SFR; for example, batch applications. You can make the EST generators to include calls to the bidirectional conversion routines by specifying the appropriate bidi transformation options in the EST generation wizards and linking the generated programs with the appropriate bidirectional conversion library, FEK.SFEKLOAD.

---

## (Optional) Diagnostic IRZ messages for generated code

This customization task does not require assistance, but does require the following resources or special customization tasks:

- LINKLIST update
- CICS region JCL update

The Developer for System z client has a code generation component called Enterprise Service Tools (EST). For code generated by EST to issue diagnostic error messages, all IRZ\* and IIRZ\* modules in the FEK.SFEKLMOD load library must be made available to the generated code.

---

## (Optional) Integrated debugger

You need the assistance of a Security, TCP/IP, and CICS administrator to complete this customization task, which requires the following resources or special customization tasks:

- Add an SVC (requires IPL)
- LPA update for SVC
- APF authorization
- Define started task
- Define security profiles and access lists
- Reserve TCP/IP ports for client-host and host confined communication
- Update CICS region JCL
- Update CICS CSD

The Developer for System z Integrated Debugger host component allows version 9.0.1 and higher clients to debug various Language Environment (LE) based applications, including CICS transactions loaded into read-only memory.

---

## (Optional) DB2 and IMS debug support

This customization task does not require assistance, special resources, or special customization tasks for the Developer for System z configuration. However, there are requirements for the IBM Debug Tool for z/OS configuration.

IBM Debug Tool for z/OS provides a customized Language Environment (LE) user exit named CEEBXITA, which returns the TEST runtime options when called by the LE initialization logic in IMS and DB2 Stored Procedures. IBM Debug Tool for z/OS also provides the Debug Tool extension for the Problem Determination Tools Common Components server, to create and manage the TEST runtime options data set on the z/OS system. Developer for System z can use and enhance IBM Debug Tool for z/OS's support for managing debug profiles for the IMS and DB2 Stored Procedure runtimes.

---

## (Optional) File Manager support

This customization task does not require assistance, special resources, or special customization tasks for the Developer for System z configuration. However, there are requirements for the IBM File Manager for z/OS configuration.

Some functions, such as unformatted QSAM editing, are part of regular data set handling by Developer for System z. More advanced functions, such as formatted data editing using copybooks or include files, require the IBM File Manager plug-in for Eclipse.

---

## **(Optional) WORKAREA and /tmp cleanup**

This customization task does not require assistance, special resources, or special customization tasks.

ISPF's TSO/ISPF Client Gateway and the SCLM Developer Toolkit function use the WORKAREA and /tmp directories to store temporary work files, which are removed before the session is closed. However, temporary output is sometimes left behind, for example, if there is a communication error while processing. For this reason, the WORKAREA and /tmp directories must be cleared periodically.

---

## Chapter 8. Host Configuration Reference

This section summarizes the information in *Rational Developer for System z Host Configuration Reference* (SC14-7290). For more details, see that publication.

---

### Understanding Developer for System z

The Developer for System z host consists of several components that interact to give the client access to the host services and data. Understanding the design of these components can help you make the correct configuration decisions.

---

### Security considerations

Developer for System z provides mainframe access to users on a non-mainframe workstation. Validating connection requests, providing secure communication between the host and the workstation, and authorizing and auditing activity are, therefore, important aspects of the product configuration.

---

### TCP/IP considerations

Developer for System z uses TCP/IP to provide mainframe access to users on a non-mainframe workstation. It also uses TCP/IP for communication between various components and other products.

---

### WLM considerations

Unlike traditional z/OS applications, Developer for System z is not a monolithic application that can be identified easily to Workload Manager (WLM). Developer for System z consists of several components that interact to give the client access to the host services and data. Some of these services are active in different address spaces, resulting in different WLM classifications.

---

### Tuning considerations

RSE (Remote Systems Explorer) is the core of Developer for System z. To manage the connections and workloads from the clients, RSE is composed of a daemon address space, which controls thread pooling address spaces. The daemon acts as a focal point for connection and management purposes, while the thread pools process the client workloads.

This makes RSE a prime target for tuning the Developer for System z setup. However, maintaining hundreds of users, each using 17 or more threads, a certain amount of storage, and possibly one or more address spaces requires proper configuration of both Developer for System z and z/OS.

---

### Performance considerations

z/OS is a highly customizable operating system. Sometimes, small system changes can have a huge impact on the overall performance. This chapter in the *Rational Developer for System z Host Configuration Reference* highlights some of the changes that can be made to improve the performance of Developer for System z.

---

## Push-to-client considerations

Push-to-client, or host-based client control, supports the central management of these things:

- Client configuration files
- Client product version
- Project definitions

---

## CICSTS considerations

This chapter in the *Rational Developer for System z Host Configuration Reference* contains information useful for a CICS Transaction Server administrator.

---

## User exit considerations

This chapter assists you with enhancing Developer for System z by writing exit routines.

---

## Customizing the TSO environment

This chapter assists you with mimicking a TSO logon procedure by adding DD statements and data sets to the TSO environment in Developer for System z.

---

## Running multiple instances

Sometimes, multiple instances of Developer for System z must be active on the same system; for example, when testing an upgrade. However, some resources such as TCP/IP ports cannot be shared, so the defaults are not always applicable. Use the information in this chapter to plan the coexistence of the different instances of Developer for System z, after which you can use this configuration guide to customize them.

---

## Troubleshooting configuration problems

This chapter is provided to assist you with some common problems that you may encounter during your configuration of Developer for System z, and has the following sections:

- Log and setup analysis using FEKLOGS
- Log files
- Dump files
- Tracing
- z/OS UNIX permission bits
- Reserved TCP/IP ports
- Address Space size
- APPC transaction and TSO Commands service
- Miscellaneous information



---

## Setting up SSL and X.509 authentication

This appendix is provided to assist you with some common problems that you may encounter when setting up Secure Socket Layer (SSL), or during checking or modifying an existing setup. This appendix also provides a sample setup to support users authenticating themselves with an X.509 certificate.

---

## Setting up TCP/IP

This appendix is provided to assist you with some common problems that you may encounter when setting up TCP/IP, or during checking or modifying an existing setup.



---

## Documentation notices for IBM Rational Developer for System Z

© Copyright IBM Corporation 2009, 2013.

U.S. Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

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:

*Intellectual Property Licensing  
Legal and Intellectual Property Law  
IBM Japan, Ltd.  
1623-14, Shimotsuruma, Yamato-shi  
Kanagawa 242-8502 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 websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites 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:

*Intellectual Property Dept. for Rational Software*  
*IBM Corporation*  
*5 Technology Park Drive*  
*Westford, MA 01886*  
*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 document 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.

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. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample 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. 2009, 2013.

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

## **Trademark acknowledgments**

IBM, the IBM logo, and [ibm.com](http://ibm.com) are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

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

Windows is a trademark 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.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Other product and service names might be trademarks of IBM or other companies.

---

## 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. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

---

## Trademark acknowledgments

IBM, the IBM logo, and [ibm.com](http://ibm.com) are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

CA Endeavor is a registered trademark of CA Technologies.

Rational are trademarks of International Business Machines Corporation and Rational Software Corporation, in the United States, other countries, or both.

Intel and Pentium are trademarks of Intel Corporation in the United States, or other countries, or both.

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

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

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

---

# Index

## Special characters

/tmp cleanup 50

## A

Activating IBM Common Access  
Repository Manager 45  
ADM, customization 46  
APF authorizations in PROGxx 8  
Application Deployment Manager,  
customization 46

## B

BPXPRMxx, Set z/OS UNIX limits in 7  
build procedures, ELAXF\* remote 10

## C

C/C++ forced includes 47  
CARMA, activating 45  
Client Gateway configuration file,  
TSO/ISPF 17  
Client user ID 4  
code analysis, host-based 46  
command security, Define JES 25  
commands, JES Job Monitor Modify 33  
commands, RSE daemon Modify 35  
COMMNDxx, Add started tasks to 8  
Common Access Repository Manager,  
Activating 45  
configuration file, FEJJCENFG 12  
configuration, ssl.properties 47  
considerations, Security 21  
controlled libraries for RSE server, Define  
MVS 24  
customization, SCLM Developer  
Toolkit 45

## D

daemon, RSE 10  
data set profiles, Define 25  
DB2 debug support 49  
debug, DB2 and IMS 49  
debugger, integrated 49  
definitions, Security 12  
Deployment and upgrade 5  
Developer for System z started tasks,  
Define 23

## E

ELAXF\* procedures, Sample 10  
ELAXF\* remote build procedures 10  
Enterprise COBOL, xUnit support 48

## F

FEJJCENFG, JES Job Monitor configuration  
file 12  
FMID HAKG900 32  
FMID HHOP900 30

## H

Host based client control 46

## I

IBM Common Access Repository  
Manager, Activating 45  
IMS debug support 49  
include.conf 47  
integrated debugger 49  
ISPF Client Gateway 19  
ISPF.conf 17  
IVP operator commands 19

## J

JES command security, Define 25  
JES Job Monitor 9  
JES Job Monitor configuration file,  
FEJJCENFG 12  
JES Job Monitor, Modify command 33  
JMON 9

## L

libraries for RSE server, Define MVS 24  
LINKLIST and LPA definitions,  
Prerequisite 8  
Logging configuration file,  
rsecomm.properties 47  
LPA definitions, Prerequisite 8

## M

migration, 8.5 to 9.0 30  
MVS program controlled libraries for RSE  
server, Define 24

## O

OMVS segment, Define 23  
operator commands, IVP 19

## P

PARMLIB, changes 7  
PassTicket reusability 19  
PassTicket support for RSE server,  
Define 24  
PL/I, xUnit support 48

Preparation 3  
Prerequisite LINKLIST and LPA  
definitions 8  
Prerequisite software 3  
PROCLIB changes 9  
profiles, Define data set 25  
PROGxx, APF authorizations in 8  
pushtoclient.properties 46

## R

remote build procedures, ELAXF\* 10  
Requisite software 3  
RSE daemon 10  
RSE daemon connection 19  
RSE daemon, Modify command 35  
RSE server, Define as a secure z/OS  
UNIX server 23  
RSE server, Define MVS program  
controlled libraries for 24  
RSE server, Define PassTicket support  
for 24  
RSE server, Define z/OS UNIX program  
controlled files for 26  
RSE server, Prerequisite LINKLIST and  
LPA definitions 8  
RSE SSL configuration, ssl.properties, 47  
RSE trace configuration,  
rsecomm.properties, 47  
rsecomm.properties, 47  
rsed.envvars 36

## S

SCLM Developer Toolkit,  
customization 45  
secure z/OS UNIX server, Define RSE  
server as a 23  
Security considerations 21  
Security definitions 12  
security settings and classes, Activate 21  
security settings, verify 26  
security, Define JES command 25  
segment, Define OMVS 23  
server, Define MVS program controlled  
libraries for RSE 24  
server, Define PassTicket support for  
RSE 24  
server, Define RSE server as a secure  
z/OS UNIX 23  
server, Define z/OS UNIX program  
controlled files for RSE 26  
Servers 4  
settings and classes, Activate security 21  
ssl.properties 47  
started tasks, Define Developer for  
System z 23  
subprojects, z/OS UNIX 47  
support for RSE server, Define  
PassTicket 24

## T

- tasks to COMMNDxx, Add 8
- tasks, Define Developer for System z 23
- trace configuration,
  - rsecomm.properties 47
- TSO/ISPF Client Gateway configuration
  - file 17

## U

- UNIX limits in BPXPRMxx, Set z/OS 7
- UNIX program controlled files for RSE
  - server, Define 26
- UNIX server, Define RSE server as 23
- upgrade, Deployment and 5
- user ID, Client 4

## V

- Verify security settings 26

## W

- WLM considerations 51
- WORKAREA cleanup 50

## X

- xUnit support 48

## Z

- z/OS UNIX limits in BPXPRMxx 7
- z/OS UNIX program controlled files for
  - RSE server, Define 26
- z/OS UNIX server, Define RSE server
  - as 23



---

## Readers' Comments — We'd Like to Hear from You

IBM Rational Developer for System z  
Version 9.0.1  
Host Configuration Quick Start Guide

Publication No. GI11-9201-08

We appreciate your comments about this publication. Please comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. The comments you send should pertain to only the information in this manual or product and the way in which the information is presented.

For technical questions and information about products and prices, please contact your IBM branch office, your IBM business partner, or your authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you. IBM or any other organizations will only use the personal information that you supply to contact you about the issues that you state on this form.

Comments:

Thank you for your support.

Send your comments to the address on the reverse side of this form.

If you would like a response from IBM, please fill in the following information:

\_\_\_\_\_  
Name

\_\_\_\_\_  
Address

\_\_\_\_\_  
Company or Organization

\_\_\_\_\_  
Phone No.

\_\_\_\_\_  
Email address



Cut or Fold  
Along Line

Fold and Tape

Please do not staple

Fold and Tape



NO POSTAGE  
NECESSARY  
IF MAILED IN THE  
UNITED STATES

**BUSINESS REPLY MAIL**

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM  
Corporation  
Building 501  
P.O Box 12195  
Research Triangle Park, NC  
USA 27709-2195



Fold and Tape

Please do not staple

Fold and Tape

Cut or Fold  
Along Line





Printed in USA

GI11-9201-08

