

IBM Rational Developer para System z



# Guía de configuración de host

*Versión 7.1.1*



IBM Rational Developer para System z



# Guía de configuración de host

*Versión 7.1.1*

**Nota**

Antes de utilizar este documento, lea la información general que figura en el apartado “Avisos” en la página 129.

**Segunda edición (diciembre de 2007)**

Esta edición atañe a IBM Rational Developer para System z Versión 7.1.1 (programa número 5724-T07) y a todos los releases y modificaciones ulteriores hasta que se indique lo contrario en nuevas ediciones.

Puede pedir las publicaciones por teléfono o por fax. IBM Software Manufacturing Solutions acepta los pedidos de publicaciones entre las 8:30 de la mañana y las 7:00 de la tarde, hora estándar del este (EST). El número de teléfono es (800) 879-2755. El número de fax es (800) 445-9269. Los faxes deben enviarse a Attn: Publications, 3rd floor.

También puede pedir publicaciones a través de su representante de IBM o de la sucursal de IBM que presta servicio en su localidad. En la dirección que figura más abajo no hay publicaciones almacenadas.

IBM agradece sus comentarios. Puede enviar sus comentarios por correo a la siguiente dirección:

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

Cuando envía información a IBM, otorga a IBM un derecho no exclusivo a utilizar o distribuir la información del modo que IBM considere oportuno sin incurrir por ello en ninguna obligación para con usted.

Nota sobre los derechos restringidos de los usuarios del Gobierno de EE. UU. - El uso, la reproducción o la divulgación están sujetos a las restricciones establecidas en el contrato GSA ADP Schedule Contract con IBM Corp.

© Copyright International Business Machines Corporation 2005, 2007. Reservados todos los derechos.

# Contenido

<b>Figuras</b>	<b>vii</b>
----------------	------------

<b>Tablas</b>	<b>ix</b>
---------------	-----------

<b>Acerca de esta publicación</b>	<b>xi</b>
-----------------------------------	-----------

A quién va dirigida esta publicación	xi
--------------------------------------	----

Publicaciones a las que se hace referencia	xi
--	----

## Capítulo 1. Instalar y configurar los componentes del host . . . . . 1

Consideraciones sobre la preinstalación	2
---	---

Consideraciones sobre la preconfiguración	3
---	---

Configuración necesaria de los productos y software obligatorios.	3
---	---

Consideraciones sobre el ID de usuario	4
--	---

Consideraciones sobre el servidor	5
-----------------------------------	---

Permisos necesarios para implementar las tareas de configuración	6
--	---

Consideraciones sobre el predespliegue	6
--	---

IBM Rational Developer para System z, FMID HHOP710	6
--	---

IBM Gestor de repositorios de acceso común (CARMA), FMID HCMA710	7
--	---

## Capítulo 2. Cambios de instalación y configuración . . . . . 9

Cambios entre la versión 7.0 y la versión 7.1.	9
--	---

IBM Rational Developer para System z, FMID HHOP710	9
--	---

IBM Gestor de repositorios de acceso común (CARMA), FMID HCMA710	9
--	---

Cambios entre la versión 6.0.1 y la versión 7.0.	10
--	----

IBM WebSphere Developer para System z, FMID HHOP700	10
---	----

IBM Gestor de repositorios de acceso común (CARMA), FMID HCMA700.	11
---	----

Hacer copia de seguridad de archivos configurados con anterioridad.	11
---	----

## Capítulo 3. Activar componentes MVS de Developer para System z . . . . . 15

Establecer MAXASSIZE en	
-------------------------	--

SYS1.PARMLIB(BPXPRMxx).	15
-------------------------	----

Biblioteca hlq.SFEKLOAD autorizada por APF	15
--	----

Personalizar el archivo de configuración del supervisor de trabajos JES, FEJJCENFG	16
--	----

Personalizar el JCL de arranque del supervisor de trabajos JES	18
--	----

Rastreo del supervisor de trabajos JES	19
--	----

Ejecutar el supervisor de trabajos JES como una tarea iniciada (STC).	19
---	----

Permisos de servidor	20
----------------------	----

Verificación del JCL de inicio del supervisor de trabajos JES	20
---	----

Acceso a spool JES y seguridad.	21
---------------------------------	----

Acceso a spool condicional	21
----------------------------	----

Mandatos disponibles	21
----------------------	----

Limitar el acceso a los archivos de spool.	22
--	----

Personalizar procedimientos de construcción remota	
--	--

ELAXF*	22
--------	----

(Opcional) Definir una transacción APPC para el servicio de mandatos TSO	24
--	----

Preparación	25
-------------	----

Implementación	26
----------------	----

(Opcional) Personalizar miembros de procedimientos almacenados DB2 ELAXM*.	27
--	----

(Opcional) Personalizar el soporte para idiomas bidireccionales CICS (bidi)	28
---	----

(Opcional) Personalizar el gestor de despliegue de aplicaciones (ADM).	29
--	----

Repositorio CRD	31
-----------------	----

Región de conexión primaria CICS	31
----------------------------------	----

Manejador de mensajes de conducto	31
-----------------------------------	----

(Opcional) Regiones de conexión no primarias	
--	--

CICS	32
------	----

## Capítulo 4. Activar componentes z/OS UNIX de Developer para System z . . . . . 33

Guardar el archivo de configuración rsed.envvars en otro directorio	34
---	----

Personalizar rsed.envvars, el archivo de configuración de RSE	35
---	----

(Opcional) Definir el rango de puertos (PORTRANGE) disponibles para RSE.	40
--	----

(Opcional) Definir parámetros de arranque Java adicionales con _RSE_*OPTS.	41
--	----

Daemon INETD y configuración de REXEC/SSH de RSE	43
--	----

Configuración del daemon RSE de INETD	43
---------------------------------------	----

Configuración de REXEC (o SSH) de INETD	45
---	----

Personalizar el archivo de configuración ISPF, ISPF.conf	46
--	----

Verificar la instalación del servidor RSE	46
---	----

Disponibilidad de los puertos	47
-------------------------------	----

Conexión REXEC	48
----------------	----

Script de shell REXEC/SSH	49
---------------------------	----

Conexión del daemon RSE	50
-------------------------	----

Conexión del supervisor de trabajos JES.	50
--	----

Conexión del servicio de mandatos TSO (utilizando SCLMDT)	50
---	----

Conexión del servicio de mandatos TSO (utilizando APPC)	52
---	----

(Opcional) Personalizar la configuración SSL de RSE, ssl.properties	52
---	----

(Opcional) Personalizar la configuración del rastreo RSE, rsecomm.properties	53
--	----

(Opcional) Personalizar la configuración de proyectos de host, projectcfg.properties	54
--	----

(Opcional) Personalizar la integración del gestor de archivos, FMIEXT.properties . . . . .	54
--	----

## Capítulo 5. (Opcional) Activar el gestor de repositorios de acceso común (CARMA) de IBM . . . . . 57

Personalizar los componentes MVS de CARMA . . . . .	57
Personalizar los componentes z/OS UNIX de CARMA . . . . .	59
(Opcional) Activar los gestores de acceso a repositorios (RAM) de ejemplo . . . . .	60
Activar el RAM de SCLM . . . . .	61
Activar el RAM de PDS . . . . .	61
Activar el RAM de esqueleto . . . . .	61

## Capítulo 6. (Opcional) Activar IBM Software Configuration and Library Manager (SCLM) Developer Toolkit . . . 63

## Capítulo 7. Consideraciones en torno al cliente Developer para System z. . . . 65

## Capítulo 8. Consideraciones sobre el rendimiento . . . . . 67

Utilizar sistemas de archivos zFS . . . . .	67
Evitar el uso de STEPLIB . . . . .	67
Mejorar el acceso a las bibliotecas del sistema . . . . .	68
Bibliotecas de tiempo de ejecución de Language Environment (LE) . . . . .	68
Desarrollo de aplicaciones . . . . .	68
Mejorar el rendimiento de la comprobación de seguridad . . . . .	69
Compartir clases entre las JVM . . . . .	69
Habilitar la prestación de compartir clases . . . . .	69
Límites de tamaño de la caché . . . . .	70
Seguridad de la caché . . . . .	70
SYS1.PARMLIB(BPXPRMxx) . . . . .	70
Espacio en disco . . . . .	71
Utilidades para la gestión de cachés . . . . .	71
Memoria dinámica Java de tamaño fijo . . . . .	72
Gestión de cargas de trabajo (WLM) . . . . .	72

## Apéndice A. Ejecutar múltiples instancias de Developer para System z. 73

Archivos de configuración diferentes con idéntico nivel de software . . . . .	73
Todas las demás situaciones . . . . .	74

## Apéndice B. Resolución de problemas de configuración . . . . . 77

Ubicación de los archivos de anotaciones . . . . .	77
Anotaciones del supervisor de trabajos JES . . . . .	78
Anotaciones de la transacción APPC (servicio de mandatos TSO) . . . . .	78
Anotaciones de RSE . . . . .	78
Anotaciones de prueba de IVP de fekfivpc . . . . .	79
Anotaciones de integración del analizador de faltas . . . . .	79

Anotaciones de integración del gestor de archivos	79
Anotaciones de CARMA . . . . .	80
Archivos de vuelco . . . . .	80
Vuelcos de MVS . . . . .	80
Vuelcos Java . . . . .	81
Ubicación de los vuelcos en z/OS UNIX. . . . .	82
Autorización de control de programa para los programas RSE . . . . .	82
Puertos TCP/IP reservados . . . . .	83
Tamaño del espacio de direcciones. . . . .	84
Requisitos para INETD . . . . .	84
Limitaciones establecidas en SYS1.PARMLIB(BPXPRMxx). . . . .	85
Limitaciones almacenadas en el perfil de seguridad . . . . .	85
Limitaciones puestas en vigor por la rutinas de salida del sistema . . . . .	85
Rastreo de información de retorno de error. . . . .	85
Transacción APPC y el servicio de mandatos TSO	86
Información miscelánea . . . . .	88
Límites del sistema . . . . .	88
Problemas conocidos . . . . .	88
Emulador de Host Connect . . . . .	89
Ponerse en contacto con el soporte de IBM . . . . .	89

## Apéndice C. Configurar TCP/IP . . . . . 93

Dependencia del nombre de host . . . . .	93
Qué son los resolventes . . . . .	93
Qué es el orden de búsqueda de la información de configuración. . . . .	94
Orden de búsqueda utilizado en el entorno z/OS UNIX . . . . .	94
Archivos de configuración de resolvente base: . . . . .	94
Tablas de conversión: . . . . .	95
Tablas de hosts locales: . . . . .	96
Cómo se aplica esto a Developer para System z . . . . .	96

## Apéndice D. Configurar INETD . . . . . 101

inetd.conf . . . . .	101
ETC.SERVICES . . . . .	102
Orden de búsqueda utilizado en el entorno z/OS UNIX . . . . .	103
Orden de búsqueda utilizado en el entorno MVS nativo . . . . .	103
Definiciones de puertos en PROFILE.TCPIP . . . . .	104
/etc/inetd.pid . . . . .	105
Arranque. . . . .	105
/etc/rc . . . . .	105
/etc/inittab . . . . .	105
BPXBATCH . . . . .	106
Sesión de shell . . . . .	106
Seguridad . . . . .	107
Requisitos de Developer para System z. . . . .	108
INETD . . . . .	108
Daemon RSE . . . . .	108

## Apéndice E. Configurar SSL . . . . . 109

Clonar la configuración RSE existente . . . . .	110
Determinar qué archivo(s) de claves hay que utilizar . . . . .	110

Crear un almacén de claves con keytool . . . . .	111
Crear una base de datos de claves (solo daemon) . . . . .	112
Crear un anillo de claves con RACF . . . . .	112
Crear una base de datos de claves con gskkyman . . . . .	113
Activar SSL actualizando ssl.properties . . . . .	117
Probar la conexión. . . . .	118

## **Apéndice F. Configurar APPC . . . . . 119**

VSAM. . . . .	119
VTAM. . . . .	120

SYS1.PARMLIB(APPCPMxx) . . . . .	121
SYS1.PARMLIB(ASCHPMxx) . . . . .	122
Activar los cambios de APPC . . . . .	123
Definir la transacción del servicio de mandatos TSO . . . . .	123

## **Glosario . . . . . 125**

## **Avisos . . . . . 129**

Marcas registradas y marcas de servicio . . . . .	131
---	-----





---

## Figuras

1. FEJCNFG, archivo de configuración del supervisor de trabajos JES. . . . . 16
2. JCL del supervisor de trabajos JES . . . . . 19
3. Paneles ISPF de REXX para APPC . . . . . 25
4. rsed.envvars – archivo de configuración de RSE . . . . . 36
5. ISPF.conf - archivo de configuración de ISPF 46
6. ssl.properties – archivo de configuración SSL 53
7. rsecomm.properties – archivo de configuración de anotaciones . . . . . 53
8. projectcfg.properties – archivo de configuración de proyectos basados en host. . 54
9. FMIEXT.properties – archivo de configuración del gestor de archivos . . . . . 55
10. CRASRV.properties – archivo de configuración de CARMA . . . . . 59
11. JCL de arranque de INETD . . . . . 106
12. Importar certificado de host. . . . . 118
13. Preferencias . . . . . 118
14. JCL para crear los VSAM de APPC . . . . 120
15. SYS1.SAMPLIB(ATBAPPL) . . . . . 121
16. SYS1.PARMLIB(APPCPMxx) . . . . . 121
17. SYS1.PARMLIB(ASCHPMxx) . . . . . 123



---

## Tablas

1.	Publicaciones a las que se hace referencia	xi	8.	Lista de comprobación de calificadores de alto nivel de ELAXF*	23
2.	Matriz de instalación y configuraciones de Developer para System z	1	9.	Lista de comprobación para la transacción APPC	25
3.	Visión general de los miembros MVS personalizados	11	10.	Miembros de procedimientos almacenados DB2 ELAXM* de ejemplo	27
4.	Visión general de los archivos z/OS UNIX personalizados	13	11.	Archivos de configuración opcionales	34
5.	Personalización en bibliotecas no de Developer para System z.	14	12.	Lista de comprobación del cliente Developer para System z.	65
6.	Mandatos de la consola del supervisor de trabajos JES	21	13.	Definiciones locales disponibles para el resolvente	98
7.	Procedimientos ELAXF* de ejemplo	22			



---

## Acerca de esta publicación

En esta publicación se estudia la configuración de las funciones de IBM Rational Developer para System z. Incluye instrucciones que explican cómo configurar servidores IBM Rational Developer para System z en su sistema de hospedaje z/OS.

De aquí en adelante, en este manual se utilizarán los siguientes nombres:

- A *IBM Rational Developer para System z* se le llamará *Developer para System z*
- A *IBM Rational Developer para System z Common Access Repository Manager* se le llamará *Gestor de repositorios de acceso común*, y algunas veces se utilizará su abreviatura, *CARMA*
- A *IBM Software Configuration and Library Manager (SCLM) Developer Toolkit* se le llamará *SCLM Developer Toolkit*, y algunas veces se utilizará su abreviatura, *SCLMDT*

**Nota:** La información de configuración que se halla en este documento es para IBM Rational Developer para System z Versión 7.1.1.

En el caso de releases anteriores, incluidos los de IBM WebSphere Developer para System z, IBM WebSphere Developer para zSeries y WebSphere Studio Enterprise Developer, utilice la información de configuración que se encuentra en la guía de configuración de host y en los directorios de programas de dichos releases.

---

## A quién va dirigida esta publicación

Esta publicación está destinada a los programadores de sistemas que instalan y configuran IBM Rational Developer para System z Versión 7.1.1 en su sistema de hospedaje z/OS. Para utilizar esta publicación, debe estar familiarizado con los sistemas de hospedaje z/OS UNIX y MVS.

---

## Publicaciones a las que se hace referencia

Las publicaciones a las que se hace referencia en este documento son:

*Tabla 1. Publicaciones a las que se hace referencia*

Título de la publicación	Número de pedido	Referencia	Enlace de referencia
Java Diagnostic Guide	SC34-6650	Java 5.0	<a href="http://www.ibm.com/developerworks/java/jdk/diagnosis/index.html">http://www.ibm.com/developerworks/java/jdk/diagnosis/index.html</a>
Java SDK and Runtime Environment User Guide		Java 5.0	<a href="http://www-03.ibm.com/servers/eserver/zseries/software/java/">http://www-03.ibm.com/servers/eserver/zseries/software/java/</a>
Program Directory for IBM Rational Developer for System z	GI11-8298-00	RD/z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/index.html">http://www-306.ibm.com/software/awdtools/rdz/library/index.html</a>

Tabla 1. Publicaciones a las que se hace referencia (continuación)

Título de la publicación	Número de pedido	Referencia	Enlace de referencia
Program Directory for IBM Rational Developer for System z Common Access Repository Manager	GI11-8299-00	RD/z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/index.html">http://www-306.ibm.com/software/awdtools/rdz/library/index.html</a>
Program Directory for IBM Software Configuration and Library Manager (SCLM) Developer Toolkit	GI11-8306-00	RD/z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/index.html">http://www-306.ibm.com/software/awdtools/rdz/library/index.html</a>
Rational Developer for System z Application Deployment Manager User's Guide	SC23-7661	RD/z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/index.html">http://www-306.ibm.com/software/awdtools/rdz/library/index.html</a>
Rational Developer for System z Common Access Repository Manager Developer's Guide	SC23-7660	RD/z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/index.html">http://www-306.ibm.com/software/awdtools/rdz/library/index.html</a>
Rational Developer para System z Guía de configuración de host	SC11-3660 (SC23-7658)	RD/z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/index.html">http://www-306.ibm.com/software/awdtools/rdz/library/index.html</a>
Rational Developer para System z Guía de planificación de host	GI11-7839-00 (GI11-8296-00)	RD/z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/index.html">http://www-306.ibm.com/software/awdtools/rdz/library/index.html</a>
SCLM Developer Toolkit Installation and Customization Guide	SC23-8504	RD/z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/index.html">http://www-306.ibm.com/software/awdtools/rdz/library/index.html</a>
ABCs of z/OS System Programming Volume 9	SG24-6989	Redbook	<a href="http://www.redbooks.ibm.com/">http://www.redbooks.ibm.com/</a>
APPC and WebSphere Developer for System z	SC23-5885	Libro blanco	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/index.html">http://www-306.ibm.com/software/awdtools/rdz/library/index.html</a>
Host Based Projects in WebSphere Developer for System z version 7.0		Libro blanco	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/index.html">http://www-306.ibm.com/software/awdtools/rdz/library/index.html</a>
Setting up SSL for RSE Communication	SC23-5816	Libro blanco	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/index.html">http://www-306.ibm.com/software/awdtools/rdz/library/index.html</a>
Manual de servidor de comunicaciones	F1A1BK61	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Communications Server IP Configuration Guide	SC31-8775	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>

Tabla 1. Publicaciones a las que se hace referencia (continuación)

Título de la publicación	Número de pedido	Referencia	Enlace de referencia
Communications Server IP Configuration Reference	SC31-8776	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Communications Server IP Diagnosis Guide	GC31-8782	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Communications Server IP System Administrator's Commands	SC31-8781	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Communications Server SNA Network Implementation Guide	SC31-8777	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Cryptographic Services System SSL Programming	SC24-5901	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Language Environment Customization	SA22-7564	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Language Environment Debugging Guide	GA22-7560	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Manual de MVS	IEA2BK60	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
MVS Initialization and Tuning Guide	SA22-7591	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
MVS Initialization and Tuning Reference	SA22-7592	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
MVS JCL Reference	SA22-7597	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
MVS Planning APPC/MVS Management	SA22-7599	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
MVS Planning Workload Management	SA22-7602	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
MVS System Commands	SA22-7627	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Security Server RACF Command Language Reference	SA22-7687	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Security Server RACF Security Administrator's Guide	SA22-7683	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
UNIX System Services Command Reference	SA22-7802	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>

*Tabla 1. Publicaciones a las que se hace referencia (continuación)*

<b>Título de la publicación</b>	<b>Número de pedido</b>	<b>Referencia</b>	<b>Enlace de referencia</b>
UNIX System Services File System Interface Reference	SA22-7808	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
UNIX System Services Planning	GA22-7800	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
UNIX System Services User's Guide	SA22-7801	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>



## Capítulo 1. Instalar y configurar los componentes del host

Para cada una de las siguientes funciones, instale los FMID necesarios. Para obtener información de instalación sobre los diversos FMID, consulte el directorio de programas correspondiente al FMID que se propone instalar.

Tabla 2. Matriz de instalación y configuraciones de Developer para System z

Si necesita esta función de IBM Rational Developer para System z	Debe instalar estos FMID	Y aquí encontrará información de instalación y configuración
<ul style="list-style-type: none"><li>• Conectividad de host</li><li>• Conectividad de JES</li><li>• Compilación remota</li><li>• Información de retorno de errores para la compilación remota</li><li>• Depuración remota</li><li>• Procedimientos almacenados DB2</li><li>• Soporte de pantallas MFS de IMS</li><li>• Soporte de mapas BMS de CICS</li><li>• Soporte de idiomas bidireccionales CICS (bidi)</li><li>• Gestor de despliegue de aplicaciones (ADM)</li><li>• Integración del gestor de archivos</li><li>• Integración del analizador de faltas</li></ul>	HHOP710, HSD3310*	<ul style="list-style-type: none"><li>• <i>Program Directory for IBM Rational Developer para System z (GI11-8298-00)</i></li><li>• <i>Rational Developer para System z Guía de configuración de host, SC11-3660 (SC23-7658)</i></li><li>• <i>Rational Developer para System z Guía de planificación del host, GI11-7839-00 (GI11-8296-00)</i></li></ul> opcional <ul style="list-style-type: none"><li>• <i>Program Directory for IBM Software Configuration and Library Manager (SCLM) Developer Toolkit (GI11-8306-00)</i></li><li>• <i>SCLM Developer Toolkit Installation and Customization Guide (SC23-8504)</i></li></ul>
<ul style="list-style-type: none"><li>• Acceso de gestión de configuraciones de software común (CARMA)</li></ul>	HCMA710, HHOP710**	<ul style="list-style-type: none"><li>• <i>Program Directory for IBM Rational Developer para System z Common Access Repository Manager (GI11-8299-00)</i></li></ul> opcional <ul style="list-style-type: none"><li>• <i>Program Directory for IBM Rational Developer para System z (GI11-8299-00)</i></li><li>• <i>Rational Developer para System z Guía de configuración de host, SC11-3660 (SC23-7658)</i></li><li>• <i>Rational Developer para System z Guía de planificación del host, GI11-7839-00 (GI11-8296-00)</i></li></ul>
<ul style="list-style-type: none"><li>• Kit de herramientas de desarrollador del gestor de bibliotecas y configuraciones de software (SCLM)</li></ul>	HSD3310	<ul style="list-style-type: none"><li>• <i>Program Directory for IBM Software Configuration and Library Manager (SCLM) Developer Toolkit (GI11-8306-00)</i></li><li>• <i>SCLM Developer Toolkit Installation and Customization Guide (SC23-8504)</i></li></ul>

(\*) Para Developer para System z se necesita una conexión con el servicio de mandatos TSO en z/OS. Esta conexión se proporciona mediante uno de estos procedimientos:

1. Instalar y configurar el SCLM Developer Toolkit, FMID HSD3310 (predeterminado)
2. Instalar y configurar una transacción APPC

(\*\*) Para CARMA se necesita una interfaz basada en host. Esta función se proporciona mediante HHOP710, pero también se puede utilizar una interfaz basada en host de construcción personalizada y omitir la instalación de HHOP710.

---

## Consideraciones sobre la preinstalación

- Si es usted un usuario anterior de IBM Rational Developer para System z, IBM WebSphere Developer para System z, IBM WebSphere Developer para zSeries o WebSphere Studio Enterprise Developer, le recomendamos que guarde los archivos personalizados relacionados antes de instalar la ampliación de IBM Rational Developer para System z Versión 7.1.1. Para obtener una visión general de los archivos que se tienen que personalizar, consulte: “Hacer copia de seguridad de archivos configurados con anterioridad” en la página 11.
- Si piensa ejecutar múltiples instancias de Developer para System z, lea: Apéndice A, “Ejecutar múltiples instancias de Developer para System z”, en la página 73.
- Desde la versión 7.1, Developer para System z proporciona dos métodos de acceso para el servicio de mandatos TSO, que ejecuta los mandatos TSO/ISPF, ya sea implícitamente por medio del producto o explícitamente por parte del usuario.
  - Llamar a una función de SCLM Developer Toolkit. Para este método hay que instalar y configurar SCLMDT, FMID HSD3310, que viene junto con Developer para System z. Este es el método predeterminado que se utiliza en los miembros de ejemplo.

Hay que realizar los siguientes pasos de personalización de SCLMDT, pasos que se describen en el manual *SCLM Developer Toolkit Installation and Customization Guide (SC23-8504)*:

    - Crear la estructura del directorio y los archivos de configuración
    - Personalizar ISPF.conf

**Nota:** Para poder utilizar esta función de SCLMDT, no se necesita el producto SCLM ni el cliente basado en Eclipse.

- Una transacción APPC (como en los releases anteriores a la versión 7.1). Para este método hay que instalar y configurar APPC, que forma parte de z/OS.
- El módulo de carga bidi ha cambiado en la versión 7.0 (nombre y codificación), lo que exige eliminar los módulos más antiguos y configurar el nuevo.
- El código bidi creado con los releases anteriores (anteriores a la versión 7.0) se deben RECOMPILAR para poder usar el nuevo módulo FEJBDTRX.

Para obtener instrucciones detalladas de la instalación de SMP/E para cada uno de los FMID, consulte el directorio de programas pertinente, cuya lista encontrará en la Tabla 2 en la página 1.

---

## Consideraciones sobre la preconfiguración

Developer para System z tiene una lista del software prerequisite que hay que instalar y debe estar operativo para que el producto funcione. También hay una lista del software corequisite para poder utilizar las características específicas de Developer para System z. Estos requisitos deben estar instalados y ser operativos en tiempo de ejecución para que la correspondiente característica funcione como es debido.

Consulte el manual *Rational Developer para System z Guía de planificación del host*, GI11-7839-00 (GI11-8296-00) si desea obtener una lista de los prerequisites y corequisites para su versión de Developer para System z.

**Atención:** No se puede utilizar Java de 64 bites.

## Configuración necesaria de los productos y software obligatorios

Consulte al programador del sistema MVS, al administrador de seguridad y al administrador de TCP/IP para saber si los productos y software obligatorios están instalados, se han probado y funcionan.

- A partir de la versión 7.1, Developer para System z proporciona dos métodos para ejecutar mandatos TSO/ISPF, ya sea implícitamente por medio del producto o explícitamente por parte del usuario.
    - Llamar a una función de SCLM Developer Toolkit. Para este método hay que instalar y configurar SCLMDT, FMID HSD3310, que viene junto con Developer para System z. Este es el método predeterminado que se utiliza en los miembros de ejemplo.
- Hay que realizar los siguientes pasos de personalización de SCLMDT, pasos que se describen en el manual *SCLM Developer Toolkit Installation and Customization Guide* (SC23-8504):
- Crear la estructura del directorio y los archivos de configuración
  - Personalizar ISPF.conf

**Nota:** Para poder utilizar esta función de SCLMDT, no se necesita el producto SCLM ni el cliente basado en Eclipse.

- Una transacción APPC (como en los releases anteriores a la versión 7.1). Para este método hay que instalar y configurar APPC, que forma parte de z/OS.
- La biblioteca de clases DLL de C/C++ CBC.SCLBDLL y las bibliotecas de tiempo de ejecución de Language Environment (LE), CEE.SCEERUN y CEE.SCEERUN2, deben estar en LINKLIST.
- Todos los usuarios de z/OS UNIX deben tener acceso de lectura y ejecución (READ y EXECUTE) a los directorios de Java.
- Developer para System z depende de INETD a la hora de poner a punto la conexión entre cliente y host. Hallará más información sobre REXEC y SSH en: Apéndice D, “Configurar INETD”, en la página 101.
- Para las acciones remotas (basadas en host) de los subproyectos z/OS UNIX, es necesario que REXEC o SSH esté activo en el host.
- Developer para System z depende de que TCP/IP tenga el nombre de host correcto en el momento de la inicialización. Ello implica que los distintos archivos de configuración de TCP/IP y del resolvente estén configurados correctamente. Para obtener información de personalización de TCP/IP y del

resolvente, vea: Apéndice C, "Configurar TCP/IP", en la página 93, así como las *sentencias de configuración de TCPIP.DATA*, en el manual *Communications Server IP Configuration Reference (SC31-8776)*.

Puede probar la configuración de TCP/IP con el mandato TSO **HOMETEST**. Consulte el apartado de *mandatos TSO*, en el manual *Communications Server IP System Administrator's Commands (SC31-8781)* para obtener más información sobre este mandato.

Salida de ejemplo del mandato **HOMETEST**:

Ejecutar IBM MVS TCP/IP CS V1R7 TCP/IP Configuration Tester

El archivo de parámetros de configuración de FTP utilizado será "SYS1.TCPPARMS(FTPDATA)"

El nombre de host TCP es: CDFMVS08

Utilizar el servidor de nombres para resolver CDFMVS08

Las siguientes direcciones IP corresponden al nombre de host TCP: CDFMVS08  
9.42.112.75

Las siguientes direcciones IP son las direcciones IP HOME definidas en PROFILE.TCPIP:  
9.42.112.75  
127.0.0.1

¡Todas las direcciones IP de CDFMVS08 están en la lista HOME!

La prueba de Home ha sido satisfactoria - ¡Se han superado todas las pruebas!

## Consideraciones sobre el ID de usuario

El ID de un usuario de Developer para System z debe tener (como mínimo) los siguientes atributos:

- Acceso TSO (con un tamaño de región normal).
- Un segmento OMVS definido en el sistema de seguridad (por ejemplo, RACF), para el ID de usuario y para el grupo predeterminado. Este segmento OMVS puede ser individual o un segmento predeterminado a escala del sistema. Para obtener más información sobre el segmento OMVS predeterminado, vea el manual *Security Server RACF Security Administrator's Guide (SA22-7683)*.
  - Un directorio inicial (con acceso de escritura, lectura y ejecución, WRITE, READ y EXECUTE) asignado para el usuario e identificado en el segmento OMVS.
  - El campo PROGRAM del segmento OMVS debe ser /bin/sh u otra shell válida de z/OS UNIX, como la shell /bin/tcsh.
  - El campo ASSIZEMAX no debe estar establecido, para que se utilicen los valores predeterminados del sistema.
  - Para el ID de usuario no se necesita el UID 0.

Ejemplo (mandato **LISTUSER userid NORACF OMVS**):

USER=userid

INFORMACIÓN de OMVS

```
-----  
UID= 0000003200  
HOME= /u/userid  
PROGRAM= /bin/sh  
CPUTIMEMAX= NONE  
ASSIZEMAX= NONE  
FILEPROCMA= NONE  
PROCUSERMA= NONE  
THREADSMA= NONE  
MMAPAREAMA= NONE
```

- Para el grupo predeterminado de ID de usuario, se necesita un GID.

Ejemplo (mandato **LISTGRP grupo NORACF OMVS**):

GROUP grupo

INFORMACIÓN de OMVS

-----  
GID= 0000003243

- Acceso de lectura y ejecución (READ y EXECUTE) a los directorios de instalación y configuración de Developer para System z, cuyo valor predeterminado es /usr/lpp/wd4z/\*.

## Consideraciones sobre el servidor

Los siguientes servicios de host y, por consiguiente, sus puertos deben estar disponibles para que el cliente se conecte a ellos. Los demás puertos que utiliza Developer para System z tienen tráfico solo de host. Es decir, para Developer para System z, los puertos enumerados son los únicos que se deben definir ante el cortafuegos que protege el host.

- Servicio INETD para iniciar el servidor RSE. En Developer para System z hay muchas maneras de hacerlo:
  - Daemon RSE, puerto predeterminado 4035
  - REXEC, puerto predeterminado 512
  - SSH, puerto predeterminado 22
- Servidor RSE para la comunicación entre cliente y host. Utiliza por defecto cualquier puerto disponible, pero puede estar limitado a un rango especificado. La comunicación en este puerto puede estar cifrada mediante SSL.
- Servicio Telnet TN3270 para el emulador Host Connect, puerto predeterminado 23. La comunicación en este puerto puede estar cifrada mediante SSL (puerto predeterminado 992). El puerto predeterminado que se asigna al servicio Telnet TN3270 depende de si el usuario elige el uso del cifrado.

**Nota:** Los clientes anteriores (versión 7.0 y anteriores) se comunican directamente con el supervisor de trabajos JES, puerto predeterminado 6715.

**Nota:** Durante una sesión de depuración remota para Cobol, PL/I o Assembler, se invoca IBM Debug Tool para z/OS. Este producto se comunica directamente con el cliente. La comunicación se inicia en el host, y se conecta al puerto 8001 del cliente.

INETD es un proceso servidor z/OS UNIX que se necesita para configurar las conexiones entre cliente y host de Developer para System z. Los valores del entorno INETD, que se pasan al iniciar un proceso, y los permisos del ID de usuario de INETD deben estar debidamente establecidos para que INETD inicie el servidor RSE.

- Si INETD se inicia mediante JCL utilizando BPXBATCH, el tamaño de la región debe ser 0.
- Si INETD se inicia desde una sesión TSO/OMVS, el tamaño de la región TSO debe ser igual o mayor que 2096128.
- Acceso de lectura y ejecución (READ y EXECUTE) a los directorios de instalación de Developer para System z, cuyo valor predeterminado es /usr/lpp/wd4z/\*.
- Permiso para BPX.DAEMON y posiblemente para el UID 0.

Para obtener más información sobre los permisos de INETD, vea: Apéndice D, “Configurar INETD”, en la página 101.

El Explorador de Sistemas Remotos (RSE) es el componente de Developer para System z que proporciona servicios centrales como los de conectar el cliente al host.

- Cada conexión entre cliente y host tiene un servidor RSE privado.
- Además de los puertos documentados en este manual para comunicar con otros servidores Developer para System z, se necesitan numerosos puertos para uso interno del servidor RSE.

## Permisos necesarios para implementar las tareas de configuración

Para la configuración de Developer para System z se necesita más que los permisos típicos del programador de sistemas, porque cabe esperar una mínima ayuda por parte de otras personas. En la siguiente lista se resaltan las áreas más importantes:

- Permiso de actualización sobre los conjuntos de datos del sistema (como SYS1.PARMLIB)
- Permiso para emitir mandatos de consola
- Permiso para crear directorios de z/OS UNIX
- Permiso para actualizar el directorio /etc de z/OS UNIX
- Permiso para establecer el bit de control de programa de z/OS UNIX
- Permiso para definir nuevos puertos TCP/IP
- Permiso para BPX.DAEMON y posiblemente para el UID 0 para (re)iniciar INETD
- Permisos de gestión de WLM/RSM
- Permisos de gestión de APPC para crear un programa de transacción
- Permisos de administrador de seguridad para consultar y crear perfiles de seguridad
- (opcional) Permisos de administrador de DB2 para crear un procedimiento almacenado
- (opcional) Permisos de administrador de CICS para definir programas ante CICS

---

## Consideraciones sobre el predespliegue

Developer para System z y el gestor de repositorios de acceso común (CARMA) permiten clonar una instalación en un sistema distinto, lo cual evita la necesidad de instalar SMP/E en cada sistema.

A continuación se indican los conjuntos de datos, directorios y archivos que son obligatorios para el despliegue en otros sistemas. Si ha copiado un archivo en una ubicación diferente para la personalización, ese archivo debe sustituir a su equivalente en las listas que figuran más abajo.

**Nota:** Las listas que siguen no cubren las necesidades de despliegue del software prerequisite y correquisito.

## IBM Rational Developer para System z, FMID HHOP710

- hlq.SFEKLOAD(\*)
- hlq.SFEKPROC(\*)
- hlq.SFEKSAMP(FEJJCNFG)

- hlq.SFEKSAMP(FEJJJCL)
- hlq.SFEKSAMP(ELAXF\*)
- /usr/lpp/wd4z/\*
- Componentes opcionales
  - hlq.SFEKSAMP(FEJTSO) - sometimiento TSO
  - hlq.SFEKSAMP(ELAXM\*) - procedimiento almacenado DB2
  - Definiciones CICS creadas por hlq.SFEKSAMP(ADNPCCSD) - conexión primaria ADM
  - Definiciones CICS creadas por hlq.SFEKSAMP(ADNARCSO) - conexión no primaria ADM
  - VSAM creado por hlq.SFEKSAMP(ADNVSAM) - servidor ADM CRD
  - Módulo de carga creado por hlq.SFEKSAMP(ADNCMSGH) - manejador de conductos ADM
  - Transacción APPC creada por hlq.SFEKSAMP(FEKAPPCC) - servicio de mandatos TSO basados en APPC
  - /etc/SCLMDT/CONFIG/ISPF.conf - añadir el servidor de mandatos TSO al entorno ISPF de SCLM Developer Toolkit

**Nota:** hlq y /usr/lpp/wd4z son el calificador de alto nivel (valor predeterminado FEK) y la vía de acceso que se emplean durante la instalación del producto.

## **IBM Gestor de repositorios de acceso común (CARMA), FMID HCMA710**

- hlq.SCRALOAD(\*)
- hlq.SCRACLIST(CRASUBMT)
- VSAM creados por los siguientes miembros de hlq.SCRASAMP
  - VSAM de mensajes hlq.SCRASAMP(CRA\$VMSG)
  - VSAM de configuraciones hlq.SCRASAMP(CRA\$VDEF)
  - VSAM de información personalizada hlq.SCRASAMP(CRA\$SVSTR)

**Nota:** hlq es el calificador de alto nivel (el valor predeterminado es CRA) que se emplea durante la instalación del producto.





---

## Capítulo 2. Cambios de instalación y configuración

En este apartado se resaltan los cambios de instalación y configuración comparados con los releases anteriores del producto. Para obtener más información, consulte las secciones relacionadas de este manual.

---

### Cambios entre la versión 7.0 y la versión 7.1

#### IBM Rational Developer para System z, FMID HHOP710

- **Adición:** opción de instalación - mandatos TSO/ISPF por medio de una transacción APPC o por medio de SCLM Developer Toolkit
- **Cambio:** la transacción APPC aprovecha una característica ISPF nueva
- **Adición:** los siguientes miembros personalizables son nuevos
  - samplib ELAXFADT
  - samplib ADNMSGH
  - /usr/lpp/wd4z/rse/lib/FMIEXT.properties
- **Cambio:** se han trasladado los miembros siguientes
  - SFEKDLL(FEJBDTRX) -> SFEKLOAD(FEJBDTRX)
- **Cambio:** los miembros personalizables que han cambiado son los siguientes
  - samplib FEKFAPPCC
  - /usr/lpp/wd4z/rse/lib/rsed.envvars
  - /usr/lpp/wd4z/rse/lib/setup.env.zseries
  - /usr/lpp/wd4z/rse/lib/server.zseries

#### IBM Gestor de repositorios de acceso común (CARMA), FMID HCMA710

- **Cambio:** las anotaciones se escriben en la sentencia DD CARMALOG
- **Cambio:** el VSAM de mensaje (CRAMSG) y el VSAM de configuración (CRADEF) de CARMA se han actualizado
- **Adición:** los siguientes miembros personalizables son nuevos
  - samplib CRA#ECOB
  - samplib CRA#EPDS
  - samplib CRA#ERAM
  - samplib CRA#ESLM
- **Cambio de nombre:** los miembros personalizables que han cambiado de nombre son los siguientes
  - samplib CRAREPR -> CRA\$VDEF
  - samplib CRAMREPR -> CRA\$VMSG
  - samplib CRASREPR -> CRA\$VSTR
  - samplib CRASALX -> CRA#ASLM
  - samplib CRACOBJ1 -> CRA#CCB1
  - samplib CRACOBJ2 -> CRA#CCB2
  - samplib CRACLICM -> CRA#CCLT
  - samplib CRARAMCS -> CRA#CPDS

- samplib CRARAMCM -> CRA#CRAM
- samplib CRATREPR -> CRA#VPDS
- samplib CRALREPR -> CRA#VSLM
- samplib CRACLIRN -> CRA#XCLT
- **Cambio:** los miembros personalizables que han cambiado son los siguientes
  - clist CRASUBMT

---

## Cambios entre la versión 6.0.1 y la versión 7.0

### IBM WebSphere Developer para System z, FMID HHOP700

- **Cambio:** ahora IBM WebSphere Developer para System z, FMID HHOP700, combina las funciones de 5 productos distintos en 1 solo para reducir el trabajo de instalación
  - v601 FMID H001600, IBM WebSphere Developer para zSeries RSE + ICU
  - v601 FMID H002600, IBM WebSphere Developer para zSeries Supervisor de trabajos JES
  - v601 FMID HEDS500, IBM WebSphere Developer para zSeries Opciones para z/OS (no todas las funciones se incluyen en HHOP700)
  - v601 FMID HBDI601, IBM WebSphere Developer para zSeries Soporte bidi
  - (Nuevo) versión 700, IBM Gestor de despliegue de aplicaciones (ADM)
- **Cambio:** los trabajos de instalación han cambiado de nombre para que los pasos que hay que realizar estén ordenados alfabéticamente
- **Cambio:** ahora la instalación de SMP/E establece nuevos bites de control de programa en z/OS UNIX
- **Cambio:** la codificación de servidor REXX se ha trasladado de samplib a un nuevo conjunto de datos, SFEKPROC
- **Cambio:** Bidi utiliza un módulo de carga nuevo, y hay que eliminar todas las referencias al antiguo
- **Cambio:** el parámetro JESNAME ya no se utiliza en el archivo de configuración del supervisor de trabajos JES (samplib FEJJCNFG)
- **Cambio:** los siguientes miembros ya no necesitan de personalización
  - /usr/lpp/wd4z/rse/lib/setup.env.zseries
- **Adición:** los nuevos miembros son los siguientes
  - samplib ELAXFCPC
  - samplib ELAXFCPP
  - samplib ELAXFMFS
  - /usr/lpp/wd4z/rse/lib/projectcfg.properties
- **Cambio de nombre:** los miembros personalizables que han cambiado de nombre son los siguientes
  - samplib FEKFRDIS -> FEKAPPCL
  - samplib FEKFRDEL -> FEKAPPCX
  - samplib FEKFRDAD -> FEKAPPCC
- **Eliminación:** los miembros que ya no se entregan son los siguientes
  - samplib ELAXFLNK
  - samplib FEJIVJCL
  - loadlib FEJBDTRN
  - loadlib FEJBDTRE

## IBM Gestor de repositorios de acceso común (CARMA), FMID HCMA700

- **Cambio:** los trabajos de instalación han cambiado de nombre para que los pasos que hay que realizar estén ordenados alfabéticamente
- **Cambio:** los siguientes miembros ya no necesitan de personalización
  - /usr/lpp/wd4z/rse/lib/rsed.envvars
  - /usr/lpp/wd4z/rse/lib/rexxsub

### Hacer copia de seguridad de archivos configurados con anterioridad

Antes de instalar Developer para System z versión 7.1.1, si es usted un usuario anterior de Developer para System z, le recomendamos que guarde los archivos personalizados relacionados. Si piensa ejecutar múltiples instancias de Developer para System z, antes de empezar la personalización, lea: Apéndice A, “Ejecutar múltiples instancias de Developer para System z”, en la página 73.

Tabla 3 y Tabla 4 en la página 13 proporcionan una visión general de los archivos que se pueden haber personalizado para Developer para System z y CARMA versión 6.0.1 o superior. Tabla 5 en la página 14 proporciona una lista de las personalizaciones de conjuntos de datos, productos y software prerequisite y corequisite que se producen durante una instalación de Developer para System z y CARMA versión 6.0.1 (o superior).

*Tabla 3. Visión general de los miembros MVS personalizados*

Miembro	Ubicación predeterminada v6.0.1	Ubicación predeterminada v7.0/v7.1	Finalidad
FEJCNFG	hlq.SFEJSAMP (hlq = FEJ)	hlq.SFEKSAMP (hlq = FEK)	Archivo de configuración del supervisor de trabajos JES
FEJJJCL	hlq.SFEJSAMP (hlq = FEJ)	hlq.SFEKSAMP (hlq = FEK)	JCL del supervisor de trabajos JES
FEJTSO	hlq.SFEJSAMP (hlq = FEJ)	hlq.SFEKSAMP (hlq = FEK)	JCL de sometimientos TSO
FEKAPPCC	no existe	hlq.SFEKSAMP (hlq = FEK)	JCL para crear una transacción APPC
FEKAPPCL	no existe	hlq.SFEKSAMP (hlq = FEK)	JCL para visualizar una transacción APPC
FEKAPPCX	no existe	hlq.SFEKSAMP (hlq = FEK)	JCL para suprimir una transacción APPC
FEKFRTAD	hlq.SFEKSAMP (hlq = FEK)	(nombre de miembro nuevo FEKAPPCC)	JCL para crear una transacción APPC

Tabla 3. Visión general de los miembros MVS personalizados (continuación)

<b>Miembro</b>	<b>Ubicación predeterminada v6.0.1</b>	<b>Ubicación predeterminada v7.0/v7.1</b>	<b>Finalidad</b>
FEKFRDIS	hlq.SFEKSAMP (hlq = FEK)	(nombre de miembro nuevo FEKAPPCL)	JCL para visualizar una transacción APPC
FEKFRDEL	hlq.SFEKSAMP (hlq =FEK)	(nombre de miembro nuevo FEKAPPLX)	JCL para suprimir una transacción APPC
ELAXF*	hlq.SCCUSAMP (hlq = CCU)	hlq.SFEKSAMP (hlq = FEK)	JCL para construcciones de proyectos remotos, etc.
ELAXMSAM	hlq.SCCUSAMP (hlq = CCU)	hlq.SFEKSAMP (hlq = FEK)	Procedimiento JCL del espacio de direcciones WLM para el constructor de procedimientos almacenados PL/I y COBOL
ELAXMJCL	hlq.SCCUSAMP (hlq = CCU)	hlq.SFEKSAMP (hlq = FEK)	JCL para definir el constructor de procedimientos almacenados PL/I y COBOL ante DB2
ADNVSAM	no existe	hlq.SFEKSAMP (hlq = FEK)	JCL para definir el repositorio CRD de ADM
ADNPCCSD	no existe	hlq.SFEKSAMP (hlq = FEK)	JCL para activar el servidor CRD en la región CICS primaria
ADNCMSGH	no existe	hlq.SFEKSAMP (hlq = FEK) (no existe en la versión 7.0)	JCL para compilar el manejador de mensajes de conducto
ADNSMSGH	no existe	hlq.SFEKSAMP (hlq = FEK)	Código fuente de ejemplo para el manejador de mensajes de conducto

Tabla 3. Visión general de los miembros MVS personalizados (continuación)

Miembro	Ubicación predeterminada v6.0.1	Ubicación predeterminada v7.0/v7.1	Finalidad
ADNARCSD	no existe	hlq.SFEKSAMP (hlq = FEK)	JCL para activar el servidor CRD en regiones CICS no primarias
CRASUBMT	hlq.SCRACLST (hlq = CRA)	hlq.SCRACLST (help = CRA)	CLIST para someter JCL de un servidor CARMA
CRAMREPR	hlq.SCRASAM (hlq = CRA)	hlq.SCRASAMP (hlq = CRA) (nombre de miembro nuevo en la versión 7.1 CRA\$VMSG)	JCL para crear el VSAM de mensajes de CARMA
CRAREPR	hlq.SCRASAM (hlq = CRA)	hlq.SCRASAMP (hlq = CRA) (nombre de miembro nuevo en la versión 7.1 CRA\$VDEF)	JCL para crear el VSAM de configuraciones de CARMA
CRASREPR	hlq.SCRASAM (hlq = CRA)	hlq.SCRASAMP (hlq = CRA) (nombre de miembro nuevo en la versión 7.1 CRA\$VSTR)	JCL para crear el VSAM de información personalizada de CARMA
CRALREPR	hlq.SCRASAM (hlq = CRA)	hlq.SCRASAMP (hlq = CRA) (nombre de miembro nuevo en la versión 7.1 CRA#VSLM)	JCL para crear el VSAM de mensajes de RAM SCLM
CRASALX	hlq.SCRASAM (hlq = CRA)	hlq.SCRASAMP (hlq = CRA) (nombre de miembro nuevo en la versión 7.1 CRA#ASLM)	JCL para crear los conjuntos de datos de RAM SCLM
CRATREPR	hlq.SCRASAM (hlq = CRA)	hlq.SCRASAMP (hlq = CRA) (nombre de miembro nuevo en la versión 7.1 CRA#VPDS)	JCL para crear el VSAM de mensajes de RAM PDS

**Nota:** hlq es el calificador de alto nivel que se emplea durante la instalación del producto. Se muestran los valores predeterminados suministrados por IBM para hlq, pero quizá no sean válidos para su local.

Tabla 4. Visión general de los archivos z/OS UNIX personalizados

Archivo	Ubicación predeterminada v6.0.1	Ubicación predeterminada versión 7.0/versión 7.1	Finalidad
rased.envvars	/usr/lpp/wd4z/rse/lib/	/usr/lpp/wd4z/rse/lib/	Archivo de configuración de RSE
rsecomm.properties	/usr/lpp/wd4z/rse/lib/	/usr/lpp/wd4z/rse/lib/	Archivo de configuración de rastreo de RSE

Tabla 4. Visión general de los archivos z/OS UNIX personalizados (continuación)

Archivo	Ubicación predeterminada v6.0.1	Ubicación predeterminada versión 7.0/versión 7.1	Finalidad
ssl.properties	/usr/lpp/wd4z/rse/ lib/	/usr/lpp/wd4z/rse/ lib/	Archivo de configuración SSL de RSE
setup.env.zseries	/usr/lpp/wd4z/rse/ lib/	(ya no está personalizado)	Exportar variables de entorno RSE
projectcfg.properties	(no existe)	/usr/lpp/wd4z/rse/ lib/	Archivo de configuración de proyectos de host
FMIEXT.properties	(no existe)	/usr/lpp/wd4z/rse/ lib/ (no existe en la versión 7.0)	Archivo de configuración del gestor de archivos
CRASRV.properties	/usr/lpp/wd4z/rse/ lib/	/usr/lpp/wd4z/rse/ lib/	Archivo de configuración de CARMA
rexsub	/usr/lpp/wd4z/rse/ lib/	(ya no está personalizado)	REXX de CARMA z/OS UNIX para ejecutar MVS CRASUBMT CLIST

**Nota:** /usr/lpp/wd4z es la vía de acceso que se emplea durante la instalación de los productos. Se muestra el valor predeterminado suministrado por IBM, pero quizá no sea válido para su local.

Tabla 5. Personalización en bibliotecas no de Developer para System z

Miembro/ Archivo	Ubicación predeterminada	Personalización necesaria
BPXPRMxx	SYS1.PARMLIB	Establecer que MAXASSIZE sea igual o mayor que 2147483647
PROGxx	SYS1.PARMLIB	hlq.SFEKLOAD autorizada por APF
ASCHPMxx	SYS1.PARMLIB	Definir una clase de transacción APPC para el servicio de mandatos TSO
services	/etc/	Definir daemon RSE
inetd.conf	/etc/	Definir daemon RSE
ISPF.conf	/etc/SCLMDT/ CONFIG/	Definir ubicación del servidor de mandatos TSO
/	APPC	Definir una transacción APPC para el servicio de mandatos TSO
/	WLM	Asociar el programa de transacción APPC a un grupo de rendimiento TSO
/	WLM	Asignar un entorno de aplicaciones para el procedimiento almacenado de DB2

---

## Capítulo 3. Activar componentes MVS de Developer para System z

Antes de instalar la versión 7.1.1, si es usted un usuario anterior de Developer para System z, le recomendamos que guarde la personalización relacionada, como se describe en: “Hacer copia de seguridad de archivos configurados con anterioridad” en la página 11.

En este capítulo, todas las referencias a hlq se refieren al calificador de alto nivel empleado durante la instalación de Developer para System z. El valor predeterminado de la instalación es FEK, pero quizá no sea válido para su local.

**Nota:** La biblioteca de clases DLL de C/C++ CBC.SCLBDLL y las bibliotecas de tiempo de ejecución de Language Environment (LE), CEE.SCEERUN y CEE.SCEERUN2, deben estar en LINKLIST.

El requisito LINKLIST se puede dejar de lado si se añade una sentencia STEPLIB a rsed.envvars; vea: “Personalizar rsed.envvars, el archivo de configuración de RSE” en la página 35.

---

### Establecer MAXASSIZE en SYS1.PARMLIB(BPXPRMxx)

MAXASSIZE define el tamaño de la región del espacio de direcciones (proceso). Establezca que MAXASSIZE, en SYS1.PARMLIB(BPXPRMxx), sea igual o mayor que 2147483647.

Este valor se puede comprobar y establecer dinámicamente (hasta la próxima IPL) con los siguientes mandatos de consola, como se describe en el manual *MVS System Commands* (SA22-7627).

1. DISPLAY OMVS,0
2. SETOMVS MAXASSIZE=2147483647

Para obtener más información sobre otras ubicaciones en las que se puede establecer o limitar el tamaño de los espacios de direcciones, consulte: “Tamaño del espacio de direcciones” en la página 84.

---

### Biblioteca hlq.SFEKLOAD autorizada por APF

Para que el supervisor de trabajos JES pueda acceder a los archivos de spool JES, el módulo FEJMON de la biblioteca de carga hlq.SFEKLOAD debe estar autorizada por APF.

Las autorizaciones de APF se definen en SYS1.PARMLIB(PROGxx), si su local siguió las recomendaciones de IBM. Consulte el manual *MVS Initialization and Tuning Reference* (SA22-7592) si desea más información sobre cómo definir autorizaciones de APF.

Cuando la finalidad es someter a prueba, las autorizaciones de APF también se pueden otorgar dinámicamente. Durarán hasta la próxima IPL del sistema. El mandato de consola que se necesita sería como este:

```
SETPROG APF,ADD,DSN=hlq.SFEKLOAD,SMS
```

Consulte el manual *MVS System Commands* (SA22-7627) si desea obtener más información sobre el mandato **SETPROG**.

---

## Personalizar el archivo de configuración del supervisor de trabajos JES, FEJJCNFG

El supervisor de trabajos JES es el componente de Developer para System z que maneja toda la conectividad de JES. Se utiliza un archivo de configuración para personalizar determinados aspectos de acuerdo con los estándares de su local.

**Nota:** Le recomendamos que copie el archivo de configuración de ejemplo en un nuevo conjunto de datos y que personalice esa copia, para evitar que se sobrescriba al aplicar el mantenimiento.

El archivo de configuración de ejemplo se encuentra en:

hlq.SFEKSAMP(FEJJCNFG)

El archivo consta de un conjunto de definiciones de variables de entorno. Las líneas de comentarios empiezan por el signo de almohadilla (#). El archivo de configuración de ejemplo figura en: Figura 1.

```
SERV_PORT=6715
CODEPAGE=UTF-8
HOST_CODEPAGE=IBM-1047
TZ=EEST5EDT
#LIMIT_VIEW=USERID
LISTEN_QUEUE_LENGTH=5
MAX_DATASETS=32
MAX_THREADS=200
TIMEOUT_INTERVAL=1200
TIMEOUT=3600
AUTHMETHOD=RACF
#_BPXK_SETIBMOPT_TRANSPORT=<nombre pila tcpip>
```

*Figura 1. FEJJCNFG, archivo de configuración del supervisor de trabajos JES*

Las definiciones que se necesitan son las siguientes:

### **SERV\_PORT**

Número de puerto del servidor de hospedaje del supervisor de trabajos JES. El puerto predeterminado es 6715. Cámbielo según desee, pero tenga en cuenta que AMBOS, el cliente y el servidor, deben estar configurados con el mismo número de puerto. Si cambia el número de puerto del servidor, todos los usuarios de clientes Developer para System z también deben cambiar el puerto del supervisor de trabajos JES para este sistema en la vista Sistemas Remotos.

**Nota:** Antes de seleccionar un puerto, verifique que el puerto está disponible en su sistema con los mandatos TSO **NETSTAT** y **NETSTAT PORTL**. Hallará más información en: “Puertos TCP/IP reservados” en la página 83.

**Nota:** Cuando se utiliza un cliente de la versión 7.1 o superior, toda comunicación en este puerto está confinada a la máquina de hospedaje.

### **CODEPAGE**

Página de códigos de la estación de trabajo. El valor predeterminado es



UTF-8. La página de códigos de la estación de trabajo tiene el valor UTF-8 y, en general, no se debe cambiar. Es posible que en algún momento tenga que cambiar UTF-8 para que coincida con la página de códigos de la estación de trabajo si tiene dificultades con caracteres del NLS, como el símbolo de moneda.

#### **HOST\_CODEPAGE**

Página de códigos del host. El valor predeterminado es IBM-1047. Cámbielo según necesite.

#### **TZ**

Selector de huso horario. El valor predeterminado es EST5EDT. El huso horario predeterminado es UTC +5 horas (horario de verano según la hora estándar del este (EST)). Cambie este valor para que represente su huso horario. Hallará información adicional en el manual *UNIX System Services File System Interface Reference (SA22-7808)*.

#### **LISTEN\_QUEUE\_LENGTH**

Longitud de la cola de escucha TCP/IP. El valor predeterminado es 5. No lo cambie, a menos que así se lo indique el centro de soporte de IBM.

#### **MAX\_DATASETS**

El valor predeterminado es 32. Es el número máximo de conjuntos de datos de salida en spool que el supervisor de trabajos JES devolverá al cliente (por ejemplo, SYSOUT, SYSPRINT, SYS00001, etcétera).

#### **MAX\_THREADS**

El valor predeterminado es 200. Número máximo de usuarios que pueden utilizar un supervisor de trabajos JES en un momento dado. Si aumenta este número, es posible que también deba aumentar el tamaño del espacio de direcciones del supervisor de trabajos JES.

#### **TIMEOUT\_INTERVAL**

El valor predeterminado es 1200. Controla la frecuencia con la que el servidor hace comprobaciones para buscar y desactivar hebras cuyo tiempo se haya agotado (vea **TIMEOUT**). El valor de **TIMEOUT\_INTERVAL** especifica el número de segundos entre cada dos comprobaciones.

#### **TIMEOUT**

El valor predeterminado es 3600. Tiempo, en segundos, que debe transcurrir antes de desactivar una hebra debido a que carece de interacción con el cliente. El valor máximo es 2147483647.

#### **AUTHMETHOD**

El valor predeterminado es RACF, y significa que se utiliza la interfaz de seguridad estándar. Este valor no se debe cambiar, ni siquiera en el caso de que utilice un producto distinto de RACF.

Las definiciones que figuran a continuación son opcionales. Si se omiten, se emplearán los valores predeterminados que se especifican más abajo:

#### **LIMIT\_VIEW=USERID**

Define qué datos de salida puede ver el usuario. El valor predeterminado (**LIMIT\_VIEW=NOLIMIT**) permite que el usuario vea todos los datos de salida de JES. Si se especifica **USERID**, la vista queda limitada a los datos de salida que sean propiedad del usuario.

**Nota:** Los únicos valores válidos son **USERID** y **NOLIMIT**.

#### **\_BPXK\_SETIBMOPT\_TRANSPORT=<nombre pila tcpip>**

Especifica la pila TCPIP que hay que utilizar en el caso de que haya múltiples pilas activas. El valor predeterminado es TCPIP. <nombre pila

tcpip> es el nombre de la pila TCPIP solicitada, tal como se define en la sentencia TCPIPJOBNAME del TCPIP.DATA relacionado.

**Nota:** El hecho de codificar una sentencia SYSTCPD DD en el JCL no establece la afinidad de pila solicitada.

#### **CONCHAR**

Especifica el carácter del mandato de consola JES. CONCHAR toma por defecto el valor CONCHAR=\$ para JES2, o el valor CONCHAR=\* para JES3. Si en su instalación se ha definido un carácter de mandato distinto, especifíquelo como valor de CONCHAR.

#### **SUBMITMETHOD=TSO**

Someter mediante TSO. El valor predeterminado (SUBMITMETHOD=JES) hace que los trabajos se sometan directamente a JES. Si se especifica SUBMITMETHOD=TSO, el trabajo se someterá por medio del mandato **SUBMIT** de TSO. Este método permite invocar rutinas de salida de TSO; sin embargo, como afecta negativamente al rendimiento, no conviene utilizarlo.

**Nota:** Los únicos valores válidos son TSO y JES.

**Nota:** Si se especifica SUBMITMETHOD=TSO, también hay que definir TSO\_TEMPLATE.

#### **TSO\_TEMPLATE=hlq.SFEKSAMP(FEJTSO)**

JCL de envoltura para someter el trabajo por medio de TSO. No hay ningún valor predeterminado. Esta sentencia hace referencia al nombre de miembro totalmente calificado del JCL que se debe usar como envoltura para el sometimiento TSO. Vea la sentencia SUBMITMETHOD para obtener más información.

**Nota:** En hlq.SFEKSAMP(FEJTSO) se proporciona un trabajo de envoltura de ejemplo. Consulte este miembro para obtener más información sobre la personalización que se necesita.

**Nota:** TSO\_TEMPLATE no tiene efecto si no se especifica también SUBMITMETHOD=TSO.

**Nota:** La definición de JESNAME ha dejado de ser necesaria. A partir de la versión 7.0, el supervisor de trabajos JES detecta automáticamente si el JES primario es JES2 o JES3.

---

## **Personalizar el JCL de arranque del supervisor de trabajos JES**

El supervisor de trabajos JES es el componente de Developer para System z que maneja toda la conectividad de JES. Para ello, el servidor debe estar activo en el sistema (ya sea como trabajo de usuario o como STC). Siga los pasos de personalización del JCL que se encuentran en hlq.SFEKSAMP(FEJJJCL) para crear el servidor del supervisor de trabajos JES según los estándares de su local.

**Nota:** Le recomendamos que copie el JCL de ejemplo en un nuevo conjunto de datos y que personalice la copia, para evitar que se sobrescriba al aplicar el mantenimiento.

## Rastreo del supervisor de trabajos JES

Debe activar el rastreo del supervisor de trabajos JES por motivos de diagnóstico, añadiendo “-TV” a la línea de PARM. El rastreo afectará negativamente al rendimiento y solo se debe activar cuando así lo indique el centro de soporte de IBM.

```
//JMONITOR EXEC PGM=FEJJMON,TIME=1440,REGION=0M,  
//          PARM=('POSIX(ON),ENVAR("_CEE_ENVFILE=DD:ENVIRON")/ -TV')
```

El rastreo también se puede controlar con los mandatos de la consola. Suponiendo que se utilice el nombre de trabajo JMON, el primer mandato de consola que figura más abajo activará el rastreo, y el segundo lo desactivará.

1. F JMON,APPL=-TV
2. F JMON,APPL=-TN

## Ejecutar el supervisor de trabajos JES como una tarea iniciada (STC)

El supervisor de trabajos JES se puede ejecutar como trabajo de usuario o como tarea iniciada (STC). Para definir que JMON sea una STC, hay que implementar las siguientes tareas:

1. No hace falta que el JCL tenga una tarjeta JOB (es preferible que no la tenga). La mayoría de las STC se inician con una tarjeta PROC, como en el ejemplo de: Figura 2.

```
//JMON      PROC HLQ=FEK,  
//          PRM=  
//*  
//* RD/Z JES JOB MONITOR  
//*  
//JMONITOR EXEC PGM=FEJJMON,TIME=1440,REGION=0M,  
//          PARM=('POSIX(ON),ENVAR("_CEE_ENVFILE=DD:ENVIRON")/&PRM')  
//STEPLIB  DD DISP=SHR,DSN=&HLQ..SFÉKLOAD  
//ENVIRON  DD DISP=SHR,DSN=&HLQ..SFEKSAMP(FEJJCNFG)  
//SYSPRINT DD SYSOUT=*  
//SYSABEND DD SYSOUT=*  
//SYSOUT   DD SYSOUT=*  
//SYSIN    DD DUMMY  
//*SYSTCPD DD DISP=SHR,DSN=SYS1.TCPIP.DATA
```

Figura 2. JCL del supervisor de trabajos JES

2. El JCL debe residir en una biblioteca de procedimientos del sistema (SYS1.PROCLIB es el valor predeterminado de IBM).  
Para que resulte fácil hacerle referencia, conviene que el nombre del miembro coincida con el nombre del procedimiento (JMON en el ejemplo anterior).
3. Conviene que las STC tengan un ID de usuario dedicado. Por cuestión de seguridad, este ID de usuario debe estar ‘protegido’, definiendo la palabra clave NOPASSWORD (en RACF). Esto quiere decir que en RACF fallará todo intento de inicio de sesión que exija una contraseña, pero sin revocar el ID de usuario.

Utilice el siguiente mandato de ejemplo para crear un ID de usuario de ese tipo, donde userid es el ID de usuario en cuestión, groupid es el grupo predeterminado y uid es el ID de UNIX.

```
ADDUSER userid DFLTGRP(groupid) NOPASSWORD OMVS(UID(uid) HOME(/tmp) PROGRAM(/bin/sh))
```

4. Las STC deben estar definidas ante el software de seguridad (por ejemplo, ante RACF). Hay varias formas de definir una STC, pero conviene hacerlo utilizando la clase STARTED. Para definir la clase STARTED, el administrador de seguridad emitiría los siguientes mandatos RACF:

- a. SETROPTS GENERIC(STARTED)
- b. RDEFINE STARTED \*\* STDATA(USER(=MEMBER))
- c. SETROPTS CLASSACT(STARTED)
- d. SETROPTS RACLIST(STARTED)

Para añadir un supervisor de trabajos JES como tarea iniciada (STC), se necesitan los siguientes mandatos RACF, donde jmon es el nombre del miembro JCL y userid es el ID de usuario cuyas autorizaciones se tienen que utilizar.

- a. RDEFINE STARTED jmon.\* STDATA(USER(userid))
- b. SETROPTS RACLIST(STARTED) REFRESH

**Nota:** Si añade la palabra clave TRUSTED(YES) al campo STDATA [STDATA(USER(userid) TRUSTED(YES))], se evitará la necesidad de definir los permisos individuales necesarios para el ID de usuario de la STC. RACF se salta las comprobaciones de seguridad de conjunto de datos en el caso de las STC de confianza. Sin embargo, antes de hacerlo, asegúrese de que no se pueda abusar de este ID de usuario, convirtiéndolo en protegido como se indicó anteriormente.

Consulte el manual *Security Server RACF Security Administrator's Guide* (SA22-7683) para obtener más información sobre las tareas iniciadas y sobre la seguridad.

5. Una vez completadas las tareas anteriores, el supervisor de trabajos JES ya se podrá iniciar y detener como tarea iniciada (STC).

El operador del sistema puede emitir los siguientes mandatos en la consola (siendo JMON el nombre de la STC del supervisor de trabajos JES):

- a. Start JMON
- b. stoP JMON
- c. Display A,JMON

Si posee la debida autorización, puede proporcionar estos mandatos desde dentro de SDSF, pero entonces hay que colocar una barra inclinada (/) antes de los mandatos. Consulte el manual *MVS System Commands* (SA22-7627) para obtener más información sobre los mandatos de la consola.

## Permisos de servidor

Para el ID de usuario asignado al servidor del supervisor de trabajos JES se necesita el acceso de lectura (READ) a la biblioteca de carga de Developer para System z, hlq.SFEKLOAD.

## Verificación del JCL de inicio del supervisor de trabajos JES

Inicie el trabajo de usuario o la STC definidos en los pasos anteriores. Si el trabajo finaliza con el código de retorno 66, sabrá que hlq.SFEKLOAD no está autorizada por APF.

---

## Acceso a spool JES y seguridad

### Acceso a spool condicional

Para que los usuarios puedan ejecutar operaciones por medio del supervisor de trabajos JES, hay que otorgarles autorización de acceso a la clase OPERCMDS. Esto se puede hacer condicionalmente, de manera que los derechos de acceso de los usuarios solo estén en vigor cuando utilicen el supervisor de trabajos JES. Para utilizar esta comprobación de acceso condicional, debe tener activa la clase CONSOLE y haber definido una consola llamada JMON (el nombre JMON es el único válido).

Por ejemplo, el administrador de seguridad emitiría los siguientes mandatos de RACF:

1. SETROPTS CLASSACT(CONSOLE)
2. RDEFINE CONSOLE JMON UACC(READ)
3. SETROPTS RACLIST(CONSOLE) REFRESH

Utilice los siguientes mandatos de RACF para permitir a los usuarios emitir mandatos JES2 solo a través del supervisor de trabajos JES, siendo id el ID de usuario o el ID de grupo de los usuarios de Developer para System z:

1. RDEFINE OPERCMDS JES2.\*\* UACC(NONE)
2. PERMIT JES2.\*\* CLASS(OPERCMDS) ID(id) ACCESS(CONTROL) WHEN(CONSOLE(JMON))
3. SETROPTS RACLIST(OPERCMDS) REFRESH
4. SETROPTS GENERIC(OPERCMDS) REFRESH

#### PRECAUCIÓN:

**El hecho de definir mandatos JES con el acceso universal NONE en su software de seguridad puede afectar a otras operaciones y aplicaciones. Debe someter a prueba este procedimiento antes de activarlo en un sistema de producción.**

### Mandatos disponibles

El supervisor de trabajos JES no proporciona a los usuarios de Developer para System z acceso pleno al spool JES. Los únicos mandatos disponibles son los que figuran en: Tabla 6. Para emitir los mandatos, se selecciona la opción pertinente en la estructura de menús del cliente (no hay indicador de mandatos). El ámbito de los mandatos está limitado adicionalmente por las técnicas descritas a continuación.

*Tabla 6. Mandatos de la consola del supervisor de trabajos JES*

Mandato	JES2	JES3
Retener	\$Hjobid	*F,J=jobid,H
Liberar	\$Ajobid	*F,J=jobid,R
Cancelar	\$Cjobid	*F,J=jobid,C
Purgar	\$Cjobid,P	*F,J=jobid,C

**Nota:** Si el usuario no es el propietario del archivo de spool, solo está permitido examinar. Todas las acciones que figuran en Tabla 6 provocarán un mensaje de tipo "No autorizado para el trabajo JOBxxx".

Aunque no posean autorización sobre estos mandatos de la consola, los usuarios todavía pueden someter trabajos y leer datos de salida de los trabajos, si se les otorga autorización de acceso a los archivos de spool.

## Limitar el acceso a los archivos de spool

Para limitar a los usuarios de forma que solo utilicen sus propios trabajos en el spool JES, defina la sentencia "LIMIT\_VIEW=USERID" en el archivo de configuración del supervisor de trabajos JES (FEJJCNFG). Si necesitan acceso a un rango de trabajos más amplio, pero no a todos, utilice las características de protección de archivo de spool estándar de su producto de seguridad, como la clase JESSPOOL, en RACF.

Al definir protección adicional, tenga presente que el supervisor de trabajos JES utiliza la **SAPI** (interfaz de programación de aplicaciones SYSOUT) para acceder al spool. Ello implica que el usuario necesita como mínimo el acceso de actualización (UPDATE) a los archivos de spool, incluso para la función de examen o navegación. Este requisito no es necesario si se ejecuta z/OS v1.7 (z/OS 1.8 para JES3) o superior. En este caso, el permiso de lectura (READ) es suficiente para la función de examen o navegación.

En el manual *Security Server RACF Security Administrator's Guide (SA22-7683)* hallará más información sobre la protección de archivos de spool JES.

---

## Personalizar procedimientos de construcción remota ELAXF\*

Developer para System z proporciona procedimientos JCL de ejemplo que se pueden usar para la generación de JCL, para las construcciones de proyectos remotos y las características de comprobación de sintaxis remota de mapas BMS de CICS, pantallas MFS de IMS y programas COBOL, PL/I, Assembler y C/C++.

Estos procedimientos permiten que las instalaciones apliquen estándares propios. Ello también garantizará que los desarrolladores utilicen los mismos procedimientos con las mismas opciones de compilador y los mismos niveles de compilador.

SMP/E instala los procedimientos JCL de ejemplo en hlq.SFEKSAMP. Si piensa utilizar estos procedimientos, debe:

1. Copiar todos los procedimientos que se llamen ELAXF\* en una biblioteca de procedimientos del sistema (como SYS1.PROCLIB) que esté disponible para los usuarios.
2. Los procedimientos copiados se deben personalizar para que reflejen los convenios de denominación empleados en el sistema destino. La personalización necesaria viene documentada dentro de cada procedimiento JCL.

Los procedimientos de ejemplo que hay que copiar y personalizar figuran en: Tabla 7.

*Tabla 7. Procedimientos ELAXF\* de ejemplo*

Miembro	Finalidad
ELAXFADT	Procedimiento de ejemplo para ensamblar y depurar programas assembler de alto nivel.
ELAXFASM	Procedimiento de ejemplo para ensamblar programas del ensamblador de alto nivel (HLASM).

Tabla 7. Procedimientos ELAXF\* de ejemplo (continuación)

Miembro	Finalidad
ELAXFBMS	Procedimiento de ejemplo para crear un objeto BMS CICS y el correspondiente miembro de inclusión, dsect o copia.
ELAXFCOC	Procedimiento de ejemplo para hacer compilaciones COBOL, conversiones CICS integradas y conversiones DB2 integradas.
ELAXFCOP	Procedimiento de ejemplo para hacer preproceso DB2 de sentencias SQL EXEC embebidas en programas COBOL.
ELAXFCOT	Procedimiento de ejemplo para hacer conversión CICS de sentencias CICS EXEC embebidas en programas COBOL.
ELAXFCPC	Procedimiento de ejemplo para hacer compilaciones C.
ELAXFCPP	Procedimiento de ejemplo para hacer compilaciones C++.
ELAXFGO	Procedimiento de ejemplo para el paso GO.
ELAXFLNK	Procedimiento de ejemplo para enlazar programas C/C++, COBOL, PLI y del ensamblador de alto nivel (HLASM).
ELAXFMFS	Procedimiento de ejemplo para crear pantallas MFS IMS.
ELAXFPLP	Procedimiento de ejemplo para hacer preproceso DB2 de sentencias SQL EXEC embebidas en programas PLI.
ELAXFPLT	Procedimiento de ejemplo para hacer conversión CICS de sentencias CICS EXEC embebidas en programas PLI.
ELAXFPL1	Procedimiento de ejemplo para hacer compilaciones PL/I, conversiones CICS integradas y conversiones DB2 integradas.
ELAXFUOP	Procedimiento de ejemplo para generar el paso UOPT al construir programas que se ejecutan en subsistemas CICS o IMS.

Tabla 8 contiene una lista de comprobación para los distintos calificadores de producto de alto nivel empleados en los procedimientos ELAXF\*.

Tabla 8. Lista de comprobación de calificadores de alto nivel de ELAXF\*

Producto	HLQ predeterminado	Valor
RD/z	FEK	
CICS	CICSTS31.CICS	
DB2	DSN810	
IMS	IMS	
COBOL	IGY.V3R4M0	
PL/I	IBMZ.V3R6M0	
C/C++	CBC	
LE	CEE	
LINKLIB del sistema	SYS1	
MACLIB del sistema	SYS1	

Si los procedimientos ELAXF\* no se pueden copiar en una biblioteca de procedimientos del sistema, cópielos en una biblioteca privada y pida a los usuarios de Developer para System z que añadan una tarjeta JCLLIB (justo después de la tarjeta JOB) a las propiedades del trabajo en el cliente. No



personalice el JCL de ejemplo en el conjunto de datos de instalación, porque el mantenimiento podría sustituir estos miembros y deshacer las personalizaciones que haya hecho.

```
//MYJOB JOB <parámetros del trabajo>  
//PROCS JCLLIB ORDER=(hlq.SFEKSAMP)
```

**Nota:** En la generación de JCL y/o las construcciones de proyectos remotos y características de comprobación de sintaxis remota realizadas desde un cliente Developer para System z se da por sentado que estos procedimientos se han personalizado y están disponibles para el usuario.

Los nombres de los procedimientos y los nombres de los pasos de cada procedimiento coinciden con las propiedades predeterminadas que vienen con el cliente. Si decide cambiar el nombre del procedimiento o el nombre de un paso del procedimiento, también hay que actualizar el correspondiente archivo de propiedades en el cliente. Le recomendamos que no cambie el nombre de los procedimientos ni de los pasos.

**Nota:** Habrá que pedir, instalar y configurar la herramienta de depuración IBM para que permita la depuración remota de los programas assembler, COBOL y PL/I. Consulte el manual *Rational Developer para System z Guía de planificación de host*, GI11-7839-00 (GI11-8296-00) para saber qué nivel de la herramienta de depuración se necesita para su versión de Developer para System z. La instalación y la personalización de este producto no se describe en este manual.

---

## (Opcional) Definir una transacción APPC para el servicio de mandatos TSO

En la versión 7.1, la tarea de definir la transacción APPC ha pasado a ser opcional. Por defecto, ahora el producto Developer para System z utiliza SCLM Developer Toolkit para proporcionar el servicio de mandatos TSO. La correspondiente configuración se realiza en `rsed.envvars`, que se describe en: “Personalizar `rsed.envvars`, el archivo de configuración de RSE” en la página 35.

**Nota:** El JCL del programa de transacción que APPC emplea para iniciar el servicio de mandatos TSO ha cambiado en la versión 7.1. Si anteriormente definió el servicio de mandatos TSO para capturar datos de salida ISPEXEC, debe definir una nueva transacción APPC o bien añadir la palabra clave NESTMACS a la línea PARM; por ejemplo:

```
// PARM='ISPSTART CMD(%FEKFRSRV TIMEOUT=60) NEWAPPL(ISR) NESTMACS'
```

El servicio de mandatos TSO se implementa como un programa de transacción APPC, FEKFRSRV, y debe estar activo para que las conexiones de Developer para System z se establezcan entre el host y el cliente. FEKFRSRV funciona a modo de servidor de hospedaje para ejecutar los mandatos TSO que se emiten desde la estación de trabajo por TCP/IP. No se necesita APPC en la estación de trabajo, porque esta se comunica con FEKFRSRV por TCP/IP. Cada estación de trabajo puede tener una conexión activa con múltiples hosts al mismo tiempo.

**Nota:** Si no está familiarizado con APPC, antes de seguir adelante con este apartado, lea: Apéndice F, “Configurar APPC”, en la página 119.



## Preparación

- Las siguientes tareas son un prerequisite y se deben realizar antes de configurar el servidor de mandatos TSO. La descripción de estas tareas se encuentra en los manuales mencionados.
  - Instale, configure e inicie VTAM en su sistema z/OS. Hallará más información en el manual *Communications Server bookshelf (F1A1BK61)*.
  - Instale, configure e inicie TCP/IP en el sistema z/OS. Hallará más información en: Apéndice C, “Configurar TCP/IP”, en la página 93, así como en el manual *Communications Server bookshelf (F1A1BK61)*.
  - Configure e inicie APPC y el subsistema del planificador de transacciones APPC (ASCH). Hallará más información en: Apéndice F, “Configurar APPC”, en la página 119, así como en el manual *MVS bookshelf (IEA2BK60)*.
- Para gestionar APPC mediante paneles ISPF, se puede usar el REXX de: Figura 3.

```
/* Administración de REXX -- APPC utilizando paneles ISPF */
address ISPEXEC
"LIBDEF ISPMLIB DATASET ID('ICQ.ICQMLIB') STACK"
"LIBDEF ISPLLIB DATASET ID('ICQ.ICQPLIB') STACK"
"LIBDEF ISPSLIB DATASET ID('ICQ.ICQSLIB') STACK"
"LIBDEF ISPTLIB DATASET ID('ICQ.ICQTLIB') STACK"
address TSO "ALTLIB ACT APPLICATION(CLIST)",
            "DSN('ICQ.ICQCCLIB') UNCOND QUIET"
"SELECT CMD(%ICQASRM0) NEWAPPL(ICQ) PASSLIB"
address TSO "ALTLIB DEACT APPLICATION(CLIST) QUIET"
"LIBDEF ISPMLIB"
"LIBDEF ISPLLIB"
"LIBDEF ISPSLIB"
"LIBDEF ISPTLIB"
exit
```

Figura 3. Paneles ISPF de REXX para APPC

**Nota:** Tenga en cuenta que puede desactivar la transacción APPC con esta herramienta; la transacción sigue ahí, pero no aceptará conexiones.

- Para definir la transacción APPC se necesitan conocimientos técnicos en diversos campos del sistema operativo MVS. Antes de seguir adelante, pida la opinión de administradores con experiencia, presentándoles la siguiente lista de comprobación.

Tabla 9. Lista de comprobación para la transacción APPC

Conocimientos técnicos	Información necesaria:	Valor
	• Valor predeterminado • Dónde encontrar la respuesta	
APPC	Nombre de conjunto de datos de TPDATA • Valor predeterminado: SYS1.APPCTP • El valor se encuentra en SYS1.PARMLIB(APPCPMxx)	
APPC	Nombre de transacción que hay que usar (puede no existir) • Valor predeterminado: FEKFRSRV • Para consultar las transacciones existentes, seleccione la opción de administración de perfiles TP en el menú ISPF de APPC	

Tabla 9. Lista de comprobación para la transacción APPC (continuación)

Conocimientos técnicos	Información necesaria:	Valor
	• Valor predeterminado • Dónde encontrar la respuesta	
APPC	Clase de transacción APPC que hay que usar • Valor predeterminado: A • Las clases APPC están definidas en SYS1.PARMLIB(ASCHPMxx)	
WLM/SRM	Grupo de rendimiento TSO y dominio • No hay valor predeterminado de IBM (depende del local)	
RACF	Cada usuario de Developer para System z tiene acceso a un segmento OMVS (es necesario) • No hay valor predeterminado de IBM (depende del local) • El mandato TSO de RACF LU <b>userid OMVS</b> se visualizará en un segmento OMVS personal existente	
RACF	Cada usuario de Developer para System z debe tener acceso de lectura (READ) a hlq.SFEKPROC(FEKFRSRV) • No hay valor predeterminado de IBM (depende del local) • El mandato TSO RACF LD <b>AUTHUSER DATASET('hlq.SFEKPROC.**')</b> visualizará los usuarios y grupos y su nivel de acceso para los conjuntos de datos cubiertos por el perfil de conjunto de datos	

En el manual *MVS Planning Workload Management (SA22-7602)* hallará más información sobre la gestión de WLM/SRM. Consulte la publicación *Security Server RACF Security Administrator's Guide (SA22-7683)* para obtener más información sobre los segmentos OMVS y los perfiles de protección de conjuntos de datos.

**Nota:** La clase de transacción APPC que se utilice debe tener suficientes iniciadores APPC para permitir un iniciador para cada usuario de Developer para System z.

**Nota:** La transacción APPC utiliza el REXX exec FEKFRSRV, situado en hlq.SFEKPROC. No cambie esta ubicación si desea que el posible mantenimiento SMP/E se active automáticamente.

## Implementación

El programador del sistema o el administrador de APPC debe llevar a cabo los siguientes pasos para configurar el servicio de mandatos:

1. Defina la información de planificación (clase) para el planificador de transacciones APPC si no va a utilizar una clase de transacción existente. Incluya una definición en SYS1.PARMLIB(ASCHPMxx) para la clase que el programa de transacción FEKFRSRV debe utilizar. Esta clase se emplea en el JCL de ejemplo hlq.SFEKSAMP(FEKAPPCC). Por lo tanto, la clase que hay en FEKAPPCC debe coincidir con la definida en SYS1.PARMLIB(ASCHPMxx). Por ejemplo:

```

CLASSADD
  CLASSNAME(A)
  MAX(20)
  MIN(1)
  MSGLIMIT(200)

```

**Nota:** Para el servicio de mandatos TSO, hay que especificar las especificaciones predeterminadas en las secciones OPTIONS y TPDEFAULT de SYS1.PARMLIB(ASCHPMxx). Hallará más información en: Apéndice F, “Configurar APPC”, en la página 119.

- Defina la transacción APPC que funcionará a modo de servidor de mandatos. Puede utilizar el JCL de ejemplo hlq.SFEKSAMP(FEKAPPCC) para definir esta transacción. Las instrucciones para personalizar este JCL se encuentran dentro del JCL. También se proporciona un JCL de ejemplo para visualizar, hlq.SFEKSAMP(FEKAPPCL), o suprimir, hlq.SFEKSAMP(FEKAPPCX), la transacción.

**Nota:** Si ha cambiado el nombre del programa de transacción (el valor predeterminado es FEKFRSRV), hay que asignar el nuevo nombre a \_FEKFSCMD\_TP\_NAME\_, en rsed.envvars, como se describe en: “Personalizar rsed.envvars, el archivo de configuración de RSE” en la página 35.

- Controle la prioridad de despacho del programa de transacción hlq.SFEKPROC(FEKFRSRV), asociando FEKFRSRV a un dominio y grupo de rendimiento en el gestor de cargas de trabajo (WLM). Dado que FEKFRSRV emite mandatos TSO, hay que asignarlo a un grupo de rendimiento TSO.
- Defina un segmento OMVS predeterminado para el sistema o bien uno individual para cada usuario que deba utilizar edición-compilación-depuración remota.
- Otorgue a los usuarios de Developer para System z acceso de lectura (READ) a hlq.SFEKPROC(FEKFSERV), el servidor de mandatos TSO.

**Nota:** La verificación de la instalación se llevará a cabo más tarde, durante la verificación de RSE. El motivo es que RSE implementa la conexión TCP/IP con el servidor de mandatos TSO.

---

## (Opcional) Personalizar miembros de procedimientos almacenados DB2 ELAXM\*

Se necesita la siguiente personalización para incorporar el procedimiento almacenado DB2 de ejemplo (constructor de procedimientos almacenados PL/I y COBOL) en su sistema. Para realizar estas tareas, necesitará ayuda de un administrador de WLM y un administrador de DB2.

Después de la aplicación de SMP/E, la biblioteca de ejemplos hlq.SFEKSAMP y la biblioteca de procedimientos hlq.SFEKPROC contienen los miembros de procedimientos almacenados DB2 que figuran en: Tabla 10.

*Tabla 10. Miembros de procedimientos almacenados DB2 ELAXM\* de ejemplo*

Miembro	Finalidad
hlq.SFEKSAMP(ELAXMJCL)	JCL de ejemplo para definir el constructor de procedimientos almacenados PL/I y COBOL ante DB2.
hlq.SFEKSAMP(ELAXMSAM)	Procedimiento JCL de ejemplo del espacio de direcciones WLM para el constructor de procedimientos almacenados PL/I y COBOL.

Tabla 10. Miembros de procedimientos almacenados DB2 ELAXM\* de ejemplo (continuación)

Miembro	Finalidad
hlq.SFEKPROC(ELAXMREX)	Código REXX para el constructor de procedimientos almacenados PL/I y COBOL.

**Nota:** El procedimiento almacenado DB2 utiliza el REXX exec ELAXMREX, situado en hlq.SFEKPROC. No cambie esta ubicación si desea que el posible mantenimiento SMP/E se active automáticamente.

**Nota:** Si desea cambiar el nombre de los miembros ELAXMSAM o ELAXMREX, vea: Apéndice A, "Ejecutar múltiples instancias de Developer para System z", en la página 73.

1. Copie ELAXMSAM en una biblioteca de procedimientos (como SYS1.PROCLIB) que esté disponible para los usuarios de los procedimientos almacenados DB2, y personalice el JCL, siguiendo las instrucciones de sus comentarios. Asegúrese de que la tarjeta DD SYSEXEC señala hacia la biblioteca en la que reside el miembro ELAXMREX (el valor predeterminado es hlq.SFEKPROC).
2. Utilice los paneles de gestión de la carga de trabajo (WLM) para asociar un entorno de aplicaciones al procedimiento JCL del espacio de direcciones WLM para el constructor de procedimientos almacenados PL/I y COBOL. En el manual *MVS Planning Workload Management (SA22-7602)* encontrará información sobre cómo hacerlo.

**Nota:** Puede crear un nuevo entorno de aplicaciones en WLM para el constructor de procedimientos almacenados PL/I y COBOL o bien añadir las definiciones necesarias a un entorno existente.

3. Copie ELAXMJCL en una biblioteca JCL privada, personalice la copia siguiendo las instrucciones de sus comentarios y pida a un administrador de DB2 que someta el trabajo. Asegúrese de que la cláusula WLM ENVIRONMENT de la sentencia CREATE PROCEDURE especifica el nombre del procedimiento del entorno WLM que se ha definido para el constructor de procedimientos almacenados PL/I y COBOL (valor predeterminado ELAXMSAM).

---

## (Opcional) Personalizar el soporte para idiomas bidireccionales CICS (bidi)

El componente Herramientas de servicio de empresa (EST) de Developer para System z admite diferentes formatos de mensaje de interfaz en árabe y hebreo, así como la presentación y edición de datos bidireccionales en todos los editores y vistas. En las aplicaciones de terminal, se soportan tanto las pantallas de izquierda a derecha como las pantallas de derecha a izquierda, así como los campos numéricos y los campos con orientación opuesta a la pantalla.

Las características y funciones bidireccionales adicionales son las siguientes:

- El peticionario de servicios EST especifica dinámicamente los atributos bidireccionales de los mensajes de interfaz.
- El proceso de datos bidireccionales en los flujos de servicio se basa en los atributos bidireccionales (tipo de texto, orientación del texto, intercambio numérico e intercambio simétrico). Estos atributos pueden especificarse en diferentes fases de la creación de flujos, tanto para flujos de interfaz como para flujos de terminal.

- El código de tiempo de ejecución generado por EST incluye la conversión de datos entre campos en mensajes que tienen diferentes atributos bidireccionales.

**Atención:** El módulo de carga (tanto el nombre como la codificación) ha cambiado desde los releases anteriores (anteriores de la versión 7.0). Si ha activado un release bidi anterior, los módulos de carga antiguos se tienen que eliminar de la concatenación RPL de CICS. Las ubicaciones predeterminadas son:

- FEJ.SFEJDLL(FEJBDTRN)
- FEJ.SFEJLMD(FEJBDTRE) (opcional, versión no PDSE de FEJBDTRN para el hebreo)

Necesitará ayuda del administrador de CICS para realizar las siguientes tareas:

1. Coloque el programa hlq.SFEKLOAD(FEJBDTRX) en la concatenación RPL de CICS (sentencia DD DFHRPL). Le recomendamos que lo haga añadiendo el conjunto de datos de instalación a la concatenación para que el mantenimiento aplicado esté automáticamente disponible para CICS.

Las transformaciones bidireccionales de EST las realiza el módulo FEJBDTRX en el entorno de tiempo de ejecución de flujo de servicios (SFR) CICS. Se llama al módulo FEJBDTRX cuando hay que hacer conversiones bidireccionales en el código generado por EST (cuando se generan correlaciones entre campos de mensajes que tienen atributos bidireccionales distintos).

**Nota:** Si no concatena el directorio de instalación, sino que copia el módulo FEJBDTRX en un conjunto de datos nuevo o existente, no olvide que este módulo es una DLL y DEBE residir en una biblioteca PDSE.

**Nota:** En la versión 7.1, el programa hlq.SFEKDLL(FEJBDTRX) se ha trasladado a una nueva biblioteca de carga, hlq.SFEKLOAD(FEJBDTRX).

2. Pida al administrador de CICS que establezca que autoinstall sea igual a autoactive.

Si autoinstall tiene el valor autoInactive, defina el programa FEJBDTRX en CICS utilizando el mandato CEDA; por ejemplo:

```
CEDA DEF PROG(FEJBDTRX) LANG(LE) G(XXX)
```

Además, el código generado por EST puede soportar la transformación bidi en entornos distintos de SFR de CICS (por ejemplo, en aplicaciones por lotes). Puede hacer que los generadores de EST incluyan llamadas a las rutinas de conversión bidireccional, especificando las opciones de transformación bidi pertinentes en los asistentes de generación de EST y enlazando los programas generados con la biblioteca de conversión bidireccional pertinente, hlq.SFEKLOAD.

**Atención:** El código bidi creado con los releases anteriores (anteriores a la versión 7.0) se deben RECOMPILAR para poder usar el nuevo módulo FEJBDTRX.

---

## (Opcional) Personalizar el gestor de despliegue de aplicaciones (ADM)

El gestor de despliegue de aplicaciones (ADM) proporciona un enfoque de despliegue común y una API de despliegue para todos los componentes de Developer para System z.

Además, el ADM proporciona un cliente y un servidor (basado en el host) de definición de recursos CICS (CRD), que aporta las siguientes funciones:

1. Permitir a los desarrolladores de aplicaciones definir los recursos CICS de manera segura, controlada y limitada.

- Los valores predeterminados de las definiciones de recursos CICS los suministra el administrador de CICS y se almacenan en el repositorio del servidor CRD en el host.
  - La capacidad de actualización de atributos de recurso está controlada por el administrador de CICS (actualizar, proteger, oculto).
  - CRD está limitado a un pequeño conjunto de recursos, que son los que normalmente necesitan los desarrolladores de aplicaciones. Son los siguientes: DB2Tran, Doctemplate, File, Mapset, Processtype, Program, TDQ y Transaction.
  - La autorización para crear definiciones de recursos CICS está controlada por RACF o por otro gestor de seguridad externo.
2. Impedir el acceso de desarrollo CICS a los conjuntos de datos VSAM no autorizados o incorrectos, proporcionando el control del administrador de CICS sobre el atributo de nombre de conjunto de datos físico en las definiciones de archivo. Esta información de enlace se almacena en el repositorio CRD en el host.
  3. Ayudas varias para el desarrollo de servidor CRD.
    - Copia nueva de programas y conjuntos de mapas.
    - Listar regiones CICS.
    - Lista de DFHRPL.
  4. Ayudas varias para el desarrollo de servicios Web del servidor CRD.
    - Realizar exploración de conductos para instalar automáticamente definiciones URIMAP y WEBSERVICE.
    - Proporcionar lista de directorios pickup de WSBind y conductos.
    - Proporcionar lista de directorios de archivos WSDL.
    - Proporcionar lista de URI de punto final.

Developer para System z suministra tres transacciones que el servidor CRD utiliza al definir y consultar recursos CICS. Se proporciona código fuente COBOL para permitir personalizaciones específicas del local.

#### **ADMD**

Para las peticiones que establecen valores predeterminados de servicios Web y recursos CICS. Normalmente, está destinado a los administradores de CICS.

**ADMI** Para las peticiones que definen, instalan o desinstalan recursos CICS.

#### **ADMR**

Para las demás peticiones que recuperan información de recursos o de entorno CICS.

En el manual *Rational Developer para System z Application Deployment Manager User's Guide* (SC23-7661) encontrará más información sobre ADM.

Para activar el servidor CRD, hay que realizar los siguientes pasos de personalización.

**Nota:** Necesitará ayuda del administrador de CICS para realizar algunas de las siguientes tareas.

Antes de instalar la versión 7.1.1, si es usted un usuario anterior del servidor CRD, le recomendamos que guarde la personalización relacionada, como se describe en: “Hacer copia de seguridad de archivos configurados con anterioridad” en la página 11.

Copie los miembros que hay que personalizar del directorio de instalación en una biblioteca personal y personalice esas copias, para evitar que se sobrescriban al aplicar el mantenimiento:

- hlq.SFEKSAMP(ADNVSAM)
- hlq.SFEKSAMP(ADNPCCSD)

opcional (manejador de mensajes de conducto)

- hlq.SFEKSAMP(ADNCMSGH)
- hlq.SFEKSAMP(ADNSMSGH)

opcional (actualización de CSD para regiones no primarias)

- hlq.SFEKSAMP(ADNARCSO)

## Repositorio CRD

Personalice y someta el trabajo ADNVSAM para asignar e inicializar el archivo VSAM del repositorio del servidor CRD. Las instrucciones de personalización están en la documentación de ADNVSAM.

Le aconsejamos que cree un repositorio aparte para cada región de conexión primaria CICS. El hecho de compartir el repositorio implica que todas las regiones CICS relacionadas utilizarán los mismos valores almacenados en el repositorio, y que múltiples espacios de direcciones se escribirán en el VSAM, que se debe configurar correctamente para poder manejarlos.

**Nota:** Si no se le indica lo contrario, el repositorio del servidor CRD actual (que contiene sus valores personalizados) se puede reutilizar en los distintos releases de Developer para System z.

## Región de conexión primaria CICS

El servidor CRD debe estar definido en la región de conexión primaria. Es la región que procesará las peticiones de servicios Web procedentes de Developer para System z.

- Coloque los módulos de carga ADM hlq.SFEKLOAD(ADNCRDS) y hlq.SFEKLOAD(ADNCRDR) en la concatenación RPL de CICS (sentencia DD DFHRPL) de la región de conexión primaria CICS. Le recomendamos que lo haga añadiendo el conjunto de datos de instalación a la concatenación para que el mantenimiento aplicado esté automáticamente disponible para CICS.
- Personalice y someta el trabajo ADNPCCSD para que actualice la definición de sistema CICS de la región de conexión primaria CICS. Las instrucciones de personalización están en la documentación de ADNPCCSD.
- Utilice el mandato CEDA pertinente para instalar el grupo ADM de esta región; por ejemplo:  
CEDA INSTALL GROUP(ADNPCRGP)

## Manejador de mensajes de conducto

El manejador de mensajes de conducto (ADNSMSGH) se utiliza para la seguridad, procesando el ID de usuario y la contraseña en la cabecera SOAP. El archivo de



configuración del conducto hace referencia al ADNSMSGH, que, por lo tanto, se debe colocar en la concatenación RPL de CICS. El ADNSMSGH también se utiliza para establecer que el ID de transacción sea igual a ADMD, ADMR o ADML, en función de la operación solicitada. Es posible que le interese personalizar el ADNSMSGH para que utilice distintos ID de transacción.

Utilizando el valor predeterminado:

- Coloque el módulo de carga de hlq.SFEKLOAD(ADNSMSGH) en la concatenación RPL de CICS (sentencia DD DFHRPL) de la región de conexión primaria CICS. Le recomendamos que lo haga añadiendo el conjunto de datos de instalación a la concatenación para que el mantenimiento aplicado esté automáticamente disponible para CICS.

Personalizando el ADNSMSGH:

- Personalice el código fuente de ADNSMSGH de ejemplo (COBOL) del manejador de mensajes de conducto.
- Personalice y someta el trabajo ADNCMSGH para compilar el fuente de ADNSMSGH personalizado. Las instrucciones de personalización están en la documentación de ADNCMSGH.
- Coloque el módulo de carga ADNSMSGH resultante en la concatenación RPL de CICS (sentencia DD DFHRPL) de la región de conexión primaria CICS.

**Nota:** Asegúrese de que el módulo de carga de ADNSMSGH personalizado se coloque antes de hacer ninguna referencia a hlq.SFEKLOAD, pues de lo contrario se utilizaría el valor predeterminado.

## (Opcional) Regiones de conexión no primarias CICS

El servidor CRD también se puede usar con una o más regiones de conexión no primarias, que suelen ser regiones propietarias de aplicación (AOR).

**Nota:** No hace falta realizar estos pasos si se utiliza CICSplex SM para gestionar el entorno CICS.

- Coloque el módulo de carga ADM hlq.SFEKLOAD(ADNCRDS) en la concatenación RPL de CICS (sentencia DD DFHRPL) de estas regiones de conexión no primarias. Le recomendamos que lo haga añadiendo el conjunto de datos de instalación a la concatenación para que el mantenimiento aplicado esté automáticamente disponible para CICS.
- Personalice y someta el trabajo ADNARCSD para que actualice la CSD de otras regiones de conexión no primarias. Las instrucciones de personalización están en la documentación de ADNARCSD.
- Utilice el mandato CEDA pertinente para instalar el grupo ADM de estas regiones; por ejemplo:  
CEDA INSTALL GROUP(ADNARRGP)



---

## Capítulo 4. Activar componentes z/OS UNIX de Developer para System z

Antes de instalar la versión 7.1.1, si es usted un usuario anterior de Developer para System z, le recomendamos que guarde la personalización relacionada descrita en: "Hacer copia de seguridad de archivos configurados con anterioridad" en la página 11.

Si no está familiarizado con z/OS UNIX, le aconsejamos que pida ayuda a un administrador de z/OS UNIX con experiencia o a otro administrador de UNIX para realizar las tareas que figuran en este capítulo.

Los mandatos de z/OS UNIX necesarios para realizar las tareas enumeradas se describen de manera resumida para su comodidad. A menos que se le indique lo contrario, consulte el manual *UNIX System Services Command Reference (SA22-7802)* para obtener más información sobre estos mandatos.

En las tareas que se describen a continuación, se espera que esté activo en z/OS UNIX. Para ello, emita el mandato TSO **OMVS**. Utilice el mandato **exit** para volver a TSO.

MVS proporciona la posibilidad de editar archivos de z/OS UNIX utilizando ISPF mediante el mandato **OEDIT**. Este mandato se puede usar en TSO y también en OMVS.

La mayoría de los archivos de z/OS UNIX tienen el permiso de escritura restringido al propietario del archivo. Esta restricción se puede eludir de muchas maneras.

- **UID 0**  
Esta configuración no está recomendada para los ID de usuario "human", porque no hay restricciones relacionadas con z/OS UNIX.
- **Acceso de lectura (READ) al perfil BPX.SUPERUSER de la clase FACILITY**  
Permite que el usuario se convierta en UID 0 con el mandato **su**. Esta es la configuración recomendada.
- **Acceso de actualización (UPDATE) al perfil SUPERUSER.FILESYS de la clase UNIXPRIV**  
Permite que el usuario lea/escriba cualquier archivo, y también que lea o busque en cualquier directorio. El acceso de CONTROL (o superior) a este perfil de seguridad añade el permiso de escritura en cualquier directorio a la lista de permisos.

Consulte el manual *UNIX System Services Planning (GA22-7800)* para obtener más información sobre la seguridad en z/OS UNIX.

En este capítulo, todas las sentencias que incluyen la vía de acceso `/usr/lpp/wd4z/` hacen referencia a la vía de acceso que se utilizó durante la instalación de Developer para System z. El valor predeterminado es `/usr/lpp/wd4z/`, pero quizá no sea válido para su local.

**Nota:** Developer para System z depende de que TCP/IP tenga el nombre de host correcto en el momento de la inicialización. Ello implica que los distintos

archivos de configuración de TCP/IP y del resolvente estén configurados correctamente. Hallará más información sobre la personalización necesaria en: Apéndice C, “Configurar TCP/IP”, en la página 93.

Puede probar la configuración de TCP/IP con el mandato TSO **HOMETEST**. En el manual *Communications Server IP System Administrator's Commands* (SC31-8781) hallará más información sobre este mandato.

**Nota:** Developer para System z depende de INETD a la hora de poner a punto la conexión entre cliente y host. Para obtener más información sobre INETD, consulte: Apéndice D, “Configurar INETD”, en la página 101.

**Nota:** Para las acciones remotas (basadas en host) de los subproyectos z/OS UNIX, es necesario que REXEC o SSH esté activo en el host.

**Nota:** Se necesita Java de 31 bites, y todos los usuarios de z/OS UNIX deben tener acceso para ejecutar (EXECUTE) y leer (READ) en los directorios del HFS Java.

**Atención:** No se puede utilizar Java de 64 bites.

Las publicaciones que le pueden ayudar a comprender z/OS UNIX son las siguientes:

- *ABCs of z/OS System Programming Volumen 9 (Redbook SG24-6989)*
- *UNIX System Services Planning (GA22-7800)*
- *UNIX System Services User's Guide (SA22-7801)*

---

## Guardar el archivo de configuración rsed.envvars en otro directorio

Le recomendamos que copie el archivo `/usr/lpp/wd4z/rse/lib/rsed.envvars` en un nuevo directorio (como `/etc/wd4z/`) y que personalice la copia para evitar que se sobrescriba al aplicar el mantenimiento. Sin embargo, si lo hace así, debe copiar asimismo los siguientes archivos del directorio de instalación (el valor predeterminado es `/usr/lpp/wd4z/rse/lib/`) en la nueva ubicación:

1. `rsed.envvars`
2. `rsecomm.properties`
3. `ssl.properties`
4. `setup.env.zseries`
5. `server.zseries`

**Nota:** No se necesita ninguna personalización para los archivos `*.zseries`, pero es importante que sustituya las versiones anteriores por las actuales. Es para mantenerlas en sincronización con el archivo `rsed.envvars` actual.

También tendrá que copiar los archivos enumerados en: Tabla 11, si piensa utilizar las características opcionales que forman parte de ellos:

Tabla 11. Archivos de configuración opcionales

archivo	función
projectcfg.properties	Proyectos basados en host  Vea: “(Opcional) Personalizar la configuración de proyectos de host, projectcfg.properties” en la página 54

Tabla 11. Archivos de configuración opcionales (continuación)

archivo	función
FMIEXT.properties	Integración del gestor de archivos  Vea: “(Opcional) Personalizar la integración del gestor de archivos, FMIEXT.properties” en la página 54
CRASRV.properties	CARMA  Vea: Capítulo 5, “(Opcional) Activar el gestor de repositorios de acceso común (CARMA) de IBM”, en la página 57

Los siguientes mandatos de ejemplo,

1. `mkdir /etc/wd4z`
2. `cd /usr/lpp/wd4z/rse/lib`
3. `cp rsed.envvars /etc/wd4z`

crean un directorio llamado `/etc/wd4z` y copian `rsed.envvars` del directorio actual en `/etc/wd4z`. Repita el mandato de copiar para los archivos restantes.

El resultado de la copia se puede verificar con el mandato `ls /etc/wd4z`, cuya salida debe ser parecida a esta (\$ es el indicador de z/OS UNIX):

```
$ ls /etc/wd4z
/etc/wd4z
rsecomm.properties  server.zseries      ssl.properties
rsed.envvars         setup.env.zseries
```

**Nota:** Si desea conservar todos los archivos de Developer para System z en el mismo HFS (privado), pero desea asimismo colocar los archivos de configuración en `/etc/wd4z`, puede utilizar enlaces simbólicos para resolver este problema. Los siguientes mandatos de ejemplo crean un nuevo directorio (`/usr/lpp/wd4z/rse/lib/cust`) en el HFS de instalación y definen un enlace simbólico (`/etc/wd4z`) que lleva a él:

1. `mkdir /usr/lpp/wd4z/rse/lib/cust`
2. `ln -s /usr/lpp/wd4z/rse/lib/cust /etc/wd4z`

## Personalizar `rsed.envvars`, el archivo de configuración de RSE

Todos los métodos de conexión de cliente de Developer para System z utilizan las variables establecidas en el archivo `rsed.envvars`, situado por defecto en el directorio de instalación, `/usr/lpp/wd4z/rse/lib/`, pero se podrían copiar en otro directorio en el paso anterior. El archivo de ejemplo proporcionado tiene las sentencias enumeradas en: Figura 4 en la página 36, donde las líneas de comentarios empiezan por un signo de almohadilla (#).

```

#####
# (1) personalizaciones necesarias
JAVA_HOME=/usr/lpp/java/J1.4
RSE_HOME=/usr/lpp/wd4z
TZ=EST5EDT
LANG=C
PATH=/bin:/usr/sbin:.
_RSE_CLASS_OPTS=""
_RSE_JAVAOPTS=""
#_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DTSO_SERVER=APPC"
#####
# (2) personalizaciones necesarias si se utiliza SCLMDT
_CMDSERV_BASE_HOME=/usr/lpp/SCLMDT
_CMDSERV_BASE_LOAD=BWB.SBWBLOAD
_CMDSERV_CONF_HOME=/etc/SCLMDT
_CMDSERV_WORK_HOME=/var/SCLMDT
STEPLIB=NONE
#STEPLIB=$_CMDSERV_BASE_LOAD
_RSE_CMDSERV_OPTS=""
#####
# (3) personalizaciones opcionales
#_RSE_PORTRANGE=8108-8118
#_FEKFLOCK_USERID=id_usuario
#_FEKFLOCK_JOBNAME=nombre_trabajo
#_FEKFSCMD_TP_NAME=nombre_tp
#_FEKFSCMD_PARTNER_LU=nombre_lu
#####
# (4) no lo cambie a menos que así se lo indique el centro de soporte de IBM
RSE_LIB=$RSE_HOME/rse/lib
ICU_LIB=$RSE_HOME/icuc/lib
_CEE_RUNOPTS="ALL31(ON) HEAP(32M,32K,ANYWHERE,KEEP,,) TRAP(ON)"
_CEE_DMPTARG=$HOME/.eclipse/RSE/$RSE_USER_ID
_BPX_SHAREAS=YES
_BPX_SPAWN_SCRIPT=YES
PATH=$JAVA_HOME/bin:$RSE_LIB:$_CMDSERV_BASE_HOME/bin:$PATH
LIBPATH=$JAVA_HOME/bin:$JAVA_HOME/bin/classic:$ICU_LIB:$RSE_LIB:.
CLASSPATH=$RSE_LIB:$RSE_LIB/dstore_core.jar:$RSE_LIB/clientserver.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/dstore_extra_server.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/dstore_miners.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/universalminers.jar:$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:$RSE_LIB/FAMiner.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/mvsutil.jar:$RSE_LIB/jesutils.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/lucene-1.4.3.jar:$RSE_LIB/cdtparser.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/wdzBidi.jar:$RSE_LIB/fmiExtensions.jar
CLASSPATH=.:$CLASSPATH
_RSE_CMDSERV_OPTS="$&SESSION=SPAWN$_RSE_CMDSERV_OPTS"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DSCLMDT_OPTS='$_RSE_CMDSERV_OPTS'"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DA_PLUGIN_PATH=$RSE_LIB"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Xbootclasspath/p:$RSE_LIB/bidiTools.jar"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Dcom.ibm.cacheLocalHost=true"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -showversion"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Duser.home=$HOME"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Dclient.username=$RSE_USER_ID"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS $_RSE_CLASS_OPTS"
_RSE_SERVER_CLASS=com.ibm.etools.systems.dstore.core.server.Server
_RSE_SERVER_TIMEOUT=120000
#####
# (5) Variables de entorno adicionales

```

*Figura 4. rsed.envvars – archivo de configuración de RSE*

Las definiciones que se necesitan son las siguientes:

## **JAVA\_HOME**

Directorio inicial Java. El valor predeterminado es /usr/lpp/java/J1.4. Cámbielo para que coincida con su instalación de Java.

## **RSE\_HOME**

Directorio inicial de RSE. El valor predeterminado es /usr/lpp/wd4z. Cámbielo para que coincida con su instalación de Developer para System z.

**TZ** Selector de huso horario. El valor predeterminado es EST5EDT. El huso horario predeterminado es UTC +5 horas (horario de verano según la hora estándar del este (EST)). Cambie este valor para que coincida con su huso horario. Hallará información adicional en el manual found in the *UNIX System Services File System Interface Reference (SA22-7808)*.

## **LANG**

Especifica el nombre del entorno local predeterminado. El valor predeterminado es C. C especifica el entorno local de POSIX, y Ja\_JP especifica el entorno local japonés. Cámbielo para que coincida con su entorno local.

**PATH** Vía de acceso del mandato. El valor predeterminado es /bin:/usr/sbin:.. Se puede cambiar, si es necesario.

## **\_RSE\_CLASS\_OPTS**

Opciones Java adicionales para compartir clases. El valor predeterminado es "". Para obtener más información sobre esta definición, vea: "(Opcional) Definir parámetros de arranque Java adicionales con \_RSE\_\*OPTS" en la página 41.

## **\_RSE\_JAVAOPTS**

Opciones Java adicionales específicas del RSE. El valor predeterminado es "". Para obtener más información sobre esta definición, vea: "(Opcional) Definir parámetros de arranque Java adicionales con \_RSE\_\*OPTS" en la página 41.

Developer para System z utiliza SCLM Developer Toolkit por defecto para el servicio de mandatos TSO. Se utiliza APPC cuando la siguiente opción \_RSE\_JAVAOPTS no está comentada:

```
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DTSO_SERVER=APPC"
```

**Nota:** Para ambos métodos del servicio de mandatos TSO, se necesitan más personalizaciones que las que figuran en rsed.envvars. Las personalizaciones necesarias para la instalación de APPC se describen en: "(Opcional) Definir una transacción APPC para el servicio de mandatos TSO" en la página 24, y las que se necesitan para SCLMDT se describen en: "Personalizar el archivo de configuración ISPF, ISPF.conf" en la página 46.

A continuación se indican las definiciones que se necesitan si se utiliza SCLM Developer Toolkit, ya sea para el servicio de mandatos TSO o al instalar el plug-in SCLMDT en el cliente Developer para System z.

## **\_CMDSERV\_BASE\_HOME**

Directorio inicial de SCLM Developer Toolkit. El valor predeterminado es /usr/lpp/SCLMDT. Cámbielo para que coincida con su instalación de SCLM Developer Toolkit. Esta directiva solo se necesita cuando se utiliza el SCLM Developer Toolkit (servicio de mandatos TSO o plug-in de cliente).

#### **\_CMDSERV\_BASE\_LOAD**

Biblioteca de carga de SCLM Developer Toolkit. El valor predeterminado es BWB.SBWBLOAD. Cámbielo para que coincida con su instalación de SCLM Developer Toolkit. Esta directiva solo se necesita cuando se utiliza el SCLM Developer Toolkit (servicio de mandatos TSO o plug-in de cliente).

#### **\_CMDSERV\_CONF\_HOME**

Directorio de configuración base de SCLM Developer Toolkit. El valor predeterminado es /etc/SCLMDT. Cámbielo para que coincida con su personalización de SCLM Developer Toolkit. Esta directiva solo se necesita cuando se utiliza el SCLM Developer Toolkit (servicio de mandatos TSO o plug-in de cliente).

#### **\_CMDSERV\_WORK\_HOME**

Directorio de trabajo base de SCLM Developer Toolkit. El valor predeterminado es /var/SCLMDT. Cámbielo para que coincida con su personalización de SCLM Developer Toolkit. Esta directiva solo se necesita cuando se utiliza el SCLM Developer Toolkit (servicio de mandatos TSO o plug-in de cliente).

#### **STEPLIB**

STEPLIB del servidor RSE. El valor predeterminado es NONE. No cambie esta línea, porque funciona a modo de valor predeterminado.

Por defecto, Developer para System z utiliza la LINKLIST para acceder a la biblioteca de carga de SCLM Developer Toolkit. Se utiliza STEPLIB cuando la siguiente directiva STEPLIB no está comentada:

```
STEPLIB=$_CMDSERV_BASE_LOAD
```

**Nota:** El hecho de utilizar STEPLIB en z/OS UNIX afecta negativamente al rendimiento, como se describe en: “Evitar el uso de STEPLIB” en la página 67.

#### **\_RSE\_CMDSERV\_OPTS**

Opciones Java adicionales específicas del servicio de mandatos TSO. El valor predeterminado es "". Para obtener más información sobre esta definición, vea: “(Opcional) Definir parámetros de arranque Java adicionales con \_RSE\_\*OPTS” en la página 41. Esta directiva solo se necesita cuando se utiliza el SCLM Developer Toolkit (servicio de mandatos TSO o plug-in de cliente).

Las definiciones que figuran a continuación son opcionales. Si se omiten, se emplearán los valores predeterminados.

#### **\_RSE\_PORTRANGE**

Especifica el rango de puertos que el servidor RSE puede abrir para establecer comunicación con un cliente. Por defecto, se puede usar cualquier puerto. Para obtener más información sobre esta definición, vea: “(Opcional) Definir el rango de puertos (PORTRANGE) disponibles para RSE” en la página 40.

#### **\_FEKFLOCK\_USERID\_**

ID de usuario que el gestor de bloqueos debe utilizar. El valor predeterminado es el ID de usuario de inicio de sesión.

#### **\_FEKFLOCK\_JOBNAME\_**

Nombre de usuario que el gestor de bloqueos debe utilizar. El valor predeterminado es FEKFLK00.

#### **\_FEKFSCMD\_TP\_NAME\_**

Nombre del programa de transacción APPC. El valor predeterminado es FEKFRSRV. Descomente y cambie esta definición si no ha utilizado el nombre de programa de transacción predeterminado al definir la transacción APPC. Para obtener más información sobre la transacción APPC, vea: “(Opcional) Definir una transacción APPC para el servicio de mandatos TSO” en la página 24.

#### **\_FEKFSCMD\_PARTNER\_LU\_**

Forzar RSE para que utilice esta LU base de APPC. Para obtener más información sobre esta definición, vea: Apéndice F, “Configurar APPC”, en la página 119.

Las siguientes definiciones son necesarias y no se deben cambiar, a menos que así lo indique el centro de soporte de IBM.

#### **RSE\_LIB**

Vía de acceso de la biblioteca del RSE. El valor predeterminado es \$RSE\_HOME/rse/lib. No lo modifique.

#### **ICU\_LIB**

Vía de acceso de la biblioteca de componentes internacionales para Unicode (ICU). El valor predeterminado es \$RSE\_HOME/icuc/lib. No lo modifique.

#### **\_CEE\_RUNOPTS**

Opciones de tiempo de ejecución Language Environment (LE) empleadas por los procesos iniciados. El valor predeterminado es "ALL31(ON) HEAP(32M,32K,ANYWHERE,KEEP,,) TRAP(ON)". No lo modifique.

#### **\_CEE\_DMPTARG**

Ubicación de vuelcos de Language Environment (LE) z/OS UNIX utilizada por la máquina virtual Java (JVM). El valor predeterminado es \$HOME/.eclipse/RSE/\$RSE\_USER\_ID. No lo modifique.

#### **\_BPX\_SHAREAS**

Ejecutar procesos en primer plano en el mismo espacio de direcciones que la shell. El valor predeterminado es YES. No lo modifique.

#### **\_BPX\_SPAWN\_SCRIPT**

Ejecutar scripts de shell directamente desde la función spawn(). El valor predeterminado es YES. No lo modifique.

**PATH** Vía de acceso del mandato. El valor predeterminado es \$JAVA\_HOME/bin:\$RSE\_LIB:\$CMDSEV\_BASE\_HOME/lib:\$PATH. No lo modifique.

#### **LIBPATH**

Vía de acceso de la biblioteca. El valor predeterminado es \$JAVA\_HOME/bin:\$JAVA\_HOME/bin/classic:\$ICU\_LIB:\$RSE\_LIB:. No lo modifique.

#### **CLASSPATH**

Vía de acceso de clases. El valor predeterminado es demasiado largo para repetirlo. No lo modifique.

#### **\_RSE\_CMDSEV\_OPTS**

Opciones Java adicionales específicas del servicio de mandatos TSO. El valor predeterminado es "&SESSION=SPAWN\$\_RSE\_CMDSEV\_OPTS". No lo modifique.



### **`_RSE_JAVAOPTS`**

Opciones Java adicionales específicas del RSE. El valor predeterminado es demasiado largo para repetirlo. No lo modifique.

### **`_RSE_SERVER_CLASS`**

Clase Java para el servidor RSE. El valor predeterminado es `com.ibm.etools.systems.dstore.core.server.Server`. No lo modifique.

### **`_RSE_SERVER_TIMEOUT`**

Valor de tiempo de espera para el servidor RSE (en espera en el cliente) en milisegundos. El valor predeterminado es 120000 (2 minutos). No lo modifique.

**Nota:** Puede eludir la necesidad de tener bibliotecas de C/C++ y Language Environment (LE) en LINKLIST, añadiendo la siguiente sentencia STEPLIB al final (END) de `rsed.envvars` (los nombres de conjuntos de datos pueden ser distintos en su local). Sin embargo, tenga presente que el hecho de utilizar STEPLIB en z/OS UNIX afecta negativamente al rendimiento, como se describe en: “Evitar el uso de STEPLIB” en la página 67.

- Si la última directiva STEPLIB definida anteriormente en `rsed.envvars` es igual a STEPLIB=NONE  
STEPLIB=CEE.SCEERUN:CEE.SCEERUN2:CBC.SCLBDLL
- Si la última directiva STEPLIB definida anteriormente en `rsed.envvars` no es igual a STEPLIB=NONE  
STEPLIB=\$STEPLIB:CEE.SCEERUN:CEE.SCEERUN2:CBC.SCLBDLL

**Nota:** Se permiten enlaces simbólicos al especificar directorios en `rsed.envvars`.

## **(Opcional) Definir el rango de puertos (PORTRANGE) disponibles para RSE**

Esta es una parte de la personalización de `rsed.envvars` que especifica los puertos en los que el servidor RSE se puede comunicar con el cliente. Este rango de puertos no tiene conexión con los puertos REXEC/SSH o del daemon RSE.

Para ayudarle a comprender la utilización de los puertos, se proporciona esta descripción corta del proceso de conexión del RSE:

1. El cliente se conecta al puerto de host 4035 (servicio del daemon RSE de INETD) o al puerto de host 512 (servicio REXEC de INETD) o al puerto de host 22 (servicio SSH de INETD).
2. El servicio INETD elegido crea un proceso RSE.
3. El proceso RSE abre un puerto de host para que el cliente se conecte. La selección de este puerto la puede configurar el usuario, ya sea en el cliente, en la pestaña de propiedades de subsistema (método no recomendado) o mediante la definición de `_RSE_PORTRANGE` en el archivo `rsed.envvars`.
4. El servicio INETD devuelve el número de puerto al cliente.
5. El cliente se conecta al puerto del host.

Para especificar el rango de puertos, para que el cliente se comunice con z/OS, descomente y cambie la siguiente línea del archivo `rsed.envvars`:

```
#_RSE_PORTRANGE=8108-8118
```



**Nota:** Antes de seleccionar un rango de puertos, verifique que el rango está disponible en su sistema, sirviéndose de los mandatos **NETSTAT** y **NETSTAT PORTL**. Hallará más información en: “Puertos TCP/IP reservados” en la página 83.

El formato de PORTRANGE es: `_RSE_PORTRANGE=min-max` (el máximo no es inclusive; por ejemplo, `_RSE_PORTRANGE=8108-8118` significa que los puertos utilizables son los comprendidos entre los números 8108 y 8117). El número de puerto que el servidor RSE utiliza se determina en el siguiente orden:

1. Si hay un número de puerto distinto de cero especificado en las propiedades de subsistema del cliente, ese es el número que se utilizará. Si el puerto no está disponible, la conexión fallará. Esta configuración no está recomendada.
2. Si en las propiedades del subsistema hay un número de puerto igual a 0, y si se especifica `_RSE_PORTRANGE` en el archivo `rsed.envvars`, el rango de puertos especificado por `_RSE_PORTRANGE` es el que se utilizará. Si no hay ningún puerto disponible en el rango, la conexión fallará.
3. Si en las propiedades del subsistema hay un número de puerto igual a 0, y si `_RSE_PORTRANGE` no está especificado en el archivo `rsed.envvars`, se utilizará cualquier puerto disponible.

**Nota:** Cuando un servidor abre un puerto y está a la escucha, no puede haber otro servidor que utilice ese número de puerto, pero una vez conectado, ese número de puerto se puede usar otra vez. Esto significa que el número de puertos del rango no supone una limitación en cuando al número de usuarios conectados de manera concurrente.

## **(Opcional) Definir parámetros de arranque Java adicionales con `_RSE_*OPTS`**

Con las distintas directivas `_RSE_*OPTS`, el archivo `rsed.envvars` ofrece la posibilidad de proporcionar parámetros adicionales a Java cuando inicia el servidor RSE.

Las opciones de ejemplo incluidas en el archivo `rsed.envvars` se pueden activar a base de quitarles el carácter de comentario.

### **`_RSE_JAVAOPTS`**

`_RSE_JAVAOPTS` define opciones Java estándar y específicas de RSE.

`_RSE_JAVAOPTS=""`

Inicialización de variables. No lo modifique.

`#_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Xquickstart"`

Mejora el tiempo de inicio a base de diferir la compilación JIT y las optimizaciones. Esto lo hace a expensas de unos ejecutables compilados ligeramente menos eficaces, lo que afecta a las tareas de larga ejecución. Quite el carácter de comentario para activarlo.

`#_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Xms128m -Xmx128m"`

Establecer el tamaño inicial (Xms) y máximo (Xmx) de la memoria dinámica. Los valores predeterminados del sistema son 1M y 64M, respectivamente. Quite el carácter de comentario y realice cambios para poner en vigor los valores de tamaño de memoria dinámica especificados.

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Dfile.encoding=Cp424"**

Selección de la página de códigos del host. Quite el carácter de comentario y realice cambios para poner en vigor la página de códigos especificada.

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS**

**-DDENY\_PASSWORD\_SAVE=true"**

Opción de guardado de contraseña. Quite el carácter de comentario si quiere impedir que los usuarios guarden las contraseñas del host en el cliente. Las contraseñas guardadas con anterioridad se eliminarán. Esta opción solo funciona con clientes de la versión 7.1 o superior.

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS**

**-DDSTORE\_TRACING\_ON=true"**

Iniciar el rastreo de dstore. Solo debe utilizarla cuando así se lo indique el centro de soporte de IBM.

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS**

**-DDSTORE\_MEMLOGGING\_ON=true"**

Iniciar el rastreo de memoria de dstore. Solo debe utilizarla cuando así se lo indique el centro de soporte de IBM.

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -DTSO\_SERVER=APPC"**

Utilizar APPC para el servicio de mandatos TSO. Hallará más información en: "(Opcional) Definir una transacción APPC para el servicio de mandatos TSO" en la página 24.

#### **\_RSE\_CLASS\_OPTS**

La directiva **\_RSE\_CLASS\_OPTS** define las opciones de Java 5.0 (o superior) necesarias para compartir clases entre múltiples servidores RSE. Hallará más información en: "Compartir clases entre las JVM" en la página 69.

**\_RSE\_CLASS\_OPTS=""**

Inicialización de variables. No lo modifique.

**#\_RSE\_CLASS\_OPTS=-Xshareclasses:name=RSE,groupAccess,nonFatal**

Solo para Java 5.0 o superior. Habilitar la prestación de compartir clases. Quite el carácter de comentario si desea compartir clases entre múltiples servidores RSE.

**#\_RSE\_CLASS\_OPTS="\$\_RSE\_CLASS\_OPTS -Xscmx6m"**

Solo para Java 5.0 o superior. Establezca el tamaño de la caché de clases compartidas. El valor predeterminado del sistema es 16M.

#### **\_RSE\_CMDSERV\_OPTS**

Las directivas **\_RSE\_CMDSERV\_OPTS** son opciones Java específicas de RSE y solo entran en vigor cuando se utiliza SCLM Developer Toolkit como servidor de mandatos TSO.

**\_RSE\_CMDSERV\_OPTS=""**

Inicialización de variables. No lo modifique.

**#\_RSE\_CMDSERV\_OPTS="\$\_RSE\_CMDSERV\_OPTS&ISPPROF=&SYSUID..ISPPROF"**

Utilice un perfil ISPF existente para el servidor de mandatos TSO. Quite el carácter de comentario y cambie el nombre del conjunto de datos para que utilice el perfil ISPF especificado. Se puede utilizar **&SYSUID.** como sustitución del ID de usuario del desarrollador.

---

## Daemon INETD y configuración de REXEC/SSH de RSE

Developer para System z se basa en el servicio INETD para iniciar el servidor del Explorador de Sistemas Remotos cuando un cliente solicita una conexión. INETD es un daemon estándar de z/OS UNIX que gestiona otros daemons que realizan el trabajo real (en este caso, iniciar el servidor RSE). La configuración de INETD no forma parte de la personalización de Developer para System z, pero podrá hallar información valiosa en: Apéndice D, “Configurar INETD”, en la página 101.

Developer para System z permite utilizar múltiples maneras de iniciar el servidor RSE.

Tendrá que personalizar como mínimo una manera, en función de cómo vayan a operar sus usuarios.

- Daemon RSE, iniciado al establecer conexión con INETD en el puerto 4035 (valor predeterminado). Este es el método recomendado, porque proporciona al usuario más control y seguridad que REXEC.
- El servidor de mandatos REXEC (ejecución remota), cuyo puerto predeterminado es el 512, ejecuta un script de shell que llama a RSE. Este método de conexión puede servir para prueba conceptual, pero no para usarlo a largo plazo. La ventaja de utilizar este método es que requiere menos trabajo de instalación y configuración, si REXEC se ha configurado antes (normalmente durante la configuración de TCP/IP).
- El servidor de mandatos SSH (shell segura), cuyo puerto predeterminado es el 22, ejecuta un script de shell que llama a RSE. Este método es comparable con REXEC, pero SSH emplea comunicaciones seguras (cifradas) con el cliente.

**Nota:** Para las acciones remotas (basadas en host) de los subproyectos z/OS UNIX, es necesario que REXEC o SSH esté activo en el host.

Puede verificar que INETD está activo con el mandato **ps -e** (proporcionado por un usuario autorizado). La salida debe contener una referencia a INETD; por ejemplo (# es el indicador de z/OS UNIX):

```
# ps -e
PID TTY      TIME CMD
  7 ?          0:00 /usr/sbin/inetd
```

**Nota:** Para que los servidores z/OS UNIX (como el daemon RSE, REXEC y SSH) permitan conexiones IPv6, hay que especificar `tcp6` para el protocolo del nombre de servicio en el archivo `/etc/inetd.conf`. Cuando `tcp6` está definido, los clientes IPv4 también están soportados. `/etc/services` solo soporta la palabra clave `tcp`, sin un sufijo numérico.

## Configuración del daemon RSE de INETD

1. Modifique `/etc/services`, añadiendo la línea:

```
rse      4035/tcp      #Developer para System z RSE
```

**rse**      Nombre de servicio del daemon, cuyo valor predeterminado es `rse` (en minúsculas). El nombre debe coincidir con el que se utilizará en `/etc/inetd.conf`.

**4035/tcp**

Puerto y protocolo empleados; el puerto predeterminado es 4035, el protocolo debe ser `tcp`.

El puerto utilizado debe coincidir con el definido en el cliente, que se establece durante la creación de una nueva conexión z/OS.

**Nota:** Antes de seleccionar un puerto, verifique que el puerto está disponible en su sistema, sirviéndose de los mandatos **NETSTAT** y **NETSTAT PORTL**. Hallará más información en: “Puertos TCP/IP reservados” en la página 83.

#### #Developer para System z RSE

Comentario, que debe empezar por un signo de almohadilla (#).

2. Modifique `/etc/inetd.conf` añadiendo estas dos líneas. Las reglas de continuación están en: Apéndice D, “Configurar INETD”, en la página 101.

```
rse stream tcp nowait OMVSKERN /usr/lpp/wd4z/rse/lib/fekfrsed
                                rsed -d /usr/lpp/wd4z/rse/lib -t 60
```

**rse** Nombre de servicio del daemon. El valor predeterminado es `rse` (en minúsculas). El nombre debe coincidir con el que se utiliza en `/etc/services`.

#### **stream tcp nowait**

Sentencias de configuración específicas de INETD (tipo de socket, protocolo, distintivo de espera). No lo modifique.

**Nota:** Utilice la palabra clave `tcp6`, en lugar de `tcp`, para poder utilizar conexiones IPv6.

#### **OMVSKERN**

ID de usuario del proceso del daemon RSE. El valor predeterminado es `OMVSKERN`. Este debe ser un ID de usuario que tenga un segmento de seguridad `OMVS` válido, permiso `BPX.DAEMON` y permiso para leer y ejecutar (`READ` y `EXECUTE`) en los directorios de instalación y configuración de Developer para System z. Para obtener más detalles sobre los requisitos de los ID de usuario que se emplean para los servicios del sistema, consulte: Apéndice D, “Configurar INETD”, en la página 101.

#### **/usr/lpp/wd4z/rse/lib/fekfrsed**

Programa servidor (ubicación absoluta de `fekfrsed`). El valor predeterminado es `/usr/lpp/wd4z/rse/lib/fekfrsed`

Todo lo que hay después del argumento INETD son argumentos de servidor, empezando por el nombre del servidor.

**rsed** Nombre del servidor. No lo modifique.

#### **-d /usr/lpp/wd4z/rse/lib**

Directorio de trabajo (ubicación de los archivos de configuración de servidor RSE). El valor predeterminado es `/usr/lpp/wd4z/rse/lib`.

**Nota:** Le recomendamos que copie los archivos de configuración de RSE personalizados en un nuevo directorio (como `/etc/wd4z/`) para evitar que se sobrescriban al aplicar el mantenimiento. El directorio de trabajo que se defina aquí debe reflejar este cambio. Por ejemplo:

```
rse stream tcp nowait OMVSKERN /usr/lpp/wd4z/rse/lib/fekfrsed rsed -d /etc/wd4z
```

Las definiciones que figuran a continuación son opcionales. Si se omiten, se emplearán los valores predeterminados.

**-t 60** Opción de tiempo de espera para especificar cuántos segundos espera el daemon RSE a que responda el servidor RSE. El valor

predeterminado es 60 segundos. El tiempo de espera del servidor RSE que está a la espera en el cliente se establece en `rsed.envvars` y es de 2 minutos por defecto.

3. Un usuario autorizado debe reiniciar INETD para que se activen los cambios realizados en los archivos `/etc`, como se describe en: Apéndice D, “Configurar INETD”, en la página 101. Vea los siguientes mandatos de ejemplo (# es el indicador de z/OS UNIX):

```
a. # ps -e | grep inetd
   50331687 ?          0:00 /usr/sbin/inetd
b. # kill 50331687
c. # _BPX_JOBNAME='INETD' /usr/sbin/inetd
d. # netstat | grep 4035
   INETD4      00000B6A 0.0.0.0..4035          0.0.0.0..0          Escucha
```

**Nota:** Si el perfil `BPX.DAEMON` está definido en la clase `FACILITY` de su producto de seguridad, y resulta que el usuario que (re)inicia INETD no tiene acceso a este recurso, cabe esperar el siguiente aviso de seguridad para cada cliente que se conecte a RSE, siendo `IBMUSER` el ID de usuario empleado para iniciar INETD.

```
ICH408I USER(IBMUSER ) GROUP(SYS1 ) NAME(IBMUSER )
        BPX.DAEMON CL(FACILITY)
        INSUFFICIENT ACCESS AUTHORITY
        ACCESS INTENT(READ ) ACCESS ALLOWED(NONE )
```

## Configuración de REXEC (o SSH) de INETD

No hay ninguna configuración específica de Developer para System z para utilizar el servidor de mandatos REXEC (o SSH) de INETD. Sin embargo, el cliente debe conocer 2 valores para poder iniciar una conexión RSE a través de REXEC/SSH:

- El directorio en el que se encuentra el script de arranque `server.zseries`.

Por defecto, es el directorio de instalación (`/usr/lpp/wd4z/rse/lib/`). Sin embargo, `server.zseries` es uno de los archivos que también se deben copiar si `rsed.envvars` se copia en un directorio distinto, como `/etc/wd4z/`.

- El puerto que se está utilizando.

Un puerto común utilizado por REXEC es el 512. Una manera rápida de comprobarlo es con el mandato **NETSTAT**, como se ve en el siguiente ejemplo (\$ es el indicador de z/OS UNIX):

```
$ netstat | grep 512
INETD4      0000002E 0.0.0.0..512          0.0.0.0..0          Escucha
```

Para verificarlo, puede comprobar `/etc/inetd.conf` y `/etc/services` para localizar el número de puerto que se emplea.

1. Localice el nombre de servicio (1ª palabra, `exec` en este ejemplo) del servidor `rexecd` (7ª palabra) en `/etc/inetd.conf`

```
exec      stream tcp nowait OMVSKERN /usr/sbin/orexecd rexecd -LV
```

2. Localice el puerto (2ª palabra, 512 en este ejemplo) conectado a este nombre de servicio (1ª palabra) en `/etc/services`

```
exec      512/tcp          #REXEC Command Server
```

Este mismo principio es válido para SSH. El puerto común es el 22, y el nombre de servicio es `sshd`.

**Nota:** Para las acciones remotas (basadas en host) de los subproyectos z/OS UNIX, es necesario que REXEC o SSH esté activo en el host. Si REXEC/SSH no está configurado para utilizar el puerto predeterminado, el cliente Developer

para System z debe definir el puerto correcto que deben utilizar los subproyectos z/OS UNIX. Para ello, se selecciona la página de preferencias **Ventana > Preferencias... > z/OS Solutions > Subproyectos USS > Opciones de acción remota.**

---

## Personalizar el archivo de configuración ISPF, ISPF.conf

Este paso solo es necesario cuando se utiliza SCLM Developer Toolkit para el servicio de mandatos TSO (este es el valor predeterminado). No se necesita al utilizar APPC para el servicio de mandatos TSO.

Para SCLM Developer Toolkit se necesitan las definiciones de ISPF.conf para crear un entorno válido que ejecute los servicios ISPF. El servicio de mandatos TSO se tiene que añadir a este entorno ISPF.

ISPF.conf se crea durante la personalización de SCLM Developer Toolkit, descrita en el manual *SCLM Developer Toolkit Installation and Customization Guide* (SC23-8504). La ubicación predeterminada es /etc/SCLMDT/CONFIG, pero quizá no sea válida para su local.

Añada las siguientes líneas a ISPF.conf, siendo hlq el calificador de alto nivel empleado para instalar Developer para System z (el valor predeterminado es FEK).

```
*****
* Developer para System z – Servidor de mandatos TSO
*****
sysexec=hlq.SFEKPROC
```

El resultado debe tener un aspecto igual al del ejemplo de la Figura 5.

```
sysproc=ISP.SISPCLIB
isplib=ISP.SISPMENU
isptlib=ISP.SISPTENU
ispplib=ISP.SISPPENU
ispslib=ISP.SISPSLIB
ispllib=BWB.SWBLOAD
*****
* Developer para System z – Servidor de mandatos TSO
*****
sysexec=FEK.SFEKPROC
```

*Figura 5. ISPF.conf - archivo de configuración de ISPF*

**Nota:** Si la sentencia sysexec ya está definida, añada el conjunto de datos hlq.SFEKPROC al final de ella, separando los nombres de los conjuntos de datos con una coma (,).

---

## Verificar la instalación del servidor RSE

La instalación de Developer para System z proporciona varios programas de verificación de instalación (IVP) para el servidor RSE. Los scripts de los IVP se encuentran en el directorio de instalación, que es /usr/lpp/wd4z/rse/lib/ por defecto.

- fekfivpa : “Conexión del servicio de mandatos TSO (utilizando APPC)” en la página 52
- fekfivpc : “Conexión del servicio de mandatos TSO (utilizando SCLMDT)” en la página 50
- fekfivpd : “Conexión del daemon RSE” en la página 50

- fekfivpj : “Conexión del supervisor de trabajos JES” en la página 50
- fekfivpr : “Conexión REXEC” en la página 48
- fekfivps : “Script de shell REXEC/SSH” en la página 49

Para todos los mandatos de ejemplo de este apartado se espera que las variables de entorno de RSE estén establecidas. De esta manera, los scripts del IVP están disponibles mediante la sentencia PATH, y se conoce la ubicación de rsed.envvars. Utilice los mandatos **pwd** y **cd** para verificar y cambiar el directorio actual para pasar al directorio que tiene el archivo rsed.envvars personalizado. Luego se puede usar el script de shell setup.env.zseries para establecer las variables de entorno de RSE, como en el ejemplo que sigue (\$ es el indicador de z/OS UNIX):

```
$ pwd
/etc
$ cd /etc/wd4z
$ . ./setup.env.zseries
```

El script de shell . ./setup.env.zseries, que reside en el mismo directorio que rsed.envvars, exporta las variables de entorno para que otros procesos puedan utilizarlas. El primer punto "." de . ./setup.env.zseries es un mandato z/OS UNIX que ejecuta la shell en el entorno actual, para que las variables de entorno establecidas en la shell estén en vigor incluso después de salir de la shell. El segundo punto hace referencia al directorio actual.

**Nota:** Si no se ejecuta . ./setup.env.zseries antes de los scripts fekfivp\*, la vía de acceso a estos scripts se tiene que especificar al llamarlos, como en este ejemplo:

```
/usr/lpp/wd4z/rse/lib/fekfivpr 512 USERID
```

Asimismo, la mayoría de los scripts fekfivp\* pedirán la ubicación del archivo rsed.envvars personalizado si no se ejecuta . ./setup.env.zseries en primer lugar.

**Nota:** Algunas pruebas de IVP emplean la API de socket REXX de TCP/IP, que exige que la biblioteca de carga de TCP/IP, cuyo valor predeterminado es TCPIP.SEZALOAD, esté en LINKLIST o en STEPLIB. Para poder ejecutar estas pruebas de IVP, podrían ser necesarios los siguientes mandatos (\$ es el indicador de z/OS UNIX):

```
$ echo $STEPLIB
none
$ STEPLIB=TCPIP.SEZALOAD
```

o bien

```
$ echo $STEPLIB
SOME.STEPLIB.DATASET
$ STEPLIB=$STEPLIB:TCPIP.SEZALOAD
```

Para obtener información sobre cómo diagnosticar los problemas de conexión del RSE, vea: Apéndice B, “Resolución de problemas de configuración”, en la página 77, o consulte las fichas técnicas de la página de soporte de Developer para System z, <http://www-306.ibm.com/software/awdtools/devzseries/support/>.

## Disponibilidad de los puertos

La disponibilidad de los puertos del supervisor de trabajos JES, de REXEC y SSH y del daemon RSE se puede verificar emitiendo el mandato **netstat**. El resultado debe mostrar los puertos empleados por estos servicios, como en los siguientes ejemplos (\$ es el indicador de z/OS UNIX):



## IPv4

```
$ netstat
MVS TCP/IP NETSTAT CS V1R7      Nombre de TCPIP: TCPIP      13:57:36
ID us.   Conexión Socket Local      Socket Foráneo  Estado
-----
INETD4   00000014 0.0.0.0..22      0.0.0.0..0      Escucha
INETD4   00000030 0.0.0.0..512     0.0.0.0..0      Escucha
INETD4   0000004B 0.0.0.0..4035    0.0.0.0..0      Escucha
JMON     00000037 0.0.0.0..6715    0.0.0.0..0      Escucha
```

## IPv6

```
$ netstat
MVS TCP/IP NETSTAT CS V1R7      Nombre de TCPIP: TCPIP      14:03:35
ID usuario Conexión Estado
-----
INETD4     00000018 Escucha
Socket local: 0.0.0.0..22
Socket foráneo: 0.0.0.0..0
INETD4     00000046 Escucha
Socket local: 0.0.0.0..512
Socket foráneo: 0.0.0.0..0
INETD4     0000004B Escucha
Socket local: 0.0.0.0..4035
Socket foráneo: 0.0.0.0..0
JMON       00000037 Escucha
Socket local: 0.0.0.0..6715
Socket foráneo: 0.0.0.0..0
```

## Conexión REXEC

Verifique la conexión REXEC ejecutando el siguiente mandato. Donde pone 512, escriba el puerto que utiliza REXEC, y donde pone USERID, escriba un ID de usuario válido.

```
fekfivpr 512 USERID
```

Después de solicitarle una contraseña, el mandato debe devolver el rastreo REXEC, un aviso de tiempo de espera, la versión de Java y el mensaje del servidor RSE, como en el siguiente ejemplo (\$ es el indicador de z/OS UNIX):

```
$ fekfivpr 512 USERID
Teclee la contraseña:
$ EZYRC01I Llamando a la función rexec_af con:
EZYRC02I Host: CDFMVS08, user USERID, cmd cd /etc/wd4z;export RSE_USER_ID=USERID;./server.zseries -ivp, port 512
EZYRC19I Socket de datos = 4, Socket de control = 6.
```

se esperan mensajes de tiempo de espera tras una prueba de IVP satisfactoria

```
java version "1.5.0"
Java(TM) 2 Runtime Environment, Standard Edition (build pmz31dev-20070201 (SR4))
IBM J9 VM (build 2.3, J2RE 1.5.0 IBM J9 2.3 z/OS s390-31 j9vmmz3123-20070201 (JIT enabled))
J9VM - 20070131_11312_bHdSMr
JIT - 20070109_1805ifx1_r8
GC - 200701_09)
JCL - 20070126
```

```
El servidor se ha iniciado satisfactoriamente
1272
El servidor se ejecuta en: CDFMVS08
```



**Nota:** Si no obtiene datos de salida de Java ni del servidor RSE, es probable que el tamaño de la región INETD sea demasiado pequeño (debe ser igual o mayor que 2096128 si se inicia desde una sesión de shell TSO/OMVS, o debe ser 0 en el caso de BPXBATCH).

**Nota:** Puede probar el script de shell utilizado por REXEC por separado, como se describe en la siguiente prueba de IVP, "Script de shell REXEC/SSH".

**Nota:** El servidor se inicia sin un cliente que intente conectarse, por lo que se agotará el tiempo de espera (al cabo de 5 segundos). Así se obtendrá un rastreo de pila Java (líneas 25+) que se parece al del siguiente ejemplo:

```
$ java.net.SocketTimeoutException: Accept timed out
    at java.net.PlainSocketImpl.socketAccept(Native Method)
    at java.net.PlainSocketImpl.accept(PlainSocketImpl.java:384)
    at java.net.ServerSocket.implAccept(ServerSocket.java:471)
    at java.net.ServerSocket.accept(ServerSocket.java:442)
    at com.ibm.etools.systems.dstore.core.server.ConnectionEstablisher.
...

```

## Script de shell REXEC/SSH

Es posible saltarse esta prueba de IVP si la prueba anterior (descrita someramente en: "Conexión REXEC" en la página 48) se llevó a cabo satisfactoriamente.

Verifique el script de shell utilizado por la conexión REXEC y SSH, ejecutando el mandato:

```
fekfivps
```

El mandato debe devolver un aviso de tiempo de espera, la versión de Java y el mensaje del servidor RSE, como en el ejemplo que sigue (\$ es el indicador de z/OS UNIX):

```
$ fekfivps
$ java version "1.5.0"
```

se esperan mensajes de tiempo de espera tras una prueba de IVP satisfactoria

```
Java(TM) 2 Runtime Environment, Standard Edition (build pmz31dev-20070201 (SR4))
IBM J9 VM (build 2.3, J2RE 1.5.0 IBM J9 2.3 z/OS s390-31 j9vmmz3123-20070201 (JIT
enabled)
J9VM - 20070131_11312_bHdSMr
JIT - 20070109_1805ifx1_r8
GC - 200701_09)
JCL - 20070126
```

```
El servidor se ha iniciado satisfactoriamente
1751
El servidor se ejecuta en: CDFMVS08$
```

**Nota:** Si no obtiene datos de salida, es probable que el tamaño de la región (TSO) sea demasiado pequeño (debe ser igual o mayor que 2096128).

**Nota:** El servidor se inicia sin un cliente que intente conectarse, por lo que se agotará el tiempo de espera (al cabo de 5 segundos). Así se obtendrá un rastreo de pila Java (líneas 25+) que se parece al del siguiente ejemplo:

```
$ java.net.SocketTimeoutException: Accept timed out
    at java.net.PlainSocketImpl.socketAccept(Native Method)
    at java.net.PlainSocketImpl.accept(PlainSocketImpl.java:384)

```

```

at java.net.ServerSocket.implAccept(ServerSocket.java:471)
at java.net.ServerSocket.accept(ServerSocket.java:442)
at com.ibm.etools.systems.dstore.core.server.ConnectionEstablisher.
...

```

## Conexión del daemon RSE

Verifique la conexión del daemon RSE ejecutando el siguiente mandato. Donde pone 4035, escriba el puerto que utiliza el daemon RSE, y donde pone USERID, escriba un ID de usuario válido.

```
fekfivpd 4035 USERID
```

Después de solicitarle una contraseña, el mandato debe devolver datos de salida parecidos a los de este ejemplo (\$ es el indicador de z/OS UNIX):

```

$ fekfivpd 4035 USERID
Contraseña:
SSL está inhabilitado
conectado
8108
570655399
Éxito

```

**Nota:** Al probar una conexión habilitada por SSL, verifique que ha especificado el puerto correcto si obtiene un mensaje de error como este:  
gsk\_secure\_socket\_init() ha fallado: Socket cerrado por interlocutor remoto

## Conexión del supervisor de trabajos JES

Verifique la conexión del supervisor de trabajos JES ejecutando el siguiente mandato. Donde pone 6715, escriba el número de puerto del supervisor de trabajos JES.

```
fekfivpj 6715
```

El mandato debe devolver el mensaje de acuse de recibo del supervisor de trabajos JES, como en el siguiente ejemplo (\$ es el indicador de z/OS UNIX):

```

$ fekfivpj 6715
En espera de una respuesta del supervisor de trabajos JES...
ACKNOWLEDGE01v03

Éxito

```

## Conexión del servicio de mandatos TSO (utilizando SCLMDT)

Esta prueba de IVP solo se necesita si utiliza SCLM Developer Toolkit, ya sea para el servicio de mandatos TSO o para el plug-in de cliente.

Verifique la conexión establecida con el servidor de mandatos TSO utilizando SCLM Developer Toolkit, emitiendo el mandato:

```
fekfivpc
```

El mandato debe devolver el resultado de las comprobaciones relacionadas con SCLM Developer Toolkit (variables, módulos HFS, entorno de ejecución REXX, iniciar y detener la sesión TSO/ISPF), como en el siguiente ejemplo (\$ es el indicador de z/OS UNIX):

```

$ fekfivpc
-----
Verificación de la instalación de host para RSE
Revise los mensajes de anotaciones de IVP procedentes del HOST, a continuación:
-----

```

Solo comprobación de inicialización de sesión TSO/ISPF base y conexión RSE

\*\*\* COMPROBACIÓN : VARIABLES DE ENTORNO - variables clave visualizadas más abajo:

Server PATH = /usr/lpp/java/J5.0/bin:/usr/lpp/wd4z/rse/lib:/usr/lpp/SCLMDT/bin:/usr/sbin:.

STEPLIB = BWB.SWBLOAD

\_CMDSERV\_BASE\_HOME = /usr/lpp/SCLMDT

\_CMDSERV\_CONF\_HOME = /etc/SCLMDT

\_CMDSERV\_WORK\_HOME = /var/SCLMDT

-----  
\*\*\* COMPROBACIÓN : MÓDULOS HFS

Comprobando directorio de instalación : /usr/lpp/SCLMDT

Comprobando módulos BWB en directorio /bin

RC=0

MSG: SATISFACTORIO

-----  
\*\*\* COMPROBACIÓN : ENTORNO DE TIEMPO DE EJECUCIÓN REXX

RC=0

MSG: SATISFACTORIO

-----  
\*\*\* COMPROBACIÓN : INICIALIZACIÓN DE TSO/ISPF

( La sesión TSO/ISPF se inicializará )

RC=0

MSG: SATISFACTORIO

-----  
\*\*\* COMPROBACIÓN: Cerrando sesión IVP de TSO/ISPF

RC=0

MSG: SATISFACTORIO

-----  
La verificación de la instalación de host se ha realizado satisfactoriamente  
-----

**Nota:** Si falla alguna de las comprobaciones SCLMDT, se mostrará información más detallada.

fekfivpc tiene varios parámetros opcionales que no dependen de la posición:

**-file** fekfivpc puede producir grandes cantidades de datos de salida (cientos de líneas). El parámetro -file envía estos datos de salida a un archivo, home/.eclipse/RSE/USERID/fekfivpc.log, donde home es la vía de acceso inicial definida en su segmento OMVS (o en el segmento OMVS predeterminado, si no tiene un segmento OMVS) y USERID es su ID de usuario (en mayúsculas).

**-plugin**

Por defecto, fekfivpc solo comprueba las funciones necesarias para el servicio de mandatos TSO. El parámetro -plugin añade pruebas adicionales para el plug-in de cliente SCLMDT.

**-debug**

El parámetro -debug crea una salida detallada de la prueba. No debe utilizar esta opción, a menos que así se lo indique el centro de soporte de IBM.

## Conexión del servicio de mandatos TSO (utilizando APPC)

No ejecute este procedimiento si no ha instalado APPC para el servicio de mandatos TSO.

Verifique la conexión establecida con el servidor de mandatos TSO (utilizando APPC), emitiendo el siguiente mandato. Donde pone USERID, escriba un ID de usuario válido.

fekfivpa USERID

Después de solicitarle una contraseña, el mandato debe devolver la conversación APPC, como en el siguiente ejemplo (\$ es el indicador de z/OS UNIX):

```
$ fekfivpa USERID
Teclee la contraseña:
20070607 13:57:18.584060 /usr/lpp/wd4z/rse/lib/fekfscmd: version=Oct 2003
20070607 13:57:18.584326 Input parms: 1.2.3.4 * NOTRACE USERID *****
20070607 13:57:18.585132 TP_name set via envvar: FEKFRSRV
20070607 13:57:18.586800 APPC: Allocate succeeded
20070607 13:57:18.587022 Conversation id is 0DDBD3F800000000
20070607 13:57:18.587380 APPC: Set Send Type succeeded
20070607 13:57:26.736674 APPC: Confirm succeeded
20070607 13:57:26.737027 Req to send recd value is 0
20070607 13:57:26.737546 APPC: SEND_DATA return_code = 0
20070607 13:57:26.737726 request_to_send_received = 0
20070607 13:57:26.737893 Send Data succeeded
20070607 13:57:26.738169 APPC: Set Prepare to Receive type succeeded
20070607 13:57:26.738580 APPC: Prepare to Receive succeeded
20070607 13:57:26.808899 APPC: Receive data
20070607 13:57:26.809122 RCV return_code = 0
20070607 13:57:26.809270 RCV data_received= 2
20070607 13:57:26.809415 RCV received_length= 29
20070607 13:57:26.809556 RCV status_received= 4
20070607 13:57:26.809712 RCV req_to_send= 0
20070607 13:57:26.809868 Receive succeeded
:IP: 0 9.42.112.75 1674 50246
20070607 13:57:26.810533 APPC: CONFIRMED succeeded
```

Para obtener información sobre cómo diagnosticar los problemas de conexión del RSE, vea: Apéndice B, “Resolución de problemas de configuración”, en la página 77, o consulte las fichas técnicas de la página de soporte de Developer para System z, <http://www-306.ibm.com/software/awdtools/devzseries/support/>.

---

## (Opcional) Personalizar la configuración SSL de RSE, ssl.properties

Todos los métodos de conexión de cliente de Developer para System z utilizan las variables de capa de sockets segura (SSL) establecidas en el archivo `ssl.properties`, situado por defecto en el directorio de instalación, `/usr/lpp/wd4z/rse/lib/`. Sin embargo, `ssl.properties` es uno de los archivos que también se deben copiar si `rsed.envvars` se copia en un directorio distinto, como `/etc/wd4z/`. El archivo de ejemplo proporcionado tiene las sentencias enumeradas en: Figura 6 en la página 53, donde las líneas de comentarios empiezan por un signo de almohadilla (#).

```
# Especifique esta propiedad como true para habilitar SSL
enable_ssl=false

#####
# Propiedades del daemon
# Hay que especificar el archivo de base de datos de claves y la contraseña para
# el daemon.
# Hay que especificar la etiqueta de clave si no es la clave predeterminada.
#daemon_keydb_file=
#daemon_keydb_password=
#daemon_key_label=

#####
# Propiedades del servidor
# Hay que especificar el archivo de almacén de claves y la contraseña para
# el servidor.
#server_keystore_file=
#server_keystore_password=
```

*Figura 6. ssl.properties – archivo de configuración SSL*

Las propiedades del daemon y del servidor solo se deben establecer si se habilita SSL. Para obtener más información sobre la configuración de SSL, consulte: Apéndice E, “Configurar SSL”, en la página 109.

---

## (Opcional) Personalizar la configuración del rastreo RSE, rsecomm.properties

Todos los métodos de conexión de cliente de Developer para System z utilizan las variables establecidas en el archivo `rsecomm.properties`, situado por defecto en el directorio de instalación, `/usr/lpp/wd4z/rse/lib/`. Sin embargo, `rsecomm.properties` es uno de los archivos que también se deben copiar si `rsecl.envvars` se copia en un directorio distinto, como `/etc/wd4z/`. El archivo de ejemplo proporcionado tiene las sentencias enumeradas en: Figura 7, donde las líneas de comentarios empiezan por un signo de almohadilla (#).

```
# server.version - ¡NO LO MODIFIQUE!
server.version=5.0.0

# Nivel de anotaciones
# 0 - Mensajes de error de anotaciones
# 1 - Mensajes de aviso y de error de anotaciones
# 2 - Mensajes informativos, de aviso y de error de anotaciones
# 3 - Mensajes de depuración, informativos, de aviso y de error de anotaciones
debug_level=1

# Ubicación de las anotaciones
# Log_To_StdOut
# Log_To_File
log_location=Log_To_File
```

*Figura 7. rsecomm.properties – archivo de configuración de anotaciones*

Al seleccionar `log_location=Log_To_File` (que es el valor predeterminado), las anotaciones se escriben en `home/.eclipse/RSE/USERID/rsecomm.log`, donde `home` es la vía de acceso inicial definida en el segmento OMVS del usuario (o en el segmento OMVS predeterminado si el usuario no tiene un segmento OMVS) y `USERID` es el ID de usuario de inicio de sesión (en mayúsculas).

**Nota:** La definición `debug_level` también controla el nivel de anotaciones de los otros archivos de anotaciones que se encuentren en este directorio.

**Atención:** El hecho de cambiar estos valores puede afectar negativamente al rendimiento, y solo se deben cambiar cuando así lo indique el centro de soporte de IBM.

---

## (Opcional) Personalizar la configuración de proyectos de host, `projectcfg.properties`

Los proyectos de z/OS se pueden definir individualmente mediante la perspectiva Proyectos z/OS en el cliente, pero también se pueden definir centralmente en el host y propagarse al cliente para cada usuario. Estos "proyectos basados en host" se parecen y funcionan exactamente igual que los proyectos definidos en el cliente, salvo que el cliente no puede modificar su estructura, sus miembros ni sus propiedades, y solo se puede acceder a ellos cuando se está conectado al host.

La ubicación de las definiciones de proyecto se define en el archivo `projectcfg.properties`, situado por defecto en el directorio de instalación, `/usr/lpp/wd4z/rse/lib/`. Sin embargo, `projectcfg.properties` es uno de los archivos que también se deben copiar si el archivo `rsed.envvars` se copia en un directorio distinto, como `/etc/wd4z/`.

El archivo de ejemplo proporcionado tiene las sentencias enumeradas en: Figura 8, donde las líneas de comentarios empiezan por un signo de almohadilla (#).

```
#
# proyectos basados en host – archivo de configuración raíz
#
# WSED-VERSION – ino la modifique!
WSED-VERSION=7.0.0.0
# Especifique la ubicación de los archivos de definición de los proyectos basados en host
PROJECT-HOME=/var/wd4z/projects
```

*Figura 8. `projectcfg.properties` – archivo de configuración de proyectos basados en host*

La única variable que hay que cambiar es `PROJECT-HOME`. Su valor, que por defecto es `/var/wd4z/projects`, es el directorio base de las definiciones de proyecto.

**Nota:** Para activar los proyectos basados en host, debe existir un archivo `project.instance` en `/var/wd4z/projects/USERID`, donde `/var/wd4z/projects` es la ubicación de los archivos de definición de los proyectos y `USERID` es el ID de usuario con el que inicia sesión el desarrollador.

Para obtener más información sobre los proyectos basados en host, lea el libro blanco *Host Based Projects in WebSphere Developer para System z version 7.0*, en la biblioteca de Developer para System z en Internet, <http://www-306.ibm.com/software/awdtools/devzseries/library/>.

---

## (Opcional) Personalizar la integración del gestor de archivos, `FMIEXT.properties`

Developer para System z admite el acceso directo desde el cliente a un conjunto limitado de funciones de IBM File Manager para z/OS. IBM File Manager para z/OS proporciona herramientas completas para trabajar con conjuntos de datos MVS, archivos z/OS UNIX y datos DB2, IMS y CICS. Estas herramientas incluyen las conocidas utilidades para examinar, editar, copiar e imprimir existentes en ISPF

y mejoradas para responder a las necesidades de los desarrolladores de aplicaciones. En la versión actual de Developer para System z, solo está permitido examinar/editar conjuntos de datos MVS (incluidos los KSDS y ESDS de VSAM).

Tenga en cuenta que el producto IBM File Manager para z/OS se debe pedir, instalar y configurar por separado. Consulte el manual *Rational Developer para System z Guía de planificación de host*, GI11-7839-00 (GI11-8296-00) para saber qué nivel del gestor de archivos se necesita para su versión de Developer para System z. La instalación y la personalización de este producto no se describe en este manual.

Las definiciones del gestor de archivos que se necesitan para Developer para System z están almacenadas en el archivo `FMIEXT.properties`, situado por defecto en el directorio de instalación, `/usr/lpp/wd4z/rse/lib/`. Sin embargo, `FMIEXT.properties` es uno de los archivos que también se deben copiar si `rse.envvars` se copia en un directorio distinto, como `/etc/wd4z/`.

El archivo de ejemplo proporcionado tiene las sentencias enumeradas en: Figura 9, donde las líneas de comentarios empiezan por un signo de almohadilla (#).

```
# Propiedades de la extensión Integración del gestor de archivos (FMI)
#
startup.script=/usr/lpp/wd4z/rse/lib/fmiSub
startup.port=1957
startup.range=100
startup.fmload=FMN.SFMNMOD1
startup.jobcard1=//JOB CARD JOB <parámetros del trabajo>
startup.jobcard2=//*
startup.jobcard3=//*
startup.sysout=*
```

Figura 9. `FMIEXT.properties` – archivo de configuración del gestor de archivos

#### **startup.script**

Ubicación absoluta de `fmiSub`, que es el script de arranque del servidor FMI. El valor predeterminado es `/usr/lpp/wd4z/rse/lib/fmiSub`.

#### **startup.port**

Primer puerto utilizado para la comunicación entre el servidor FMI y el servidor RSE, que retransmite la información al cliente. El puerto predeterminado es 1957. La comunicación en este puerto está confinada a la máquina de hospedaje.

**Nota:** Antes de seleccionar un puerto, verifique que el puerto está disponible en su sistema, sirviéndose de los mandatos **NETSTAT** y **NETSTAT PORTL**. Hallará más información en: “Puertos TCP/IP reservados” en la página 83.

#### **startup.range**

Rango de puertos, empezando por `startup.port`, que se utilizará para la comunicación del servidor FMI. El valor predeterminado es 100. Por ejemplo, cuando se utilizan los valores predeterminados, el servidor FMI puede emplear los puertos comprendidos entre el 1957 y el 2056 (inclusive).

#### **startup.fmload**

Ubicación absoluta de la biblioteca de carga del gestor de archivos. El valor

predeterminado es FMN.SFMNMOD1. No utilice comillas simples (') para hacer que el nombre del conjunto de datos sea absoluto, no se añade un prefijo.

**Nota:** El gestor de archivos (FM) tiene múltiples bibliotecas de carga. Aquella a la que hay que hacer referencia en este archivo de configuración es SFMNMOD1.

**startup.jobcard1**

**startup.jobcard2**

**startup.jobcard3**

Información de tarjetas de trabajo para el servidor FMI. Los valores predeterminados son //JOB CARD JOB <parámetros del trabajo>, //\* y //. El nombre del trabajo se sustituirá por FEK<puerto> para garantizar su exclusividad.

**startup.sysout**

Clase sysout para el servidor FMI. El valor predeterminado es \*.



---

## Capítulo 5. (Opcional) Activar el gestor de repositorios de acceso común (CARMA) de IBM

El gestor de repositorios de acceso común (CARMA, FMID: HCMA710) es una ayuda de productividad para los desarrolladores que se proponen crear interfaces API para los gestores de configuración de software (SCM). A su vez, las aplicaciones (por ejemplo, Developer para System z) pueden utilizar estas API para acceder a los SCM.

Antes de instalar la versión 7.1.1, si es usted un usuario anterior de CARMA, le recomendamos que guarde la personalización relacionada, como se describe en: "Hacer copia de seguridad de archivos configurados con anterioridad" en la página 11.

Después de la instalación, debe configurar CARMA siguiendo estos pasos:

1. Configure el servidor CARMA en su host z/OS (hay que llevar a cabo acciones en MVS y z/OS UNIX).
2. (Opcional) Configure los RAM de ejemplo.
3. (Opcional) Restrinja el acceso a los archivos de inicialización y a los clústeres VSAM. En la mayoría de las circunstancias, solo los administradores del sistema y los desarrolladores de RAM de CARMA tendrán que escribir en estos archivos, mientras que los otros usuarios solo necesitarán acceso de lectura.

**Nota:** Los gestores de acceso a repositorios (RAM) son API escritas por el usuario para intercambiar información con los gestores de configuraciones de software (SCM) de z/OS.

En la Tabla 2 en la página 1 encontrará una lista de los manuales que proporcionan más información sobre CARMA y su utilización.

El usuario puede controlar la cantidad de información de rastreo generada por CARMA, estableciendo el Nivel de rastreo en la pestaña Propiedades de la conexión CARMA en el cliente. Las opciones de nivel de rastreo son:

- Inhabilitar anotaciones
- Anotaciones de error
- Anotaciones de aviso
- Anotaciones informativas
- Anotaciones de depuración

El valor predeterminado es  
Anotaciones de error

Para obtener más información sobre la ubicación de los archivos de anotaciones, consulte: "Ubicación de los archivos de anotaciones" en la página 77.

---

### Personalizar los componentes MVS de CARMA

En este apartado, todas las referencias a hlq se refieren al calificador de alto nivel empleado durante la instalación de CARMA. El valor predeterminado de la instalación es CRA, pero quizá no sea válido para su local.

**Nota:** En la versión 7.1, se han añadido mensajes nuevos al VSAM de mensajes de CARMA, CRAMSG. Conviene que actualice el VSAM de mensajes anterior. Además, ha habido un cambio de nombre para el RAM de esqueleto de ejemplo en la versión 7.1, con el consiguiente cambio en el VSAM de configuración de CARMA, CRADEF. La actualización de este VSAM solo es necesaria si piensa utilizar el RAM de esqueleto.

Para configurar el host MVS, siga estos pasos:

1. Copie los miembros que hay que personalizar del directorio de instalación en una biblioteca personal y personalice esas copias, para evitar que se sobrescriban al aplicar el mantenimiento.
  - hlq.SCRACLST(CRASUBMT)
  - hlq.SCRASAMP(CRA\$VDEF)
  - hlq.SCRASAMP(CRA\$VMSG)
  - hlq.SCRASAMP(CRA\$VSTR)
2. Personalice la CLIST de hlq.SCRACLST(CRASUBMT). Las instrucciones de personalización están en la documentación de CRASUBMT. La CLIST de CRASUBMT somete un servidor CARMA.

**Nota:** Si lo desea, puede cambiar el valor de tiempo de espera de CARMA, modificando la línea PROC 1 PORT TIMEOUT(420) en la CLIST de hlq.SCRACLST(CRASUBMT). El valor de tiempo de espera es el número de segundos que CARMA esperará a que llegue el próximo mandato del cliente. Si se establece que el valor es 0, se obtiene el valor de tiempo de espera predeterminado, que actualmente es de 420 segundos (7 minutos).

3. Personalice y someta el JCL de hlq.SCRASAMP(CRA\$VDEF). Las instrucciones de personalización están en la documentación de CRA\$VDEF.

**Nota:** Puede utilizar el JCL de CRA\$VDEF para actualizar el clúster VSAM de CRADEF (configuración de CARMA) en un momento posterior. Para actualizar el clúster, debe hacer que la sentencia DD INPUT señale hacia el conjunto de datos secuencial (SDS) elegido, en lugar de señalar hacia CRAINIT. Para obtener más información sobre cómo definir este conjunto de datos secuencial (SDS), consulte la publicación *Rational Developer para System z Common Access Repository Manager Developer's Guide* (SC23-7660).

4. Personalice y someta el JCL de hlq.SCRASAMP(CRA\$VMSG). Las instrucciones de personalización están en la documentación de CRA\$VMSG. CRA\$VMSG crea y pone a punto el VSAM de mensajes de CARMA, CRAMSG.
5. Personalice y someta el JCL de hlq.SCRASAMP(CRA\$VSTR). Las instrucciones de personalización están en la documentación de CRA\$VSTR.

**Nota:** Puede utilizar el JCL de CRA\$VSTR para actualizar el clúster VSAM de CRASTRS (información personalizada de CARMA) en un momento posterior. Para actualizar el clúster, debe hacer que la sentencia DD INPUT señale hacia el conjunto de datos secuencial (SDS) elegido, en lugar de señalar hacia CRASINIT. Para obtener más información sobre cómo definir este conjunto de datos secuencial (SDS), consulte la publicación *Rational Developer para System z Common Access Repository Manager Developer's Guide* (SC23-7660).

---

## Personalizar los componentes z/OS UNIX de CARMA

Si no está familiarizado con z/OS UNIX, conviene que pida ayuda a un administrador de z/OS UNIX con experiencia o a otro administrador de UNIX para realizar las tareas que figuran en este apartado.

Los mandatos de z/OS UNIX necesarios para realizar las tareas enumeradas se describen de manera resumida para su comodidad. A menos que se le indique lo contrario, consulte el manual *UNIX System Services Command Reference (SA22-7802)* para obtener más información sobre estos mandatos.

- En las tareas que se describen a continuación, se espera que esté activo en z/OS UNIX. Para ello, emita el mandato TSO **OMVS**. Utilice el mandato **exit** para volver a TSO.
- MVS proporciona la posibilidad de editar archivos de z/OS UNIX utilizando ISPF mediante el mandato **OEDIT**. Este mandato se puede usar en TSO y también en OMVS.

En esta sección, todas las sentencias que incluyen la vía de acceso `/usr/lpp/wd4z/` hacen referencia a la vía de acceso que se utilizó durante la instalación de Developer para System z, no de CARMA. El valor predeterminado es `/usr/lpp/wd4z/`, pero quizá no sea válido para su local.

Para configurar los componentes z/OS UNIX de CARMA, que se instalan durante el proceso de instalación de IBM Rational Developer para System z (FMID: HHOP710), siga estos pasos:

1. El archivo de configuración `CRASRV.properties` debe residir en el mismo directorio que el archivo `rsed.envvars` personalizado. Ambos archivos residen por defecto en el directorio de instalación (cuya vía de acceso predeterminada es `/usr/lpp/wd4z/rse/lib/`). Pero, como se describe en: “Guardar el archivo de configuración `rsed.envvars` en otro directorio” en la página 34, conviene que los copie en otro directorio para evitar que se sobrescriban al aplicar el mantenimiento. En los ejemplos que figuran en esta publicación, este directorio es `/etc/wd4z/`.

```
cp /usr/lpp/wd4z/rse/lib/CRASRV.properties /etc/wd4z
```

2. El archivo de configuración de ejemplo `CRASRV.properties` consta de un conjunto de definiciones de variables de entorno. Hay que cambiar el archivo de configuración de ejemplo para que coincida con los estándares de su local y contenga las sentencias enumeradas en: Figura 10, donde las líneas de comentarios empiezan por un signo de almohadilla (#).

```
# Opción de configuración de CARMA
#
port.start=5227
port.range=100
startup.script.name=/usr/lpp/wd4z/rse/lib/rexxsub
clist.dsname='hlq.SCRACLST(CRASUBMT)'
```

Figura 10. *CRASRV.properties* – archivo de configuración de CARMA

### **port.start**

Primer puerto utilizado para la comunicación entre MVS de CARMA y los componentes z/OS UNIX. El puerto predeterminado es 5227. La comunicación en este puerto está confinada a la máquina de hospedaje.

**Nota:** Antes de seleccionar un puerto, verifique que el puerto está disponible en su sistema, sirviéndose de los mandatos **NETSTAT** y **NETSTAT PORTL**. Hallará más información en: "Puertos TCP/IP reservados" en la página 83.

**port.range**

Rango de puertos, empezando por `port.start`, que se utilizará para la comunicación del servidor CARMA. El valor predeterminado es 100. Por ejemplo, cuando se utilizan los valores predeterminados, CARMA puede emplear los puertos comprendidos entre el 5227 y el 5326 (inclusive).

**startup.script.name**

Define la vía de acceso absoluta del script de sometimiento REXX `rexsub`. El valor predeterminado es `/usr/lpp/wd4z/rse/lib/rexsub`. Este exec REXX desencadenará la ejecución de la CLIST `CRASUBMT` en MVS.

**clist.dsname**

Define la ubicación de la CLIST `CRASUBMT`, mediante los convenios de referencias de MVS. La ubicación, si lleva apóstrofes ('), es absoluta, sin el prefijo de usuario que precede al nombre proporcionado para el conjunto de datos. El valor predeterminado es `'hlq.SCRACLST(CRASUBMT)'`. La instalación de SMP/E de CARMA que crea `CRASUBMT` emplea `CRA` como valor predeterminado de `hlq`. Esta CLIST iniciará un servidor CARMA cuando se abra una conexión.

**Nota:** En Developer para System z versión 7.0, el conjunto de datos CLIST y su nombre de miembro se han trasladado de `rexsub` (variable `DSNAME`) a `CRASRV.properties`, lo que evita la necesidad de personalizar `rexsub`. Deje `rexsub` en el directorio de instalación si desea que el posible mantenimiento de SMP/E se active automáticamente.

---

## (Opcional) Activar los gestores de acceso a repositorios (RAM) de ejemplo

Los gestores de acceso a repositorios (RAM) son API escritas por el usuario para intercambiar información con los gestores de configuraciones de software (SCM) de z/OS. Siga las instrucciones de los apartados que figuran más abajo para los RAM de ejemplo que desea activar.

**Nota:** Los RAM de ejemplo se proporcionan con el fin de probar la configuración del entorno CARMA y como ejemplos que le ayudarán a desarrollar sus propios RAM. NO debe utilizar los RAM de ejemplo proporcionados en un entorno de producción.

**Nota:** En este apartado, todas las referencias a `hlq` se refieren al calificador de alto nivel empleado durante la instalación de CARMA. El valor predeterminado de la instalación es `CRA`, pero quizá no sea válido para su local.

Para obtener más información sobre los RAM de ejemplo y el código fuente de ejemplo proporcionados, consulte el manual *Rational Developer para System z Common Access Repository Manager Developer's Guide* (SC23-7660).

## Activar el RAM de SCLM

1. Copie los miembros que hay que personalizar del directorio de instalación en una biblioteca personal y personalice esas copias, para evitar que se sobrescriban al aplicar el mantenimiento.
  - hlq.SCRASAMP(CRA#VSLM)
  - hlq.SCRASAMP(CRA#ASLM)
  - hlq.SCRACLST(CRASUBMT)
2. Personalice y someta el JCL de hlq.SCRASAMP(CRA#VSLM). Las instrucciones de personalización están en la documentación de CRA#VSLM. CRA#VSLM crea y pone a punto el VSAM de mensajes de SCLM.
3. Quite el carácter de comentario de la sentencia DD CRARAM2 en CRASUBMT y proporcione el nombre de conjunto de datos del VSAM de mensaje del RAM de SCLM. Observe que CRASUBMT se ha personalizado anteriormente en “Personalizar los componentes MVS de CARMA” en la página 57.
4. Personalice el JCL de hlq.SCRASAMP(CRA#ASLM). Las instrucciones de personalización están en la documentación de CRA#ASLM. CRA#ASLM asigna los conjuntos de datos que necesitan los clientes RAM de SCLM.

**Nota:** Cada usuario debe someter CRA#ASLM una vez antes de utilizar CARMA con el RAM de SCLM. Si no se hace así, se produce un error de asignación.

## Activar el RAM de PDS

1. Copie los miembros que hay que personalizar del directorio de instalación en una biblioteca personal y personalice esas copias, para evitar que se sobrescriban al aplicar el mantenimiento.
  - hlq.SCRASAMP(CRA#VPDS)
  - hlq.SCRACLST(CRASUBMT)
2. Personalice y someta el JCL de hlq.SCRASAMP(CRA#VPDS). Las instrucciones de personalización están en la documentación de CRA#VPDS. CRA#VPDS crea y pone a punto el VSAM de mensajes de RAM de PDS.
3. Quite el carácter de comentario de la sentencia DD CRARAM1 en CRASUBMT y proporcione el nombre de conjunto de datos del VSAM de mensaje del RAM de PDS. Observe que CRASUBMT se ha personalizado antes en “Personalizar los componentes MVS de CARMA” en la página 57.

## Activar el RAM de esqueleto

1. Copie los miembros que hay que personalizar del directorio de instalación en una biblioteca personal y personalice esas copias, para evitar que se sobrescriban al aplicar el mantenimiento.
  - hlq.SCRASAMP(CRA#CRAM)
  - hlq.SCRACLST(CRASUBMT)
2. Personalice y someta el JCL de hlq.SCRASAMP(CRA#CRAM). Las instrucciones de personalización están en la documentación de CRA#CRAM. CRA#CRAM compila el RAM del esqueleto.
3. Añada la biblioteca de carga que contiene el módulo RAM del esqueleto compilado, CRARAMSA, a la DD STEPLIB de CRASUBMT. Observe que CRASUBMT se ha personalizado anteriormente en “Personalizar los componentes MVS de CARMA” en la página 57.



---

## Capítulo 6. (Opcional) Activar IBM Software Configuration and Library Manager (SCLM) Developer Toolkit

IBM Software Configuration and Library Manager (SCLM) Developer Toolkit (FMID: HSD3310) proporciona las herramientas necesarias para ampliar las prestaciones de SCLM para el cliente. El propio SCLM es un gestor de código fuente basado en host que viene como parte de ISPF.

El SCLM Developer Toolkit, que viene junto con el producto Developer para System z, es un plug-in basado en Eclipse que intercambia información con SCLM y proporciona acceso a todos los procesos SCLM para el desarrollo de código de legado, así como soporte para el desarrollo completo de Java y J2EE en la estación de trabajo en sincronización con SCLM en el sistema central, incluidas las tareas de construir, ensamblar y desplegar el código J2EE desde el sistema central.

En la Tabla 2 en la página 1 encontrará una lista de los manuales que proporcionan más información sobre SCLM Developers Toolkit y su instalación, personalización y utilización.





---

## Capítulo 7. Consideraciones en torno al cliente Developer para System z

Los usuarios del cliente Developer para System z deben conocer el resultado de determinadas personalizaciones del host, como la de los números de puerto TCP/IP, para que el cliente funcione como es debido. Utilice la lista de comprobación de la Tabla 12 para reunir la información que necesite.

*Tabla 12. Lista de comprobación del cliente Developer para System z*

Personalización	Valor
Número de puerto del servidor del supervisor de trabajos JES (el valor predeterminado es 6715 ):  Vea SERV_PORT en: “Personalizar el archivo de configuración del supervisor de trabajos JES, FEJJCNFG” en la página 16.	
Ubicación de los procedimientos ELAXF* si no están en la biblioteca de procedimientos del sistema:  Vea la nota de JCLLIB en: “Personalizar procedimientos de construcción remota ELAXF*” en la página 22.	
Nombres de procedimiento y/o paso de los procedimientos ELAXF*, si se han cambiado  Vea la nota que indica cómo cambiarlos, en: “Personalizar procedimientos de construcción remota ELAXF*” en la página 22.	
Nombre de procedimiento almacenado DB2 (el valor predeterminado es ELAXMSAM):  Vea información sobre los procedimientos almacenados DB2, en: Apéndice A, “Ejecutar múltiples instancias de Developer para System z”, en la página 73.	
Ubicación del procedimiento almacenado DB2 si no está en una biblioteca de procedimientos del sistema:  Vea: “(Opcional) Personalizar miembros de procedimientos almacenados DB2 ELAXM*” en la página 27.	
Utilizar el método de conexión DAEMON, REXEC o SSH para RSE:  Vea: “Daemon INETD y configuración de REXEC/SSH de RSE” en la página 43.	
Número de puerto TCP/IP del daemon RSE (el valor predeterminado es 4035):  Vea: “Configuración del daemon RSE de INETD” en la página 43.	
Vía de acceso al script de shell server.zseries para REXEC/SSH (el valor predeterminado es /usr/lpp/wd4z/rse/lib, el valor aconsejado es /etc/wd4z):  Vea: “Configuración de REXEC (o SSH) de INETD” en la página 45.	

Tabla 12. Lista de comprobación del cliente Developer para System z (continuación)

Personalización	Valor
<p>Número de puerto de REXEC o SSH (el valor predeterminado es 512 ó 22, respectivamente):</p> <p>Vea: “Configuración de REXEC (o SSH) de INETD” en la página 45.</p> <p><b>Nota:</b> Para las acciones remotas (basadas en host) de los subproyectos z/OS UNIX, es necesario que REXEC o SSH esté activo en el host.</p>	
<p>Ubicación del JCL de CRA#ASLM para las asignaciones de conjuntos de datos RAM de SCLM de CARMA:</p> <p>Vea la nota de CRA#ASLM, en: “Activar el RAM de SCLM” en la página 61.</p>	

---

## Capítulo 8. Consideraciones sobre el rendimiento

z/OS es un sistema operativo sumamente personalizable, y los cambios de sistema (a veces pequeños) pueden afectar considerablemente al rendimiento global. En este capítulo se resaltan algunos de los cambios que se pueden hacer para mejorar el rendimiento de Developer para System z.

En los manuales *MVS Initialization and Tuning Guide (SA22-7591)* y *UNIX System Services Planning (GA22-7800)* hallará más información sobre cómo ajustar el sistema.

---

### Utilizar sistemas de archivos zFS

zFS (sistema de archivos de zSeries) y HFS (sistema de archivos jerárquico) son sistemas de archivos UNIX que se pueden usar en un entorno UNIX de z/OS. Sin embargo, zFS proporciona las siguientes ventajas y características:

- Aumento del rendimiento en muchos entornos de cliente al acceder a archivos con un tamaño cercano a 8K que se actualizan con frecuencia. El rendimiento del acceso a archivos de menor tamaño es equivalente al del HFS.
- Clonación solo de lectura de un sistema de archivos en el mismo conjunto de datos. El sistema de archivos clonado se puede poner a disposición de los usuarios para proporcionar una copia puntual solo de lectura de un sistema de archivos. Esta es una característica opcional que solo está disponible en un entorno que no sea sysplex.
- zFS es el sistema de archivos UNIX estratégico de z/OS. La funcionalidad del HFS se ha estabilizado y las mejoras realizadas en el sistema de archivos solo serán para zFS.

En el manual *UNIX System Services Planning (GA22-7800)* encontrará más información sobre zFS.

---

### Evitar el uso de STEPLIB

Cada proceso z/OS UNIX que tenga una STEPLIB que se propague de padre a hijo o a través de un exec consumirá unos 200 bytes de área de almacenamiento común ampliada (ECSA). Si no se define ninguna variable de entorno STEPLIB, o si se define como STEPLIB=CURRENT, z/OS UNIX propaga todas las asignaciones de TASKLIB, STEPLIB y JOBLIB actualmente activas durante una función fork(), spawn() o exec(). El servidor RSE inicia varios procesos, y cada conexión de cliente tiene un servidor RSE privado. Esto puede hacer que las cifras aumenten con rapidez.

Developer para System z tiene el valor predeterminado STEPLIB=NONE codificado en `rsed.envvars`, como se describe en: “Personalizar `rsed.envvars`, el archivo de configuración de RSE” en la página 35. Por los motivos mencionados más arriba, no conviene cambiar esta directiva, y en cambio es aconsejable colocar los conjuntos de datos tomados como objetivo en LINKLIST o en LPA (área de módulos residentes).

Si no utiliza la directiva STEPLIB, debe verificar el contenido de `rsed.envvars` para ver si la sentencia STEPLIB es la primera o no.

- Si la última directiva STEPLIB definida anteriormente en `rsed.envvars` es igual a STEPLIB=NONE

STEPLIB=first.steplib.dataset:second.steplib.dataset

- Si la última directiva STEPLIB definida anteriormente en rsed.envvars no es igual a STEPLIB=NONE

STEPLIB=\$STEPLIB:first.steplib.dataset:second.steplib.dataset

---

## Mejorar el acceso a las bibliotecas del sistema

Algunas bibliotecas del sistema y algunos módulos de carga se utilizan muy a menudo en z/OS UNIX y en las actividades de desarrollo de aplicaciones. El hecho de mejorar el acceso a ellas (por ejemplo, añadirlas al área de módulos residentes, LPA) puede mejorar el rendimiento del sistema. En el manual *MVS Initialization and Tuning Reference (SA22-7592)* hallará más información sobre cómo cambiar los miembros SYS1.PARMLIB descritos a continuación.

### Bibliotecas de tiempo de ejecución de Language Environment (LE)

Los programas C (incluida la shell de z/OS UNIX), cuando se ejecutan, suelen utilizar rutinas de la biblioteca de tiempo de ejecución de Language Environment (LE). Como promedio, unos 4 MB de la biblioteca de tiempo de ejecución se cargan en memoria para cada espacio de direcciones que se ejecute en un programa habilitado para LE, y se copian en cada bifurcación.

El conjunto de datos CEE.SCEELPA contiene un subconjunto de rutinas de tiempo de ejecución de LE, que se utilizan muy a menudo en z/OS UNIX. Conviene añadir este conjunto de datos a SYS1.PARMLIB(LPALSTxx) para obtener el máximo rendimiento. Así, los módulos se leen del disco una sola vez y se colocan en una ubicación compartida.

**Nota:** Añada la siguiente sentencia a SYS1.PARMLIB(PROGxx) si prefiere añadir los módulos de carga a la LPA (área de módulos residentes) dinámica:

```
LPA ADD MASK(*) DSNAME(CEE.SCEELPA)
```

Conviene asimismo colocar las bibliotecas de tiempo de ejecución de LE CEE.SCEERUN y CEE.SCEERUN2 en LINKLIST, añadiendo los conjuntos de datos a SYS1.PARMLIB(LNKLSTxx) o a SYS1.PARMLIB(PROGxx). Ello elimina la actividad adicional que supone utilizar la STEPLIB de z/OS UNIX, y se reduce la entrada/salida debido a la gestión por parte de LLA y VLF, o de productos similares.

**Nota:** Añada la biblioteca de clases DLL de C/C++ CBC.SCLBDLL también a LINKLIST, por los mismos motivos.

Si decide que no quiere poner estas bibliotecas en LINKLIST, debe configurar la sentencia STEPLIB pertinente en rsed.envvars, como se describe en: "Personalizar rsed.envvars, el archivo de configuración de RSE" en la página 35. Aunque este método siempre utiliza almacenamiento virtual adicional, podrá mejorar el rendimiento definiendo las bibliotecas de tiempo de ejecución de LE en LLA o en un producto similar. Esto reduce la E/S necesaria para cargar los módulos.

### Desarrollo de aplicaciones

En los sistemas cuya actividad primaria es el desarrollo de aplicaciones, el rendimiento también podrá mejorar si pone el editor de enlazamiento en la LPA dinámica, añadiendo estas líneas a SYS1.PARMLIB(PROGxx):

```
LPA ADD MODNAME(CEEINIT,CEEBLIBM,CEEV003,EDCV) DSNAME(CEE.SCEERUN)
LPA ADD MODNAME(IEFIB600,IEFXB603) DSNAME(SYS1.LINKLIB)
```

En el caso del desarrollo C/C++, también puede añadir el conjunto de datos de compilador CBC.SCCNCMP a SYS1.PARMLIB(LPALSTxx).

Las sentencias anteriores son ejemplos de posibles candidatos a la LPA, pero las necesidades en su local puede ser distintas. En el manual *Language Environment Customization (SA22-7564)* hallará información sobre cómo poner otros módulos de carga de LE en la LPA dinámica. Consulte el manual *UNIX System Services Planning (GA22-7800)* para obtener más información sobre cómo poner módulos de carga de compilador C/C++ en la LPA dinámica.

---

## Mejorar el rendimiento de la comprobación de seguridad

Para mejorar el rendimiento de la comprobación de seguridad que se realiza para z/OS UNIX, defina el perfil BPX.SAFFASTPATH en la clase FACILITY del software de seguridad. Así se reduce la actividad adicional que supone realizar comprobaciones de seguridad en z/OS UNIX para una gran variedad de operaciones. Entre ellas está la comprobación de acceso a los archivos de inclusión, la comprobación de acceso a IPC y la comprobación de ser propietario del proceso. En el manual *UNIX System Services Planning (GA22-7800)* hallará más información sobre este perfil.

**Nota:** Los usuarios no necesitan tener autorización sobre el perfil BPX.SAFFASTPATH.

---

## Compartir clases entre las JVM

La máquina virtual Java (JVM) de IBM, versión 5 o superior, le permite compartir clases de rutina de carga y de aplicación entre las JVM, almacenándolas en una caché de memoria compartida. El hecho de compartir clases reduce el consumo global de memoria virtual cuando hay más de una JVM que comparte una caché. El hecho de compartir clases también reduce el tiempo de arranque de una JVM después de haberse creado la caché.

La caché de clases compartidas es independiente de las JVM activas y persiste más allá del tiempo de vida de la JVM que creó la caché. Dado que la caché de clases compartidas persiste más allá del tiempo de vida de las JVM, la caché se actualiza dinámicamente para reflejar las modificaciones que se hayan podido hacer en los JAR o en las clases del sistema de archivos.

La actividad adicional que supone crear y poblar una caché nueva es mínima. El coste en tiempo del arranque de una sola JVM se suele situar entre 0 y el 5% más de tiempo si se compara con un sistema que no utilice clases compartidas, y depende de la cantidad de clases cargadas. La mejora del tiempo de arranque de JVM con una caché poblada se suele situar entre el 10% y el 40% menos de tiempo si se compara con un sistema que no utilice clases compartidas, y depende del sistema operativo y del número de clases que se carguen. Si hay múltiples JVM en ejecución concurrente, el tiempo de arranque global mejorará.

En el manual *Java SDK and Runtime Environment User Guide* obtendrá más información sobre clases compartidas.

## Habilitar la prestación de compartir clases

Para habilitar la prestación de compartir clases en el servidor RSE, descomente la siguiente directiva en `rsed.envvars`, como se describe en: “(Opcional) Definir

parámetros de arranque Java adicionales con `_RSE_*OPTS` en la página 41. La primera sentencia define una caché llamada RSE con acceso de grupo, y permite que el servidor RSE se inicie incluso si falla la prestación de compartir clases. La segunda sentencia es opcional y establece que el tamaño de la caché sea igual a 6 megabytes (el valor predeterminado del sistema es de 16 MB).

```
_RSE_CLASS_OPTS=-Xshareclasses:name=RSE,groupAccess,nonFatal
#_RSE_CLASS_OPTS="$_RSE_CLASS_OPTS -Xscmx6m
```

**Nota:** Como ya se ha mencionado en: “Seguridad de la caché”, todos los usuarios que utilicen la clase compartida deben tener el mismo ID de grupo primario (GID). Esto significa que los usuarios deben tener definido el mismo grupo predeterminado en el software de seguridad, o que los distintos grupos predeterminados tengan el mismo GID en el correspondiente segmento OMVS.

## Límites de tamaño de la caché

El tamaño máximo de la caché teórica compartida es 2 GB. El tamaño de caché que se puede especificar está limitado por la cantidad de memoria física y de espacio de intercambio físico disponible en el sistema. Dado que el espacio de direcciones virtuales de un proceso se comparte entre la caché de clases compartidas y la memoria dinámica Java, si se aumenta el tamaño máximo de la memoria dinámica Java, disminuye el tamaño de la caché de clases compartidas que se puede crear.

## Seguridad de la caché

El acceso a la caché de clases compartidas está limitado por los permisos del sistema operativo y por los permisos de la seguridad Java.

Por defecto, las cachés de clases se crean con seguridad a nivel de usuario, por lo que el usuario que ha creado la caché es el único que puede acceder a ella. En z/OS UNIX, existe una opción, `groupAccess`, que da acceso a todos los usuarios del grupo primario del usuario que creó la caché. Sin embargo, sea cual sea el nivel de acceso que se emplee, el único que puede destruir una caché es el usuario que la ha creado o un usuario root (UID 0).

En el manual *Java SDK and Runtime Environment User Guide* obtendrá más información sobre las opciones de seguridad adicionales al utilizar un gestor de seguridad Java.

## SYS1.PARMLIB(BPXPRMxx)

Algunos de los valores de `SYS1.PARMLIB(BPXPRMxx)` afectan al rendimiento de las clases compartidas. Si se emplean valores incorrectos, las clases compartidas podrían dejar de funcionar. Estos valores también podrían afectar al rendimiento. Para obtener más información sobre cómo se utilizan estos parámetros y sobre cómo afectan al rendimiento, consulte los manuales *MVS Initialization and Tuning Reference (SA22-7592)* y *UNIX System Services Planning (GA22-7800)*. Los parámetros `BPXPRMxx` más significativos que afectan al funcionamiento de las clases compartidas son:

- `MAXSHAREPAGES`, `IPCSHMSPAGES`, `IPCSHMMPAGES` y `IPCSHMNSEGS`.

Estos valores afectan a la cantidad de páginas de memoria compartida disponibles para la JVM. El tamaño de páginas compartidas en el caso de un servicio de sistema z/OS UNIX de 31 bites se ha fijado en 4 KB. Las clases compartidas intentan crear una caché de 16 MB por defecto. Por lo tanto, establezca que `IPCSHMMPAGES` sea mayor que 4096.

Si establece un tamaño de caché utilizando `-Xscmx`, la JVM redondeará el valor por exceso al megabyte más cercano. Debe tenerlo en cuenta cuando establezca `IPCSHMPAGES` en su sistema.

- `IPCSEMNIDS` e `IPCSEMNSEMS`

Estos valores afectan a la cantidad de semáforos disponibles en los procesos UNIX. Las clases compartidas utilizan semáforos IPC para la comunicación entre máquinas virtuales Java (JVM).

## Espacio en disco

La caché de clases compartidas necesita espacio en disco en el que almacenar información de identificación sobre las cachés que existen en el sistema. Esta información se almacena en `/tmp/javasharedresources`. Si el directorio de información de identificación se suprime, la JVM no puede identificar las clases compartidas en el sistema y debe volver a crear la caché.

## Utilidades para la gestión de cachés

El mandato de línea Java `-Xshareclasses` puede tener varias opciones, algunas de las cuales son utilidades para la gestión de cachés. Tres de ellas se muestran en el ejemplo que sigue (\$ es el indicador de z/OS UNIX). En el manual *Java SDK and Runtime Environment User Guide* encontrará una visión general completa de las opciones de línea de mandatos soportadas.

```
$ java -Xshareclasses:listAllCaches
Cachá compartida      OS shmid      en uso      Hora de última desconexión
RSE                   401412        0           Lun Jun 18 17:23:16 2007
```

No se ha podido crear la máquina virtual Java (JVM).

```
$ java -Xshareclasses:name=RSE,printStats
```

Estadísticas actuales de la caché "RSE":

```
dirección base      = 0x0F300058
dirección final     = 0x0F8FFFF8
puntero de asignación = 0x0F4D2E28
```

```
tamaño de caché     = 6291368
bytes libres        = 4355696
bytes de ROMClass   = 1912272
bytes de metadatos  = 23400
% de metadatos usados = 1%
```

```
nº de ROMClasses    = 475
nº de vías de acceso de clases = 4
nº de URL           = 0
nº de símbolos      = 0
nº de clases obsoletas = 0
% de clases obsoletas = 0%
```

La caché está 30% llena

No se ha podido crear la máquina virtual Java (JVM).

```
$ java -Xshareclasses:name=RSE,destroy
JVMSHRC010I La caché compartida "RSE" se ha destruido
No se ha podido crear la máquina virtual Java (JVM).
```

**Nota:** Las utilidades de caché realizan la operación necesaria en la caché especificada sin iniciar la JVM, por lo tanto, el mensaje "No se ha podido crear la máquina virtual Java (JVM)." es normal.



**Nota:** Solo se puede destruir una caché si han concluido todas las JVM que la utilizan, y si el usuario tiene permisos suficientes.

---

## Memoria dinámica Java de tamaño fijo

Con una memoria dinámica de tamaño fijo, no se producen ampliaciones ni contracciones de la memoria dinámica, lo que puede aumentar notablemente el rendimiento en algunas situaciones. Sin embargo, el hecho de utilizar una memoria dinámica de tamaño fijo no suele ser una buena idea, porque retarda el inicio de la recogida de basura hasta que la memoria dinámica esté llena, y en ese momento pasará a ser una tarea importante. También aumenta el riesgo de fragmentación, lo que exige una compactación de la memoria dinámica. Por lo tanto, solo debe utilizar memorias dinámicas de tamaño fijo después de haberlas probado debidamente o cuando así lo indique el centro de soporte de IBM. En el manual *Java Diagnostics Guide (SC34-6650)* hallará más información sobre los tamaños de la memoria dinámica y sobre la recogida de basura.

Por defecto, el tamaño de la memoria dinámica inicial de una máquina virtual Java (JVM) en z/OS es de 1 megabyte. El tamaño máximo es de 64 megabytes. Los límites se pueden establecer con las opciones `-Xms` (inicial) y `-Xmx` (máximo) de la línea de mandatos Java.

En Developer para System z, las opciones de línea de mandatos Java se definen en la directiva `_RSE_JAVA_OPTS` del archivo `rsed.envvars`, como se describe en: “(Opcional) Definir parámetros de arranque Java adicionales con `_RSE_*OPTS`” en la página 41.

```
#_RSE_JAVA_OPTS="$_RSE_JAVA_OPTS -Xms128m -Xmx128m"
```

---

## Gestión de cargas de trabajo (WLM)

Cada local tiene sus necesidades específicas, y puede personalizar el sistema operativo z/OS para poder sacar el mayor partido de los recursos disponibles y responder a dichas necesidades. Con la gestión de cargas de trabajo (WLM), se definen objetivos de rendimiento y se asigna una importancia de negocio a cada objetivo. Los objetivos se definen para el trabajo en términos de negocio, y el sistema decide la cantidad de recursos (por ejemplo, la cantidad de CPU y de almacenamiento) que hay que dar al trabajo para responder a su objetivo.

El rendimiento de Developer para System z se puede equilibrar estableciendo los objetivos correctos para sus procesos. A continuación figuran algunas directrices generales.

- Asigne la transacción APPC a un grupo de rendimiento TSO.
- Asigne el servidor RSE a un grupo de rendimiento TSO, o a uno situado justo debajo de él, si tiene una utilización de TSO significativa.

Consulte el manual *MVS Planning Workload Management (SA22-7602)* para obtener más información al respecto.



---

## Apéndice A. Ejecutar múltiples instancias de Developer para System z

En algunas ocasiones le interesará tener múltiples instancias de Developer para System z activas en el mismo sistema; por ejemplo, al probar una ampliación. Sin embargo, algunos recursos como los puertos TCP/IP no se pueden compartir, por lo que los valores predeterminados no siempre son aplicables. Utilice la información de este apéndice para planificar la coexistencia de distintas instancias de Developer para System z, y después podrá usar esta guía de configuración para personalizarlas.

Aunque es posible compartir algunas partes de Developer para System z entre 2 (o más) instancias, es mejor que NO lo haga, a menos que los correspondientes niveles de software sean idénticos y que los únicos cambios estén en los miembros de configuración. Developer para System z deja suficiente margen de personalización para crear múltiples instancias que no se solapen, y le aconsejamos encarecidamente que utilice estas características.

---

### Archivos de configuración diferentes con idéntico nivel de software

En un conjunto de circunstancias limitado, puede compartir todo salvo (algunos de) los componentes personalizables. Por ejemplo, proporcionar acceso no SSL para la utilización en el local y comunicación codificada por SSL para la utilización fuera del local.

**Atención:** La configuración compartida NO se puede usar de manera segura para probar el mantenimiento, un avance tecnológico o un nuevo release.

Para configurar otra instancia de una instalación activa de Developer para System z, rehaga los pasos de personalización para los componentes que sean distintos, utilizando diferentes conjuntos de datos/directorios/puertos para evitar que se solape con la instalación actual.

En el ejemplo SSL mencionado más arriba, las personalizaciones del servidor RSE actual se pueden clonar y, después, el archivo `ssl.properties` clonado se puede actualizar. Las personalizaciones de MVS (supervisor de trabajos JES, etc.) se pueden compartir entre las instancias SSL y las no SSL. Ello daría como resultado las siguientes acciones:

1. Crear un nuevo directorio en el que poner los miembros de configuración personalizados SSL
2. Copiar los miembros de configuración activos en este directorio
3. Actualizar el archivo `ssl.properties` copiado para proporcionar la información relacionada con SSL
4. Definir un nuevo número de puerto para el daemon RSE en `/etc/services`
5. Definir un nuevo proceso de daemon RSE en `/etc/inetd.conf`, utilizando el nuevo directorio para el parámetro `-d`
6. Reiniciar INETD para activar el nuevo daemon RSE

---

## Todas las demás situaciones

Cuando hay cambios de código implicados (de mantenimiento, avances tecnológicos, nuevo release), o cuando los cambios son bastante complejos, lo mejor es hacer otra instalación de Developer para System z. En este apartado se describen los posibles puntos de conflicto entre distintas instalaciones.

La siguiente lista es una visión general resumida de los elementos que deben ser distintos (o conviene que lo sean) entre las instancias de Developer para System z:

- Bibliotecas de instalación
- Puerto TCP/IP del supervisor de trabajos JSE, así como su archivo de configuración FEJJC�FG
- JCL de inicio del supervisor de trabajos JES
- Nombre de la transacción APPC
- Archivo de configuración de RSE, rsed.envvars
- Puerto TCP/IP del daemon RSE

A continuación se proporciona una visión general más detallada.

- Bibliotecas de instalación
  1. Instale cada instancia de Developer para System z en conjuntos de datos y directorios separados. Tenga presente que solo puede cambiar la vía de acceso z/OS UNIX prefijando el valor predeterminado de IBM, que es /usr/lpp/wd4z. Por ejemplo, /service/usr/lpp/wd4z.
- Componentes obligatorios
  1. El archivo de configuración del supervisor de trabajos JES, hlq.SFEKSAMP(FEJJC�FG), contiene el número de puerto TCP/IP del supervisor de trabajos JES y, por lo tanto, no se puede compartir. El propio miembro se puede red denominar (si también se actualiza el JCL), lo que le permite colocar todas las versiones personalizadas de este miembro en el mismo conjunto de datos, si no va a hacer las actualizaciones en el conjunto de datos de instalación.
  2. El JCL de arranque del supervisor de trabajos JES, hlq.SFEKSAMP(FEJJJCL), hace referencia a FEJJC�FG y, por lo tanto, tampoco se puede compartir. Tras red denominar el miembro (y la tarjeta JOB), puede colocar todos los JCL en el mismo conjunto de datos.
  3. La transacción APPC tiene una referencia a hlq.SFEKPROC(FEKFRSRV), el servidor de mandatos TSO. Este es específico del nivel de software, por lo que debe crear una transacción APPC por cada instancia. Observe que, desde la versión 7.1, se puede acceder al servidor de mandatos TSO mediante una función de SCLM Developer Toolkit.
  4. El archivo de configuración ISPF.conf (SCLM Developer Toolkit) tiene una referencia a hlq.SFEKPROC(FEKFRSRV), el servidor de mandatos TSO. Es específico del nivel de software, por lo que debe crear un archivo ISPF.conf por cada instancia. Tenga en cuenta que solo es necesario si utiliza SCLM Developer Toolkit para acceder al servidor de mandatos TSO.
  5. El archivo de configuración de RSE, /usr/lpp/wd4z/rse/lib/rsed.envvars, contiene referencias a la vía de instalación y al nombre de la transacción APPC, que tiene que ser exclusivo. El nombre del archivo es obligatorio, por lo tanto, no podrá conservar distintas copias en el mismo directorio. Tenga presente que, dado que el nombre de la transacción APPC ha cambiado, hay que definir la variable \_FEKFCMD\_TP\_NAME\_ en rsed.envvars.

6. Todos los archivos de configuración y scripts de shell obligatorios deben residir en el mismo directorio, dado que rsed.envvars no se puede compartir, por lo que rsed.envvars debe estar en una ubicación no compartida.
  7. Los puertos TCP/IP de REXEC y SSH se pueden compartir sin restricciones.
  8. El daemon RSE no se puede compartir, porque cada daemon RSE es propietario de su propio puerto TCP/IP. Hay que crear un nuevo daemon que tenga un nombre de servicio y un puerto distintos.
- Componentes opcionales

1. Algunos procedimientos ELAXF\* tienen una referencia a hlq.SFEKLOAD, la biblioteca de carga de Developer para System z. Para encontrar una posible solución de cómo hacer que distintos conjuntos estén disponibles para los usuarios, vea la nota sobre JCLLIB en: "Personalizar procedimientos de construcción remota ELAXF\*" en la página 22.
2. Para activar dos instancias del procedimiento almacenado DB2, hay que llevar a cabo las siguientes tareas. Sin embargo, tenga en cuenta que esta descripción no está soportada tal cual.
  - a. Copie hlq.SFEKPROC(ELAXMREX) en un miembro cuyo nombre sea diferente; por ejemplo, ELAXMRXX
  - b. Copie el miembro de ejemplo hlq.SFEKSAMP(ELAXMSAM) en un miembro cuyo nombre sea diferente; por ejemplo, ELAXMWDZ.
  - c. Cambie el miembro de ejemplo hlq.SFEKSAMP(ELAXMJCL) para que refleje estos cambios de nombre; por ejemplo:

```
//SYSIN DD *
CREATE PROCEDURE SYSPROC.ELAXMRXX
  ( IN FUNCTION_REQUEST  VARCHAR(20)      CCSID EBCDIC
  ...
  , OUT RETURN_VALUE     VARCHAR(255)     CCSID EBCDIC )
PARAMETER STYLE GENERAL RESULT SETS 1
LANGUAGE REXX
COLLID DSNREXCS          WLM ENVIRONMENT ELAXMWDZ
PROGRAM TYPE MAIN        MODIFIES SQL DATA
STAY RESIDENT NO         COMMIT ON RETURN NO
ASUTIME NO LIMIT         SECURITY USER;

COMMENT ON PROCEDURE SYSPROC.ELAXMRXX IS
  'PLI & COBOL PROCEDURE PROCESSOR (ELAXMRXX), INTERFACE LEVEL 0.01';

GRANT EXECUTE ON PROCEDURE SYSPROC.ELAXMRXX TO PUBLIC;
//
```

- d. Siga con la personalización, pero ahora con los nuevos miembros, como se describe en: "(Opcional) Personalizar miembros de procedimientos almacenados DB2 ELAXM\*" en la página 27.
  - e. El nombre nuevo del entorno WLM (por ejemplo, ELAXMWDZ) se debe utilizar en el asistente de procedimientos almacenados DB2 en el cliente.
3. bidi se basa en un miembro de la biblioteca de carga y, por lo tanto, no se puede compartir entre releases. Sin embargo, si el nombre del módulo de carga es idéntico para todas las instancias, podrá compartir la versión más reciente entre las instancias, incluso en los distintos releases. La compatibilidad hacia atrás no está disponible si el módulo de carga ha cambiado de nombre.
  4. Los módulos de carga ADM son compatibles hacia atrás y, por lo tanto, la versión más reciente se puede compartir en los distintos releases.
  5. El VSAM CRD de ADM es compatible hacia atrás y, por lo tanto, la versión más reciente se puede compartir en los distintos releases.

6. Las definiciones de recursos CICS de ADM son compatibles hacia atrás y, por lo tanto, la versión más reciente se puede compartir en los distintos releases.
7. Todos los archivos de configuración y scripts de shell opcionales deben residir en el mismo directorio, dado que `rsed.envvars` no se puede compartir, por lo que `rsed.envvars` debe estar en una ubicación no compartida.
8. En `rsed.envvars` puede activar la posibilidad de compartir las clases Java entre múltiples servidores RSE, actualizando la directiva `_RSE_JAVAOPTS`. El grupo de compartimiento debe tener un nombre exclusivo (`-Xshareclasses:name=RSE`).
9. Los VSAM de CARMA podrían cambiar entre niveles de software, por lo que no conviene compartirlos.

---

## Apéndice B. Resolución de problemas de configuración

Este apéndice se propone ayudarle a resolver algunos problemas comunes que pueden surgir durante la configuración de Developer para System z.

Encontrará más información en la sección de soporte del sitio Web de Developer para System z (<http://www-306.ibm.com/software/awdtools/devzseries/support/>), donde hay fichas técnicas que le aportarán la información más reciente de nuestro equipo de soporte.

En la sección de biblioteca del sitio Web también hallará la versión más reciente de la documentación de Developer para System z, incluidos los libros blancos.

También encontrará información valiosa en la biblioteca Internet de z/OS, disponible en <http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/>

---

### Ubicación de los archivos de anotaciones

Developer para System z crea archivos de anotaciones que le ayudarán a usted y al centro de soporte de IBM a identificar y resolver problemas. La lista que sigue es una visión general de los archivos de anotaciones que se pueden crear. Junto a estos archivos de anotaciones específicos del producto, no olvide consultar SYSLOG por si hay mensajes relacionados.

Las anotaciones basadas en MVS se pueden localizar mediante la sentencia DD pertinente. Los archivos de anotaciones basados en z/OS UNIX se encuentran en los siguientes directorios:

- `home/.eclipse/RSE/USERID/`

La mayoría de los archivos de anotaciones se encuentran en `home/.eclipse/RSE/USERID/`, donde `home` es la vía de acceso inicial definida en el segmento OMVS del usuario (o en el segmento OMVS predeterminado, si el usuario no tiene un segmento OMVS) y `USERID` es el ID de usuario de inicio de sesión (en mayúsculas).

- `daemon.log` - Anotaciones del daemon
- `fa.log` - Anotaciones de la integración del analizador de faltas
- `fekfivpc.log` - Anotaciones de la prueba del IVP `fekfivpc`
- `ffs.log` - Anotaciones del servidor FFS, que ejecuta funciones nativas
- `ffsget.log` - Anotaciones del lector de archivos, que lee un conjunto de datos secuencial (SDS) o un miembro PDS
- `ffsput.log` - Anotaciones del transcriptor de archivos, que escribe un conjunto de datos secuencial (SDS) o un miembro PDS
- `lock.log` - Anotaciones del gestor de bloqueos, que bloquea/desbloquea un conjunto de datos secuencial (SDS) o un miembro PDS
- `rmt_class_loader.cache.jar` - Caché de las clases cargadas por el cargador de clases remoto RSE
- `rsecomm.log` - Anotaciones del servidor RSE, que maneja mandatos del cliente (puede contener un rastreo de la pila de excepciones Java)
- `stderr.log` - Datos redirigidos de `stderr`, que es la salida de errores estándar
- `stdout.log` - Datos redirigidos de `stdout`, que es la salida estándar

- **/tmp/**  
En /tmp/ están los archivos de anotaciones creados antes de haberse verificado el ID de usuario.  
– rsedaemon.log - Anotaciones del daemon antes del inicio de sesión

## Anotaciones del supervisor de trabajos JES

- **DD SYSOUT**  
Anotaciones de las operaciones normales. El valor predeterminado del JCL de ejemplo hlq.SFEKSAMP(FEJJJCL) es SYSOUT=\*.
- **DD SYSPRINT**  
Anotaciones de rastreo. El valor predeterminado del JCL de ejemplo hlq.SFEKSAMP(FEJJJCL) es SYSOUT=\*. El rastreo se activa con el parámetro -TV; hallará más detalles en: “Personalizar el JCL de arranque del supervisor de trabajos JES” en la página 18.

## Anotaciones de la transacción APPC (servicio de mandatos TSO)

- **DD SYSPRINT**  
El programa de utilidad de administración APPC, cuando añade y modifica un perfil de programa de transacción (TP), comprueba el perfil del TP y su JCL por si hay errores de sintaxis. En esta fase, los datos de salida constan de los mensajes de error de sintaxis del perfil del TP, los mensajes de proceso del programa de utilidad y de las sentencias de conversión del JCL. Las anotaciones de los mensajes de esta fase se controlan mediante la sentencia DD SYSPRINT para el programa de utilidad ATBDSDFMU. El valor predeterminado en el JCL de ejemplo hlq.SFEKSAMP(FEKAPPC) es SYSOUT=\*. Vea el manual *MVS Planning APPC/MVS Management (SA22-7599)* si desea obtener más detalles.
- **&SYSUID.FEKFRSRV.&TPDATE.&TPTIME.LOG**  
Cuando se ejecuta un TP, los mensajes de tiempo de ejecución del TP (como los mensajes de asignación y los de terminación) se escriben en un archivo de anotaciones nombrado por la palabra clave MESSAGE\_DATA\_SET en el correspondiente perfil del TP. El valor predeterminado en el JCL de ejemplo hlq.SFEKSAMP(FEKAPPC) es &SYSUID.FEKFRSRV.&TPDATE.&TPTIME.LOG. Vea el manual *MVS Planning APPC/MVS Management (SA22-7599)* si desea obtener más detalles.

**Nota:** En función de sus definiciones de la transacción APPC y de los valores predeterminados del local, este archivo de anotaciones podría no aparecer, a menos que se añada la palabra clave KEEP\_MESSAGE\_LOG(ALWAYS) a las definiciones de la transacción. Para obtener más información al respecto, consulte el manual *MVS Planning APPC/MVS Management (SA22-7599)*.

## Anotaciones de RSE

- **home/.eclipse/RSE/USERID/**  
Los componentes relacionados con RSE crean varios archivos de anotaciones, la mayoría de los cuales se encuentran en home/.eclipse/RSE/USERID/, donde home es la vía de acceso inicial definida en el segmento OMVS del usuario (o en el segmento OMVS predeterminado, si el usuario no tiene un segmento OMVS) y USERID es el ID de usuario de inicio de sesión (en mayúsculas).  
– daemon.log - Anotaciones del daemon  
– ffs.log - Anotaciones del servidor FFS, que ejecuta funciones nativas de MVS

- `ffsget.log` - Anotaciones del lector de archivos, que lee un conjunto de datos secuencial (SDS) o un miembro PDS
- `ffsput.log` - Anotaciones del transcriptor de archivos, que escribe un conjunto de datos secuencial (SDS) o un miembro PDS
- `lock.log` - Anotaciones del gestor de bloqueos, que bloquea/desbloquea un conjunto de datos secuencial (SDS) o un miembro PDS
- `rmt_class_loader.cache.jar` - Caché de las clases cargadas por el cargador de clases remoto RSE
- `rsecomm.log` - Anotaciones del servidor RSE, que maneja mandatos del cliente (puede contener un rastreo de la pila de excepciones Java)
- `stderr.log` - Datos redirigidos de `stderr`, que es la salida de errores estándar.
- `stdout.log` - Datos redirigidos de `stdout`, que es la salida estándar.

La cantidad de datos escritos en `ffs*.log`, `lock.log` y `rsecomm.log` se controla mediante el valor `debug_level` existente en `rsecomm.properties`. Encontrará más detalles en: “(Opcional) Personalizar la configuración del rastreo RSE, `rsecomm.properties`” en la página 53.

- **`/tmp/rsedaemon.log`**

Este archivo contiene anotaciones del daemon antes del inicio de sesión.

## Anotaciones de prueba de IVP de fekfivpc

- **`home/.eclipse/RSE/USERID/fekfivpc.log`**

Salida del mandato `fekfivpc -file` (prueba del IVP relacionada con SCLM Developer Toolkit), donde `home` es la vía de acceso del directorio inicial definida en el segmento OMVS del usuario (o en el segmento OMVS predeterminado si el usuario no tiene un segmento OMVS) y `USERID` es el ID de usuario de inicio de sesión (en mayúsculas).

Encontrará más detalles en: “Conexión del servicio de mandatos TSO (utilizando SCLMDT)” en la página 50.

## Anotaciones de integración del analizador de faltas

- **`home/.eclipse/RSE/USERID/fa.log`**

Anotaciones de la integración del analizador de faltas, donde `home` es la vía de acceso inicial definida en el segmento OMVS del usuario (o en el segmento OMVS predeterminado, si el usuario no tiene un segmento OMVS) y `USERID` es el ID de usuario de inicio de sesión (en mayúsculas).

## Anotaciones de integración del gestor de archivos

- **Trabajo servidor de integración del gestor de archivos**

Al abrir una conexión con el gestor de archivos, se iniciará un trabajo servidor cuyo propietario será el ID del usuario. Su nombre es `FEKpuerto`, siendo `puerto` el número de puerto TCP/IP que se utiliza.

- **DD SYSPRINT**

El `SYSPRINT` del trabajo servidor contiene las anotaciones del servidor FMI.

- **`home/.eclipse/RSE/USERID/rsecomm.log`**

Anotaciones de comunicación de la FMI, donde `home` es la vía de acceso inicial definida en el segmento OMVS del usuario (o en el segmento OMVS predeterminado, si el usuario no tiene un segmento OMVS) y `USERID` es el ID de usuario de inicio de sesión (en mayúsculas).



La cantidad de datos escritos en este archivo se controla mediante el valor `debug_level` existente en `rsecomm.properties`. Encontrará más detalles en: “(Opcional) Personalizar la configuración del rastreo RSE, `rsecomm.properties`” en la página 53.

## Anotaciones de CARMA

- Trabajo servidor de CARMA

Al abrir una conexión con CARMA, `hlq.SCRACLST(SCRASUBMT)` iniciará un trabajo servidor (cuyo propietario será el ID del usuario) llamado `CRApuerto`, siendo puerto el número de puerto TCP/IP que se utiliza.

- DD CARMALOG

Si se especifica la sentencia DD CARMALOG en `hlq.SCRACLST(SCRASUBMT)`, las anotaciones de CARMA se redirigen a esta sentencia DD en el trabajo servidor; de lo contrario, van a SYSPRINT.

La cantidad de anotaciones viene controlada por el valor `Nivel` de rastreo en el cliente. Para obtener más detalles sobre el valor `Nivel` de rastreo, vea: Capítulo 5, “(Opcional) Activar el gestor de repositorios de acceso común (CARMA) de IBM”, en la página 57.

- DD SYSPRINT

El SYSPRINT del trabajo servidor contiene las anotaciones de CARMA, si la sentencia DD CARMALOG no está definida.

- `home/.eclipse/RSE/USERID/rsecomm.log`

Anotaciones de comunicación de CARMA, donde `home` es la vía de acceso inicial definida en el segmento OMVS del usuario (o en el segmento OMVS predeterminado, si el usuario no tiene un segmento OMVS) y `USERID` es el ID de usuario de inicio de sesión (en mayúsculas).

La cantidad de datos escritos en este archivo se controla mediante el valor `debug_level` existente en `rsecomm.properties`. Encontrará más detalles en: “(Opcional) Personalizar la configuración del rastreo RSE, `rsecomm.properties`” en la página 53.

---

## Archivos de vuelco

Cuando un producto se interrumpe de forma anómala, se crea un vuelco de almacenamiento para ayudar a determinar el problema. La disponibilidad y la ubicación de los vuelcos depende en gran medida de los valores específicos del local. Por lo que podría suceder que no se crearan o que se creen en ubicaciones distintas de las que se mencionan a continuación.

## Vuelcos de MVS

Si el programa se ejecuta en MVS, compruebe los archivos de vuelco del sistema y compruebe también el JCL de las siguientes sentencias DD (en función del producto):

- SYSABEND
- SYSMDUMP
- SYSUDUMP
- CEEDUMP
- SYSPRINT
- SYSOUT



Consulte los manuales *MVS JCL Reference (SA22-7597)* y *Language Environment Debugging Guide (GA22-7560)* para obtener más información sobre estas sentencias DD.

## Vuelcos Java

En z/OS UNIX, los vuelcos de Developer para System z están controlados por la máquina virtual Java (JVM).

La JVM crea por defecto un conjunto de agentes de vuelco durante su inicialización (SYSTDUMP y JAVADUMP). Puede alterar temporalmente este conjunto de agentes de vuelco utilizando la variable de entorno `JAVA_DUMP_OPTS`, y aún puede alterar adicionalmente el conjunto utilizando `-Xdump` en la línea de mandatos. Las opciones de línea de mandatos de la JVM están definidas en la directiva `_RSE_JAVA_OPTS` de `rsed.envvars`. No cambie ninguno de los valores, a menos que se lo indique el centro de soporte de IBM.

**Nota:** La opción `-Xdump:what` de la línea de mandatos permite determinar qué agentes de vuelco existen al realizarse el inicio.

Los tipos de vuelco que se pueden producir son:

### SYSTDUMP

Vuelco de transacciones Java. Es un vuelco de almacenamiento sin formatear generado por z/OS.

El vuelco se escribe en un conjunto de datos MVS secuencial, utilizando un nombre predeterminado con el formato `&idusuario.JVM.TDUMP.&nombretrabajo.D&fecha.T&hora`, o tal como viene determinado por el valor de la variable de entorno `JAVA_DUMP_TDUMP_PATTERN`. Si no quiere que se creen vuelcos de transacciones, añada la variable de entorno `IBM_JAVA_ZOS_TDUMP=NO` a `rsed.envvars`.

### CEEDUMP

Vuelco de Language Environment (LE). Vuelco del sistema, resumido y formateado, que muestra rastreo de la pila para cada hebra que esté en el proceso de la JVM, junto con información de registro y un vuelco de almacenamiento corto para cada registro.

El vuelco se escribe en un archivo de z/OS UNIX llamado `CEEDUMP.aaaammdd.hhmmss.pid`, donde `aaaammdd` es la fecha actual, `hhmmss` es la hora actual y `pid` es el ID del proceso actual. Las posibles ubicaciones de este archivo se describen en: “Ubicación de los vuelcos en z/OS UNIX” en la página 82.

### HEAPDUMP

Vuelco formateado (en forma de lista) de los objetos que se encuentran en la memoria dinámica Java.

El vuelco se escribe en un archivo de z/OS UNIX llamado `HEAPDUMP.aaaammdd.hhmmss.pid.TXT`, donde `aaaammdd` es la fecha actual, `hhmmss` es la hora actual y `pid` es el ID del proceso actual. Las posibles ubicaciones de este archivo se describen en: “Ubicación de los vuelcos en z/OS UNIX” en la página 82.

### JAVADUMP

Análisis formateado de la JVM. Contiene información de diagnóstico relacionada con la JVM y la aplicación Java, como el entorno de la aplicación, las hebras, la pila nativa, los bloqueos y la memoria.

El vuelco se escribe en un archivo de z/OS UNIX llamado JAVADUMP.aaaammdd.hhmmss.pid.TXT, donde aaaammdd es la fecha actual, hhmmss es la hora actual y pid es el ID del proceso actual. Las posibles ubicaciones de este archivo se describen en: "Ubicación de los vuelcos en z/OS UNIX".

Consulte el manual *Java Diagnostics Guide* (SC34-6650) si desea obtener más información sobre los vuelcos de la JVM, y el manual *Language Environment Debugging Guide* (GA22-7560) para obtener información específica de LE.

## Ubicación de los vuelcos en z/OS UNIX

La JVM comprueba cada una de las siguientes ubicaciones para ver si tienen permiso de escritura y existencia, y almacena los archivos CEEDUMP, HEAPDUMP y JAVADUMP en la primera ubicación disponible. Tenga en cuenta que debe tener suficiente espacio en disco libre para que el archivo de vuelco se escriba correctamente.

1. El directorio de la variable de entorno `_CEE_DMPTARG`, si se encuentra. Esta variable se establece en `rsed.envvars` como `home/.eclipse/RSE/USERID`, donde `home` es la vía de acceso inicial definida en el segmento OMVS del usuario (o en el segmento OMVS predeterminado, si el usuario no tiene un segmento OMVS) y `USERID` es el ID de usuario de inicio de sesión (en mayúsculas).
2. El directorio de trabajo actual, si no es el directorio raíz (`/`) y si se puede escribir en él.
3. El directorio de la variable de entorno `TMPDIR` (una variable de entorno que indica la ubicación de un directorio temporal si no es `/tmp`), si se encuentra.
4. El directorio `/tmp`.
5. Si el vuelco no se puede almacenar en ninguna de las ubicaciones anteriores, se pone en `stderr`, que es la salida de errores estándar.

---

## Autorización de control de programa para los programas RSE

El Explorador de Sistemas Remotos (RSE) es el componente de Developer para System z que proporciona servicios centrales como los de conectar el cliente al host. Se debe ejecutar en modalidad controlada por programa para poder realizar tareas como las de pasar al ID de usuario del cliente.

El bit de control de programa se establece durante la instalación de SMP/E, cuando sea necesario.

Este resultado se puede verificar con el mandato `ls -lE fekf*`, que proporciona datos de salida como los del siguiente ejemplo (\$ es el indicador de z/OS UNIX):

```
$ ls -lE /usr/lpp/wd4z/rse/lib/fekf*
-rwxr-xr-x -ps- 2 user group 94208 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfdir.dll
-rwxr-xr-x -ps- 2 user group 143360 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfdivp
-rwxr-xr-x --s- 2 user group 480 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfivpa
-rwxr-xr-x --s- 2 user group 342 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfivpc
-rwxr-xr-x --s- 2 user group 445 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfivpd
-rwxr-xr-x --s- 2 user group 1491 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfivpj
-rwxr-xr-x --s- 2 user group 883 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfivpr
-rwxr-xr-x --s- 2 user group 307 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfivps
-rwxr-xr-x -ps- 2 user group 139264 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekflock
-rwxr-xr-x -ps- 2 user group 196608 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfrsed
-rwxr-xr-x --s- 2 user group 42443 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfscmd
```

**Nota:** Para `fekfivp*` y `fekfscmd` no se necesita el atributo `+p`.

Emita los siguientes mandatos si el bit de control de programa se tiene que establecer manualmente, suponiendo que se utiliza el directorio de instalación predeterminado (/usr/lpp/wd4z/rse/lib/):

1. cd /usr/lpp/wd4z/rse/lib
2. extattr +p fekdir.dll fekfivp fekflock fekfrsed

**Nota:** Para poder utilizar el mandato **extattr +p**, debe tener como mínimo acceso de lectura (READ) al perfil BPX.FILEATTR.PROGCTL en la clase FACILITY del software de seguridad. Hallará más información en el manual *UNIX System Services Planning (GA22-7800)*.

## Puertos TCP/IP reservados

Con el mandato **NETSTAT** (TSO o z/OS UNIX) puede obtener una visión general de los puertos que se utilizan en este momento. Los datos de salida de este mandato se parecerán a los de uno de los ejemplos que siguen. Los puertos utilizados son el último número (a continuación de '..') de la columna/fila "Socket Local". Como estos puertos ya se están utilizando, no se pueden utilizar para la configuración de Developer para System z.

### IPv4

MVS ID us.	TCP/IP Conexión	NETSTAT CS V1R7 Socket Local	Nombre de TCPIP: TCPIP Socket Foráneo	TCPIP: TCPIP Estado
BPX0INIT	00000014	0.0.0.0..10007	0.0.0.0..0	Escucha
INETD4	0000004D	0.0.0.0..512	0.0.0.0..0	Escucha
INETD4	0000004B	0.0.0.0..4035	0.0.0.0..0	Escucha
JMON	00000038	0.0.0.0..6715	0.0.0.0..0	Escucha

### IPv6

MVS ID usuario	TCP/IP Conexión	NETSTAT CS V1R7 Estado	Nombre de TCPIP: TCPIP	TCPIP: TCPIP Estado
BPX0INIT	00000018	Escucha		
	Socket local:	0.0.0.0..10007		
	Socket foráneo:	0.0.0.0..0		
INETD4	00000046	Escucha		
	Socket local:	0.0.0.0..512		
	Socket foráneo:	0.0.0.0..0		
INETD4	0000004B	Escucha		
	Socket local:	0.0.0.0..4035		
	Socket foráneo:	0.0.0.0..0		
JMON	00000037	Escucha		
	Socket local:	0.0.0.0..6715		
	Socket foráneo:	0.0.0.0..0		

Otra posible limitación son los puertos TCP/IP reservados. Hay dos lugares comunes en los que se reservan puertos TCP/IP:

- **PROFILE.TCPIP**

Este es el conjunto de datos al que hace referencia la sentencia DD PROFILE de la tarea iniciada TCP/IP, que a menudo se llama SYS1.TCPPARMS(TCPPROF).

- PORT: Reserva un puerto para los nombres de trabajo especificados.
- PORTRANGE: Reserva un rango de puertos para los nombres de trabajo especificados.

Para obtener más información sobre estas sentencias, consulte el manual *Communications Server IP Configuration Guide (SC31-8775)*.

- **SYS1.PARMLIB(BPXPRMxx)**

- INADDRANYPORT: Especifica el número de puerto inicial del rango de números de puerto que el sistema reserva para utilizar con los enlaces PORT 0, INADDR\_ANY. Este valor solo se necesita para CINET.
- INADDRANYCOUNT: Especifica el número de puertos que el sistema reserva, empezando por el número de puerto especificado en el parámetro INADDRANYPORT. Este valor solo se necesita para CINET.

Para obtener más información sobre estas sentencias, consulte los manuales *UNIX System Services Planning (GA22-7800)* y *MVS Initialization and Tuning Reference (SA22-7592)*.

Para obtener una lista de estos puertos reservados, se puede utilizar el mandato **NETSTAT PORTL** (TSO o z/OS UNIX), que crea una salida parecida a la de este ejemplo:

NºPto	Prot	Usuario	Dstivos	Rango	Nombre de TCPIP: Dirección IP	17:08:32
00007	TCP	MISCSERV	DA			
00009	TCP	MISCSERV	DA			
00019	TCP	MISCSERV	DA			
00020	TCP	OMVS	D			
00021	TCP	FTPD1	DA			
00025	TCP	SMTP	DA			
00053	TCP	NAMESRV	DA			
00080	TCP	OMVS	DA			
03500	TCP	OMVS	DAR	03500-03519		
03501	TCP	OMVS	DAR	03500-03519		

En el manual *Communications Server IP System Administrator's Commands (SC31-8781)* hallará más información sobre el mandato **NETSTAT**.

**Nota:** El mandato **NETSTAT** solo muestra la información definida en `PROFILE.TCPIP`, que debe solapar las definiciones de `BPXPRMxx`. En caso de duda o problemas, compruebe el miembro `parmlib BPXPRMxx` para verificar los puertos que se reservan aquí.

**Nota:** Si tiene problemas relacionados con el enlace `INETD` con los puertos reservados, lea: “Definiciones de puertos en `PROFILE.TCPIP`” en la página 104.

---

## Tamaño del espacio de direcciones

Para el servidor RSE, que es un proceso Java UNIX, se necesita una región de gran tamaño para efectuar sus funciones. Por lo tanto, es importante establecer límites de almacenamiento grandes para los espacios de direcciones de OMVS.

### Requisitos para INETD

`INETD` inicia un servidor RSE cuando un cliente se conecta al puerto del daemon RSE o cuando el cliente emite el script de arranque utilizando `REXEC/SSH`. Esto se hace empleando los límites de almacenamiento de `INETD`, por lo que estos se tienen que establecer con el suficiente tamaño.

- Si `INETD` del proceso servidor UNIX se inicia mediante el JCL utilizando `BPXBATCH`, el tamaño de la región debe ser 0.
- Si `INETD` se inicia desde una sesión TSO/OMVS, el tamaño de la región TSO debe ser igual o mayor que 2096128.

## Limitaciones establecidas en SYS1.PARMLIB(BPXPRMxx)

Establezca que MAXASSIZE, en SYS1.PARMLIB(BPXPRMxx) (que define el tamaño de región (proceso) de espacio de direcciones OMVS predeterminado), sea igual o mayor que 2147483647.

Este valor se puede comprobar y establecer dinámicamente (hasta la próxima IPL) con los siguientes mandatos de consola, como se describe en el manual *MVS System Commands* (SA22-7627):

1. DISPLAY OMVS,0
2. SETOMVS MAXASSIZE=2147483647

## Limitaciones almacenadas en el perfil de seguridad

Compruebe ASSIZEMAX, en el segmento OMVS del ID de usuario del cliente, y establezca que sea igual a 2147483647 o, preferiblemente, igual a NONE para que utilice el valor SYS1.PARMLIB(BPXPRMxx).

Con RACF, este valor se puede comprobar y establecer con los siguientes mandatos TSO, como se describe en el manual *Security Server RACF Command Language Reference* (SA22-7687):

1. LISTUSER userid NORACF OMVS
2. ALTUSER userid OMVS(NOASSIZEMAX)

## Limitaciones puestas en vigor por la rutinas de salida del sistema

Asegúrese de que no permite que las rutinas de salida IEFUSI o IEALIMIT del sistema controlen los tamaños de las regiones del espacio de direcciones de OMVS. Una manera posible de lograrlo es escribiendo SUBSYS(OMVS,NOEXITS) en el código de SYS1.PARMLIB(SMFPRMxx).

Los valores de SYS1.PARMLIB(SMFPRMxx) se pueden comprobar y activar con los siguientes mandatos de consola, como se describe en el manual *MVS System Commands* (SA22-7627):

1. DISPLAY SMF,0
2. SET SMF=xx

---

## Rastreo de información de retorno de error

El siguiente procedimiento permite reunir la información necesaria para diagnosticar problemas de información de retorno de errores con procedimientos de construcción remotos. Este rastreo afectará negativamente al rendimiento y solo se debe activar cuando así lo indica el centro de soporte de IBM. En este apartado, todas las referencias a hlq se refieren al calificador de alto nivel empleado durante la instalación de Developer para System z. El valor predeterminado de la instalación es FEK, pero quizá no sea válido para su local.

1. Haga una copia de seguridad del procedimiento de compilación ELAXFC0C activo. Este procedimiento viene por defecto en el conjunto de datos hlq.SFEKSAMP, pero es posible que se haya copiado en una ubicación distinta, como SYS1.PROCLIB, según se describe en: "Personalizar procedimientos de construcción remota ELAXF\*" en la página 22.
2. Cambie el procedimiento ELAXFC0C activo para que incluya la serie 'MAXTRACE' en la opción de compilación EXIT(ADEXIT(ELAXMGUX)).

```
//COBOL EXEC PGM=IGYCRCTL,REGION=2048K,
//*      PARM=('EXIT(ADEXIT(ELAXMGUX))'),
//      PARM=('EXIT(ADEXIT('MAXTRACE',ELAXMGUX))',
//      'ADATA',
//      'LIB',
//      'TEST(NONE,SYM,SEP)',
//      'LIST',
//      'FLAG(I,I)'&CICS &DB2 &COMP)
```

**Nota:** Tendrá que duplicar los apóstrofes en torno a MAXTRACE. Ahora, la opción es: EXIT(ADEXIT('MAXTRACE',ELAXMGUX))

3. Realice una comprobación de sintaxis remota en un programa COBOL. Developer para System z viene como programa COBOL de ejemplo en hlq.SFEKSAMP(ADNSMSGH).
4. El componente SYSOUT de la salida de JES empezará enumerando los nombres de los conjuntos de datos de SIDEFILE1, SIDEFILE2, SIDEFILE3 y SIDEFILE4.

```
ABOUT TOO OPEN SIDEFILE1 - NAME = 'uid.DT021207.TT110823.M0000045.C0000000'
SUCCESSFUL OPEN SIDEFILE1 - NAME = 'uid.DT021207.TT110823.M0000045.C0000000'
ABOUT TOO OPEN SIDEFILE2 - NAME = 'uid.DT021207.TT110823.M0000111.C0000001'
SUCCESSFUL OPEN SIDEFILE2 - NAME = 'uid.DT021207.TT110823.M0000111.C0000001'
ABOUT TOO OPEN SIDEFILE3 - NAME = 'uid.DT021207.TT110823.M0000174.C0000002'
SUCCESSFUL OPEN SIDEFILE3 - NAME = 'uid.DT021207.TT110823.M0000174.C0000002'
ABOUT TOO OPEN SIDEFILE4 - NAME = 'uid.DT021207.TT110823.M0000236.C0000003'
SUCCESSFUL OPEN SIDEFILE4 - NAME = 'uid.DT021207.TT110823.M0000236.C0000003'
```

**Nota:** En función de los valores que tenga, SIDEFILE1 y SIDEFILE2 podrían señalar hacia una sentencia DD (SUCCESSFUL OPEN SIDEFILE1 - NAME = DD:WSEDSF1). Consulte el componente JESJCL de la salida (situado antes del componente SYSOUT) para obtener el nombre del conjunto de datos real.

```
22 //COBOL.WSEDSF1 DD DISP=MOD,
// DSN=uid.ERRCOB.member.SF1.Z682746.XML
23 //COBOL.WSEDSF2 DD DISP=MOD,
// DSN=uid.ERRCOB.member.SF1.Z682747.XML
```

5. Copie estos cuatro conjuntos de datos en el PC, creando por ejemplo un proyecto COBOL local en Developer para System z y añadiendo los conjuntos de datos SIDEFILE1->4.
6. Copie las anotaciones de trabajo JES completas en el PC, abriendo por ejemplo la salida de trabajo en Developer para System z y guardándola en el proyecto local, seleccionado **Archivo > Guardar como...**
7. Restaure el procedimiento ELAXFCOC a su estado original, ya sea deshaciendo el cambio (elimine la serie "MAXTRACE" de las opciones de compilación) o restaurando la copia de seguridad.
8. Envíe los archivos recogidos (SIDEFILE1->4 y anotaciones de trabajo) al centro de soporte de IBM.

---

## Transacción APPC y el servicio de mandatos TSO

Si no puede usar la versión APPC del servicio de mandatos TSO, las áreas en las que pueden surgir problemas son dos: al iniciar la transacción del servidor APPC y al conectar con RSE.

- Si no ve los mensajes que indican cómo instalar APPC, mire a ver si en las anotaciones del sistema hay mensajes de RACF (el ID de los mensajes es ICHxxxxx) u otros mensajes relacionados con el mandato emitido o con el ID de usuario que lo emitió. Las causas comunes de este tipo de problemas son:
  - No posee autorización de lectura sobre el conjunto de datos hlq.SFEKPROC.



- TCP/IP no está activo, tiene conectado un nombre DNS no válido o no se puede acceder al sistema (no responde al mandato ping) debido a problemas de red, a una dirección IP incorrecta o a otras causas.
- Si ve mensajes sobre cómo instalar APPC, pero no ve el mensaje que confirma que la instalación ha sido satisfactoria, es probable que la transacción del servidor APPC no haya podido iniciarse. Consulte las anotaciones de error de la transacción (userid.FEKFRSRV.&TPDATE.&TPTIME.LOG ). Algunas de las causas probables de los problemas son:
  - La pila TCP/IP no utiliza el nombre predeterminado de TCPIP, y la tarjeta DD SYSTCPD no se ha establecido o señala hacia un conjunto de datos equivocado.
  - El servidor no ha podido asignar SYSPROC o SYSTSPRT.
  - El JCL señala hacia un SYSPROC equivocado (SYSPROC debe incluir hlq.SFEKPROC).
  - El servidor no ha podido abrir o acceder al conjunto de datos de mensajes (anotaciones) al que hace referencia MESSAGE\_DATA\_SET.
  - No hay suficientes iniciadores disponibles para el planificador APPC.
  - Los espacios de direcciones APPC o ASCH no están activos.
  - La clase utilizada (que se llama "A" por defecto) no está definida ante el planificador APPC ASCH.
  - No hay ningún segmento OMVS predeterminado para el sistema, y el usuario no tiene un segmento OMVS personal, o hay un error de definición en alguno de ellos.
  - El grupo predeterminado del segmento OMVS predeterminado o el grupo predeterminado del usuario no tienen un número de GID.

El REXX proporcionado en "(Opcional) Definir una transacción APPC para el servicio de mandatos TSO" en la página 24 puede ayudarle a resolver problemas de APPC, porque le da la posibilidad de gestionar APPC interactivamente a través de los paneles ISPF. Sin embargo, tenga en cuenta que puede desactivar la transacción con esta herramienta; la transacción sigue ahí, pero no aceptará conexiones.

La lista que sigue es una selección de fichas técnicas disponibles actualmente en el sitio Web de soporte (<http://www-306.ibm.com/software/awdtools/devzseries/support/>). Si desea información adicional, la encontrará en el sitio Web de soporte:

- La verificación de APPC falla con el código de retorno 2016 - EHOSTNOTFOUND
- La verificación de APPC falla con el código de retorno 1004 - EIBMIUCVERR
- La verificación de APPC falla con el código de retorno 9 - Nombre de TP no reconocido
- La verificación de APPC falla con el código de retorno 10 - TP no disponible No se reintentará
- La verificación de APPC falla con el código de retorno 19 - Error de parámetro
- La verificación de APPC falla con el código de retorno 20 - Error específico del producto
- La verificación de APPC falla con el código de retorno 26 - Anomalía de recurso
- CEE3501S: No se ha encontrado el módulo IOSTREAM
- El servidor no se ha podido iniciar: EDC5129I No existe ese archivo o directorio
- exec/tcp: bind: EDC5111I Permiso denegado, rsn=744C7246
- No se obtiene respuesta del servidor, con uno de estos mensajes

- IEA995I SALIDA VUELCO SÍNTOMAS 473 CÓDIGO CÓDIGO TERMINACIÓN USUARIO=4093 CÓDIGO RAZÓN=0000001C (en ANOTACIONES SDSF)
- CEE3512S Una carga HFS del módulo libicudata32.0.dll ha fallado. El código de retorno del sistema era 0000000157; el código de razón era 0BDF019B. (En CEEDUMP)
- No se ha podido obtener espacio (en cliente .log)
- El mandato C\_CONNECT no está disponible
- Mensaje de error "La inicialización del servidor FFS ha fallado" al conectar con el host
- "EDC5139I Operación no permitida" al conectar con el host
- "RSEG1056U La inicialización del servidor FFS ha fallado" al abrir un archivo MVS

**Nota:** La lista no es definitiva, conviene que consulte el sitio Web de soporte por si hay fichas técnicas adicionales.

---

## Información miscelánea

### Límites del sistema

SYS1.PARMLIB(BPXPRMxx) define muchas limitaciones relacionadas con z/OS UNIX, que se podrían alcanzar cuando hay varios clientes Developer para System z activos. La mayoría de los valores BPXPRMxx se pueden cambiar dinámicamente con los mandatos de consola **SETOMVS** y **SET OMVS**.

#### Conexión rehusada

Cada conexión RSE inicia varios procesos que están permanentemente activos. Se pueden rehusar nuevas conexiones debido al límite establecido en SYS1.PARMLIB(BPXPRMxx) sobre la cantidad de procesos, especialmente cuando los usuarios comparten un mismo UID (como cuando se utiliza el segmento OMVS predeterminado).

- El límite por cada UID se establece mediante la palabra clave MAXPROCUSER, y su valor predeterminado es 25.
- El límite a escala del sistema se establece mediante la palabra clave MAXPROCSYS, y su valor predeterminado es 200.

Otra fuente de conexiones rehusadas es el límite de la cantidad de espacios de direcciones z/OS activas y de usuarios de z/OS UNIX activos.

- La cantidad máxima de ID de espacios de direcciones (ASID) se define en SYS1.PARMLIB(IEASYSxx) con la palabra clave MAXUSER, y su valor predeterminado es 255.
- La cantidad máxima de ID de usuarios (UID) de z/OS UNIX se define en SYS1.PARMLIB(BPXPRMxx) con la palabra clave MAXUIDS, y su valor predeterminado es 200.

## Problemas conocidos

### No se pueden abrir conjuntos de datos MVS

Para leer y escribir un conjunto de datos MVS, se necesita un dominio del sistema de archivos físico por sockets. Si el sistema de archivos no está debidamente



definido o si no tiene suficientes sockets, el gestor de bloqueos (FFS) podría no satisfacer las peticiones de lectura/escritura. Los archivos `ffs*.log` mostrarán mensajes como los siguientes:

- Error 127 al obtener par de sockets - el puerto se establecerá en 0.
- No se puede crear socket en el dominio UNIX. El error es: "La familia de direcciones no está soportada"

Verifique que el miembro `SYS1.PARMLIB(BPXPRMxx)` contiene las siguientes sentencias:

```
FILESYSTYPE TYPE(UDS) ENTRYPPOINT(BPXTUINT)
NETWORK DOMAINNAME(AF_UNIX)
        DOMAINNUMBER(1)
        MAXSOCKETS(200)
        TYPE(UDS)
```

### Los enlaces DVIPA fallan

Cuando se utiliza VIPA (dirección IP virtual) dinámica a través de múltiples pilas de TCPIP, se necesita una coordinación a escala del sistema de los puertos efímeros, para que la tupla 4 (combinación de direcciones IP y puertos origen y destino) de cada conexión permanezca exclusiva. Esto se puede hacer añadiendo el parámetro `SYSPLEXPORTS` opcional a la primera sentencia `VIPADISTRIBUTE`, como se describe en el manual *Communications Server IP Configuration Guide* (SC31-8775).

Cuando utilice esta técnica, asegúrese de que se ha definido la estructura del recurso de acoplamiento `EZBEPOR` (que contiene información de asignación de puertos sysplex. En el manual *Communications Server SNA Network Implementation Guide* (SC31-8777) encontrará información sobre cómo llevarlo a la práctica.

## Emulador de Host Connect

- El emulador de Host Connect utiliza telnet TN3270 (no el servidor RSE) para establecer conexión con el host.
- Cuando se utiliza telnet seguro (SSL) y se trabaja con certificados no firmados por una CA de todos conocida, cada cliente debe añadir el certificado de la CA a su lista de CA de confianza del emulador de Host Connect.
- La opción `NOSNAEXT` de `TELNETPARMS` de TCP/IP podría ser necesaria para inhabilitar las extensiones funcionales de SNA. Si se especifica `NOSNAEXT`, el servidor telnet TN3270 no negocia las funciones de resolución de contiendas y detección de SNA.

---

## Ponerse en contacto con el soporte de IBM

Después de haber leído este manual y de consultar el sitio Web de soporte (<http://www-306.ibm.com/software/awdtools/devzseries/support/>), si todavía tiene dificultades y necesita ayuda del personal de soporte de IBM, reúna los siguientes datos y abra un registro de problemas con IBM:

- Su nivel de software actual, documentado en:
  - `hlq.SFEKJCL(FEK#LVL#)` para Developer para System z. El valor predeterminado de `hlq` es FEK.
  - `hlq.SCRAJCL(CRA#LVL#)` para CARMA. El valor predeterminado de `hlq` es CRA.

La siguiente información le ayudará a resolver los problemas de conexión

- La salida de los siguientes mandatos de consola:
  - `D OMVS,0`

- La salida de los siguientes mandatos TSO:
  - HOMETEST
  - LISTUSER userid NORACF OMVS
  - LISTGROUP group NORACF OMVS
- Archivo rsed.envvars, situado por defecto en /usr/lpp/wd4z/rse/lib/, pero cuya ubicación personalizada conviene que sea en /etc/wd4z/
- Archivo /etc/inetd.conf
- Todos los archivos \*.log situados en home/.eclipse/RSE/USERID/, donde home es la vía de acceso inicial definida en el segmento OMVS del usuario (o en el segmento OMVS predeterminando si el usuario no tiene un segmento OMVS), y USERID es el ID de usuario de inicio de sesión (en mayúsculas).
- Mensajes de SYSLOG relacionados
- Complete la salida de las pruebas del IVP relacionado, como se describe en: “Verificar la instalación del servidor RSE” en la página 46.
  - netstat : “Disponibilidad de los puertos” en la página 47
  - fekfivps : “Script de shell REXEC/SSH” en la página 49
  - fekfivpr : “Conexión REXEC” en la página 48
  - fekfivpd : “Conexión del daemon RSE” en la página 50.
  - fekfivpj : “Conexión del supervisor de trabajos JES” en la página 50.
  - fekfivpc : “Conexión del servicio de mandatos TSO (utilizando SCLMDT)” en la página 50
  - fekfivpa : “Conexión del servicio de mandatos TSO (utilizando APPC)” en la página 52

Si utiliza SCLM Developer Toolkit para el servicio de mandatos TSO (método predeterminado)

- Complete la salida de la prueba del IVP SCLMDT (fekfivpc), como se describe en: “Conexión del servicio de mandatos TSO (utilizando SCLMDT)” en la página 50
- Archivo /etc/SCLMDT/CONFIG/ISPF.conf

Si utiliza APPC para el servicio de mandatos TSO

- Complete la salida de la prueba del IVP APPC (fekfivpa), como se describe en: “Conexión del servicio de mandatos TSO (utilizando APPC)” en la página 52
- El JCL que define la transacción del servidor APPC, que es hlq.SFEKSAMP(FEKAPPCC) si utiliza el trabajo de ejemplo
- Miembro SYS1.PARMLIB(APPCPMxx)
- Miembro SYS1.PARMLIB(ASCHPMxx)
- Conjunto de datos userid.FEKFSERV.&TPDATE.&TPTIME.LOG
- La salida de los siguientes mandatos de consola:
  - D APPC,TP,ALL
  - D APPC,LU,ALL
  - D ASCH,ALL

Proporcione toda la información que parezca relevante para los problemas funcionales, como

- Anotaciones de trabajo
- Mensajes de SYSLOG relacionados

- Archivos de anotaciones situados en `home/.eclipse/RSE/USERID/`, donde `home` es la vía de acceso inicial definida en el segmento OMVS del usuario (o en el segmento OMVS predeterminado, si el usuario no tiene un segmento OMVS) y `USERID` es el ID de usuario de inicio de sesión (en mayúsculas).



---

## Apéndice C. Configurar TCP/IP

Este apéndice se propone ayudarle a resolver algunos problemas comunes que pueden surgir al configurar TCP/IP, o durante la tarea de comprobar y/o modificar una configuración existente.

Consulte los manuales *Communications Server: IP Configuration Guide* (SC31-8775) y *Communications Server IP Configuration Reference* (SC31-8776) para obtener información adicional sobre la configuración de TCP/IP.

---

### Dependencia del nombre de host

Developer para System z depende de que TCP/IP tenga el nombre de host correcto en el momento de la inicialización. Ello implica que los distintos archivos de configuración de TCP/IP y del resolvente estén configurados correctamente.

Puede probar la configuración de TCP/IP con el mandato TSO **HOMETEST**. En el manual *Communications Server IP System Administrator's Commands* (SC31-8781) hallará más información sobre el mandato **HOMETEST**.

Salida de ejemplo del mandato **HOMETEST**:

```
Ejecutar IBM MVS TCP/IP CS V1R7 TCP/IP Configuration Tester
```

```
El archivo de parámetros de configuración de FTP utilizado será "SYS1.TCPPARMS(FTPDATA)"
```

```
El nombre de host TCP es: CDFMVS08
```

```
Utilizar servidor de nombres para resolver CDFMVS08
```

```
Las siguientes direcciones IP corresponden al nombre de host TCP: CDFMVS08  
9.42.112.75
```

```
Las siguientes direcciones IP son las direcciones IP HOME definidas en PROFILE.TCPIP:  
9.42.112.75  
127.0.0.1
```

```
iTodas las direcciones IP de CDFMVS08 están en la lista HOME!
```

```
La prueba de Home ha sido satisfactoria – iSe han superado todas las pruebas!
```

---

### Qué son los resolventes

El resolvente funciona en nombre de los programas como un cliente que accede a los servidores de nombres para obtener una resolución de nombre en dirección o una resolución de dirección en nombre. Para resolver la consulta del programa peticionario, el resolvente puede acceder a los servidores de nombres disponibles, utilizar definiciones locales (por ejemplo, `/etc/resolv.conf`, `/etc/hosts`, `/etc/ipnodes`, `HOSTS.SITEINFO`, `HOSTS.ADDRINFO` o `ETC.IPNODES`), o utilizar una combinación de ambos.

El resolvente, al iniciarse su espacio de direcciones, lee un conjunto de datos de instalación opcional del resolvente hacia el que señala la tarjeta DD SETUP en el procedimiento del JCL del resolvente. Si no se proporciona información de instalación, el resolvente utiliza el orden de búsqueda nativo de MVS o z/OS UNIX aplicable sin ninguna información de `GLOBALTCPIPDATA`, `DEFAULTTCPIPDATA`, `GLOBALIPNODES`, `DEFAULTIPNODES` o `COMMONSEARCH`.

**Nota:** Si no hay ningún procedimiento JCL de resolvente en el sistema proclib, el espacio de direcciones se iniciará utilizando los valores predeterminados del sistema (sin DD SETUP).

---

## Qué es el orden de búsqueda de la información de configuración

Es importante comprender el orden de búsqueda de archivos de configuración que las funciones de TCP/IP utilizan, y conviene saber cuándo se puede alterar temporalmente el orden de búsqueda predeterminado con las variables de entorno, con el JCL o con otras variables que proporcione. Este conocimiento le permite acomodar sus estándares de denominación de los conjuntos de datos y archivos de HFS locales, y también le resultará de utilidad conocer el conjunto de datos o archivo de HFS de configuración al diagnosticar problemas.

Otro punto importante a tener en cuenta es que cuando se aplica un orden de búsqueda para cualquier archivo de configuración, la búsqueda finaliza con el primer archivo que se encuentre. Por lo tanto, es posible obtener resultados inesperados si coloca información de configuración en un archivo que nunca se va a encontrar, ya sea porque existen otros archivos antes según el orden de la búsqueda, o porque el archivo no está incluido en el orden de búsqueda elegido por la aplicación.

Al buscar archivos de configuración, puede indicar explícitamente a TCP/IP dónde está la mayoría de los archivos de configuración, utilizando para ello sentencias DD en los procedimientos del JCL o estableciendo variables de entorno. Por otro lado, puede dejar que sea TCP/IP el que determine dinámicamente la ubicación de los archivos de configuración, basándose en el orden de búsqueda documentado en el manual *Communications Server IP Configuration Guide* (SC31-8775).

El componente de configuración de la pila de TCP/IP utiliza TCPIP.DATA durante la inicialización de la pila de TCP/IP para determinar el nombre de host (HOSTNAME) de la pila. Para obtener este valor, se utiliza el orden de búsqueda del entorno z/OS UNIX.

**Nota:** Utilice el recurso de resolvente de rastreo para determinar qué valores de TCPIP.DATA utiliza el resolvente y dónde se han leído. Para obtener información sobre cómo iniciar dinámicamente el rastreo, consulte el manual *Communications Server IP Diagnosis Guide* (GC31-8782). Una vez que el rastreo esté activo, emita un mandato TSO **NETSTAT HOME** y un mandato de shell z/OS UNIX **netstat -h** para visualizar los valores. Si se emite un mandato PING de un nombre de host desde TSO y desde la shell z/OS UNIX, también se muestra la actividad de los servidores DNS que podrían estar configurados.

---

## Orden de búsqueda utilizado en el entorno z/OS UNIX

El archivo o tabla concreto que se busca puede ser un conjunto de datos MVS o un archivo HFS, en función de los valores de configuración del resolvente y de la presencia de determinados archivos en el sistema.

### Archivos de configuración de resolvente base:

El archivo de configuración de resolvente base contiene sentencias TCPIP.DATA. Además de las directivas del resolvente, se le hace referencia para determinar, entre otras cosas, el prefijo de conjunto de datos (valor de la sentencia

DATASETPREFIX) que hay que utilizar al intentar acceder a algunos de los archivos de configuración especificados en esta sección.

El orden de búsqueda empleado para acceder al archivo de configuración de resolvente base es el siguiente:

1. **GLOBALTCPIPDATA**

Si está definido, se utiliza el valor de la sentencia de configuración GLOBALTCPIPDATA del resolvente (vea también: “Qué son los resolventes” en la página 93). La búsqueda continúa hasta encontrar un archivo de configuración adicional. La búsqueda finaliza con el próximo archivo encontrado.

2. El valor de la variable de entorno **RESOLVER\_CONFIG**

Se utiliza el valor de la variable de entorno. Esta búsqueda fallará si el archivo no existe o si se ha asignado de manera exclusiva en otra parte.

3. **/etc/resolv.conf**

4. Tarjeta **DD //SYSTCPD**

Se utiliza el conjunto de datos asignado a la DD de nombre SYSTCPD. En el entorno z/OS UNIX, un proceso hijo no tiene acceso a la DD SYSTCPD. Ello se debe a que la asignación de SYSTCPD no se hereda del proceso padre por las llamadas a fork() o a la función exec.

5. **userid.TCPIP.DATA**

userid es el ID de usuario asociado al entorno de seguridad actual (espacio de direcciones o tarea/hebra).

6. **jobname.TCPIP.DATA**

jobname es el nombre especificado en la sentencia JCL de JOB para los trabajos por lotes o el nombre de un procedimiento iniciado.

7. **SYS1.TCPPARMS(TCPDATA)**

8. **DEFAULTTCPIPDATA**

Si está definido, se utiliza el valor de la sentencia de configuración DEFAULTTCPIPDATA del resolvente (vea también: “Qué son los resolventes” en la página 93).

9. **TCPIP.TCPIP.DATA**

## Tablas de conversión:

A las tablas de conversión (de EBCDIC a ASCII y de ASCII a EBCDIC) se les hace referencia para determinar los conjuntos de datos de conversión que hay que utilizar. El orden de búsqueda empleado para acceder a este archivo de configuración es el siguiente. La búsqueda finaliza en el primer archivo encontrado:

1. El valor de la variable de entorno **X\_XLATE** El valor de variable de entorno es el nombre de la tabla de conversión producida por el mandato TSO CONVXLAT.

2. **userid.STANDARD.TCPXLBIN**

userid es el ID de usuario asociado al entorno de seguridad actual (espacio de direcciones o tarea/hebra).

3. **jobname.STANDARD.TCPXLBIN**

jobname es el nombre especificado en la sentencia JCL de JOB para los trabajos por lotes o el nombre de un procedimiento iniciado.

4. **hlq.STANDARD.TCPXLBIN**

hlq representa el valor de la sentencia DATASETPREFIX especificada en el archivo de configuración de resolvente base (si se da con él); en caso contrario, hlq es TCPIP por defecto.

5. Si no se encuentra ninguna tabla, el resolvente emplea una tabla predeterminada codificada por programa, idéntica a la tabla que figura en el miembro de conjunto de datos SEZATCPX(STANDARD).

## Tablas de hosts locales:

Por defecto, en primer lugar el resolvente intenta utilizar los servidores de nombres de dominio que estén configurados para las peticiones de resolución. Si la petición de resolución no se puede satisfacer, se emplean las tablas de hosts locales. El comportamiento del resolvente se controla mediante las sentencias TCPIP.DATA.

Las sentencias TCPIP.DATA del resolvente definen si hay que utilizar los servidores de nombres de dominio y cómo hay que hacerlo. La sentencia LOOKUP TCPIP.DATA también puede servir para controlar cómo se utilizan los servidores de nombres de dominio y las tablas de hosts locales. Para obtener más información sobre las sentencias TCPIP.DATA, consulte el manual *Communications Server IP Configuration Reference* (SC31-8776).

El resolvente emplea el orden de búsqueda exclusivo de Ipv4 para obtener información de nombres de locales incondicionalmente para las llamadas a la API getnetbyname. El orden de búsqueda exclusivo de Ipv4 para obtener información de nombres de locales es el siguiente. La búsqueda finaliza en el primer archivo encontrado:

1. El valor de la variable de entorno **X\_SITE**  
El valor de la variable de entorno es el nombre del archivo de información HOSTS.SITEINFO creado por el mandato TSO **MAKESITE**.
2. El valor de la variable de entorno **X\_ADDR**  
El valor de la variable de entorno es el nombre del archivo de información HOSTS.ADDRINFO creado por el mandato TSO **MAKESITE**.
3. **/etc/hosts**
4. **userid.HOSTS.SITEINFO**  
userid es el ID de usuario asociado al entorno de seguridad actual (espacio de direcciones o tarea/hebra).
5. **jobname.HOSTS.SITEINFO**  
jobname es el nombre especificado en la sentencia JCL de JOB para los trabajos por lotes o el nombre de un procedimiento iniciado.
6. **hlq.HOSTS.SITEINFO**  
hlq representa el valor de la sentencia DATASETPREFIX especificada en el archivo de configuración de resolvente base (si se da con él); en caso contrario, hlq es TCPIP por defecto.

---

## Cómo se aplica esto a Developer para System z

Tal como se ha indicado antes, Developer para System z depende de que TCP/IP tenga el nombre de host correcto en el momento de la inicialización. Ello implica que los distintos archivos de configuración de TCP/IP y del resolvente estén configurados correctamente.

En el ejemplo que sigue, nos centraremos en algunas tareas de configuración de TCP/IP y del resolvente. Fíjese en que no se trata de describir una configuración



completa de TCP/IP o del resolvente, sino tan solo de resaltar algunos aspectos clave que podrían ser válidos para su local.

1. En el siguiente JCL vemos que TCP/IP empleará SYS1.TCPPARMS(TCPDATA) para determinar el nombre de host de la pila.

```
//TCPIP    PROC  PARMS='CTRACE(CTIEZB00)',PROF=TCPPROF,DATA=TCPDATA
//*
//* RED TCP/IP
//*
//TCPIP    EXEC  PGM=EZBTCP,REGION=0M,TIME=1440,PARM=&PARMS
//PROFILE  DD   DISP=SHR,DSN=SYS1.TCPPARMS(&PROF)
//SYSTCPD  DD   DISP=SHR,DSN=SYS1.TCPPARMS(&DATA)
//SYSPRINT DD   SYSOUT=*,DCB=(RECFM=VB,LRECL=132,BLKSIZE=136)
//ALGPRINT DD   SYSOUT=*,DCB=(RECFM=VB,LRECL=132,BLKSIZE=136)
//CFGPRINT DD   SYSOUT=*,DCB=(RECFM=VB,LRECL=132,BLKSIZE=136)
//SYSOUT   DD   SYSOUT=*,DCB=(RECFM=VB,LRECL=132,BLKSIZE=136)
//CEEDUMP  DD   SYSOUT=*,DCB=(RECFM=VB,LRECL=132,BLKSIZE=136)
//SYSERROR DD   SYSOUT=*
```

2. SYS1.TCPPARMS(TCPDATA) nos indica que queremos que el nombre del sistema sea el nombre de host y que no utilizamos un servidor de nombres de dominio (DNS); todos los nombres se resolverán por medio de la búsqueda en la tabla de locales.

```
; HOSTNAME especifica el nombre de host TCP de este sistema. Si no
; se especifica, el HOSTNAME predeterminado será el nombre de nodo especificado
; en el miembro IEFSSNxx PARMLIB.
;
; HOSTNAME
;
; DOMAINORIGIN especifica el origen de dominio que se añadirá
; a los nombres de host que se pasen al resolvente. Si un nombre de host contiene
; puntos, el DOMAINORIGIN no se añadirá al final del
; nombre de host.
;
DOMAINORIGIN  RALEIGH.IBM.COM
;
; NSINTERADDR especifica la dirección IP del servidor de nombres.
; LOOPBACK (14.0.0.0) especifica el servidor de nombres local. Si
; no se va a emplear un servidor de nombres, no codifique una sentencia NSINTERADDR.
; (Descomente la siguiente línea NSINTERADDR). Esto hará que todos los nombres
; se resuelvan por medio de la búsqueda de la tabla de locales.
;
; NSINTERADDR 14.0.0.0
;
; TRACE RESOLVER provocará un rastreo completo de todas las consultas y
; respuestas del servidor de nombres o de las tablas de locales, que se escribirá en
; la consola del usuario. Este mandato solo tiene como finalidad la depuración.
;
; TRACE RESOLVER
```

3. En el JCL del resolvente vemos que no se utilizar la sentencia DD SETUP. Como ya se mencionó en: “Qué son los resolventes” en la página 93, esto quiere decir que no se empleará la variable GLOBALTCPIPDATA ni tampoco otras variables.

```
//RESOLVER PROC  PARMS='CTRACE(CTIRES00)'
//*
//* IP NAME RESOLVER – START WITH SUB=MSTR
//*
//RESOLVER EXEC  PGM=EZBREINI,REGION=0M,TIME=1440,PARM=&PARMS
//*SETUP    DD   DISP=SHR,DSN=USER.PROCLIB(RESSETUP),FREE=CLOSE
```

4. Si damos por sentado que la variable de entorno RESOLVER\_CONFIG no está establecida, podemos ver en la Tabla 13 en la página 98 que el resolvente intentará utilizar /etc/resolv.conf como archivo de configuración base.

```

TCPIPJOBNAME TCPIP
DomainOrigin RALEIGH.IBM.COM
HostName CDFMVS08

```

Como ya se ha mencionado en: “Orden de búsqueda utilizado en el entorno z/OS UNIX” en la página 94, el archivo de configuración base contiene sentencias TCPIP.DATA. Si el nombre del sistema es CDFMVS08 (TCPDATA indicaba que se utiliza el nombre del sistema como nombre de host), podemos ver que /etc/resolv.conf está en sincronización con SYS1.TCPPARMS(TCPDATA). No hay definiciones de DNS y, por lo tanto, se utilizará la búsqueda de la tabla de locales.

5. La Tabla 13 también nos indica que si no tenemos nada que hacer para utilizar la tabla de conversión ASCII-EBCDIC predeterminada.
6. Suponiendo que no se utiliza el mandato TSO **MAKESITE** (puede crear las variables X\_SITE y X\_ADDR), /etc/hosts será la tabla de locales empleada para la búsqueda de nombres.

```

# Resolvente /etc/hosts file cdfmvs08
9.42.112.75 cdfmvs08 # Host CDFMVS08
9.42.112.75 cdfmvs08.raleigh.ibm.com # Host CDFMVS08
127.0.0.1 localhost

```

El contenido mínimo de este archivo es información sobre el sistema actual. En el ejemplo anterior definimos cdfmvs08 y cdfmvs08.raleigh.ibm.com como un nombre válido de una dirección IP de nuestro sistema z/OS.

Si utilizásemos un servidor de nombres de dominio (DNS), el DNS contendría la información de /etc/hosts, y /etc/resolv.conf y SYS1.TCPPARMS(TCPDATA) tendrían sentencias que identificarían el DNS ante nuestro sistema.

Para evitar confusiones, conviene mantener los archivos de configuración de TCP/IP y del resolvente sincronizados entre sí.

*Tabla 13. Definiciones locales disponibles para el resolvente*

Descripción de tipo de archivo	Interfaces API afectadas	Archivos candidatos
Archivos de configuración de resolvente base	Todas las API	<ol style="list-style-type: none"> <li>1. GLOBALTCPIPDATA</li> <li>2. Variable de entorno RESOLVER_CONFIG</li> <li>3. /etc/resolv.conf</li> <li>4. SYSTCPD DD-name</li> <li>5. userid.TCPIP.DATA</li> <li>6. jobname.TCPIP.DATA</li> <li>7. SYS1.TCPPARMS(TCPDATA)</li> <li>8. DEFAULTTCPIPDATA</li> <li>9. TCPIP.TCPIP.DATA</li> </ol>
Tablas de conversión	Todas las API	<ol style="list-style-type: none"> <li>1. Variable de entorno X_XLATE</li> <li>2. userid.STANDARD.TCPXLBIN</li> <li>3. jobname.STANDARD.TCPXLBIN</li> <li>4. hlq.STANDARD.TCPXLBIN</li> <li>5. Tabla de conversión proporcionada por el resolvente, miembro STANDARD de SEZATCPX</li> </ol>

Tabla 13. Definiciones locales disponibles para el resolvente (continuación)

Descripción de tipo de archivo	Interfaces API afectadas	Archivos candidatos
Tablas de hosts locales	endhostent endnetent getaddrinfo gethostbyaddr gethostbyname gethostent GetHostNumber GetHostResol GetHostString getnameinfo getnetbyaddr getnetbyname getnetent IsLocalHost Resolve sethostent setnetent	IPv4 1. Variable de entorno X_SITE 2. Variable de entorno X_ADDR 3. /etc/hosts 4. userid.HOSTS.xxxxINFO 5. jobname.HOSTS.xxxxINFO 6. hlq.HOSTS.xxxxINFO  IPv6 1. GLOBALIPNODES 2. Variable de entorno RESOLVER_IPNODES 3. userid.ETC.IPNODES 4. jobname.ETC.IPNODES 5. hlq.ETC.IPNODES 6. DEFAULTIPNODES 7. /etc/ipnodes

**Nota:** La tabla anterior es una copia parcial de una tabla del manual *Communications Server IP Configuration Guide (SC31-8775)*. En ese manual encontrará la tabla completa.



---

## Apéndice D. Configurar INETD

Este apéndice se propone ayudarle a resolver algunos problemas comunes que pueden surgir al configurar INETD, o durante la tarea de comprobar y/o modificar una configuración existente.

El daemon INETD proporciona gestión de servicio para una red IP. Reduce la carga del sistema invocando otros daemons solo cuando se necesiten y proporcionando varios servicios simples de Internet (como el de echo) internamente. INETD lee el archivo de configuración inetd.conf para determinar qué servicios adicionales hay que proporcionar. ETC.SERVICES se emplea para enlazar los servicios a los puertos.

---

### inetd.conf

Los servicios que se basan en INETD se definen en inetd.conf, que INETD lee en el momento del arranque. La ubicación y el nombre predeterminados de inetd.conf es /etc/inetd.conf. Encontrará un archivo inetd.conf de ejemplo en /samples/inetd.conf.

Las reglas de sintaxis válidas para las entradas de inetd.conf son las siguientes:

- Los comentarios empiezan con una almohadilla (#) o con punto y coma (;) y siguen hasta el final de la línea.
- Las entradas son sensibles a las mayúsculas/minúsculas.
- Las entradas son sensibles a los campos, pero no a las columnas.
- Los campos se separan con un espacio o con un carácter de tabulación.
- Las entradas abarcan múltiples líneas, ajustándose a estas reglas de sintaxis adicionales.
  - La división debe estar entre dos palabras separadas (separadas por un espacio o por un carácter de tabulación).
  - La línea de continuación debe empezar con un espacio o con un carácter de tabulación.
  - No se pueden intercalar comentarios en la línea de continuación.

Cada entrada consta de 7 campos posicionales, correspondientes al formato:

nombre\_servicio tipo\_socket protocolo distintivo\_espera id\_usuario programa\_servidor argumentos\_pr

#### [dirección\_ip:]nombre\_servicio

dirección\_ip es una IP local, seguida de dos puntos (:). Si se especifica, se utiliza la dirección en lugar de INADDR\_ANY o del valor predeterminado actual. Para solicitar específicamente INADDR\_ANY, utilice "\*:". Si se especifica dirección\_ip (o dos puntos), sin ninguna otra entrada en la línea, esta pasa a ser la dirección predeterminada en las líneas ulteriores hasta que se especifique un nuevo valor predeterminado. nombre\_servicio es un nombre de servicio de todos conocido o definido por el usuario. El nombre especificado debe coincidir con uno de los nombres de servidor definidos en ETC.SERVICES.

#### tipo\_socket

Puede ser stream o dgram, para indicar que se emplea un socket en modalidad continua o un socket de datagramas para el servicio.

**protocolo[,sndbuf=n[,rcvbuf=n]]**

protocolo puede ser tcp[4|6] o udp[4|6], y sirve para calificar adicionalmente el nombre del servicio. El nombre del servicio y también el protocolo deben coincidir con una entrada de ETC.SERVICES, salvo que el "4" o el "6" no se deben incluir en la entrada ETC.SERVICES.

sndbuf y rcvbuf especifican el tamaño de las memorias intermedias de envío y recepción. El tamaño, representado por n, debe estar expresado en bytes, pero también se puede añadir una "k" o una "m" para indicar kilobytes o megabytes, respectivamente. sndbuf y rcvbuf se pueden usar en cualquier orden.

**distintivo\_espera[.max]**

wait o nowait. wait indica que el daemon es de una sola hebra y no se servirá otra petición hasta que se complete la primera. Si se especifica nowait, INETD emite una aceptación cuando se recibe una petición de conexión en un socket de modalidad continua. Si se especifica wait, el servidor es el encargado de emitir la aceptación si este es un socket de modalidad continua.

max es el número máximo de usuarios permitidos para solicitar servicio en un intervalo de 60 segundos. El valor predeterminado es 40. Si se sobrepasa, el puerto del servicio de cierra.

**userid[.group]**

userid es el ID de usuario bajo el que se debe ejecutar el daemon bifurcado. Este ID de usuario puede ser diferente del ID de usuario de INETD. Los permisos asignados a este ID de usuario dependen de las necesidades del servicio. El ID de usuario de INETD necesita el permiso BPX.DAEMON para conmutar el proceso bifurcado a este ID de usuario.

El valor group opcional, separado del ID de usuario por un punto (.), permite que el servidor se ejecute con un ID de grupo distinto del predeterminado para este ID de usuario.

**programa\_servidor**

programa\_servidor es el nombre de vía de acceso completo del servicio. Por ejemplo: /usr/sbin/rlogind es el nombre de vía de acceso completo del mandato rlogind.

**argumentos\_programa\_servidor**

Puede haber un máximo de 20 argumentos. El primer argumento es el nombre del servidor.

---

## ETC.SERVICES

INETD utiliza ETC.SERVICES para correlacionar números de puertos y protocolos con los servicios a los que debe dar soporte. Puede ser un conjunto de datos MVS o un archivo z/OS UNIX. Hay un ejemplo que viene en SEZAINST(SERVICES) y que también puede estar disponible como /usr/lpp/tcpip/samples/services. El orden de búsqueda de ETC.SERVICES depende del método de arranque de INETD, z/OS UNIX o MVS nativo.

Las reglas de sintaxis válidas para la especificación de información de servicio son las siguientes:

- Un conjunto de datos MVS ETC.SERVICES debe ser fijo o un bloque fijo con un LRECL comprendido entre 56 y 256

- Un archivo HFS de ETC.SERVICES debe tener una longitud de línea máxima de 256
- Los elementos de una línea se separan con espacios o con caracteres de tabulación
- Cada servicio ocupa una sola línea
- El nombre del servicio debe empezar en la primera posición de una línea
- La longitud máxima del nombre de servicio o del nombre de alias es de 32 caracteres
- El número máximo de alias que se reconocerán es 35
- Los nombres de servicio y de alias son sensibles a las mayúsculas/minúsculas
- Los comentarios empiezan con una almohadilla (#) o con punto y coma (;) y siguen hasta el final de la línea.

Cada entrada consta de 4 campos posicionales, correspondientes al formato:

nombre\_servicio    número\_puerto/protocolo    alias

**nombre\_servicio**

Especifica un nombre de servicio de todos conocido o definido por el usuario

**número\_puerto**

Especifica el número de puerto de socket empleado para el servicio

**protocolo**

Especifica el protocolo de transporte empleado para el servicio. Los valores válidos son tcp y udp

**alias**    Especifica una lista de los nombres de servicio no oficiales

## Orden de búsqueda utilizado en el entorno z/OS UNIX

El orden de búsqueda empleado para acceder a ETC.SERVICES en z/OS UNIX es el siguiente. La búsqueda finaliza en el primer archivo encontrado:

1. **/etc/services**
2. **userid.ETC.SERVICES**  
userid es el ID de usuario empleado para iniciar INETD.
3. **hlq.ETC.SERVICES**  
hlq representa el valor de la sentencia DATASETPREFIX especificada en el archivo de configuración de resolvente base (si se da con él); en caso contrario, hlq es TCPIP por defecto.

## Orden de búsqueda utilizado en el entorno MVS nativo

El orden de búsqueda empleado para acceder a ETC.SERVICES en MVS es el siguiente. La búsqueda finaliza en el primer conjunto de datos encontrado:

1. **Tarjeta DD //SERVICES**  
Se utiliza el conjunto de datos asignado a la sentencia DD SERVICES.
2. **userid.ETC.SERVICES**  
userid es el ID de usuario empleado para iniciar INETD.
3. **jobname.ETC.SERVICES**  
jobname es el nombre especificado en la sentencia JCL de JOB para los trabajos por lotes o el nombre de un procedimiento iniciado.
4. **hlq.ETC.SERVICES**

hlq representa el valor de la sentencia DATASETPREFIX especificada en el archivo de configuración de resolvente base (si se da con él); en caso contrario, hlq es TCPIP por defecto.

**Nota:** El hecho de iniciar INETD mediante BPXPATCH no hace que se utilice el orden de búsqueda nativo de MVS, ya que BPXBATCH ejecuta el mandato de inicio en el entorno z/OS UNIX. El orden de búsqueda nativo de MVS solo se utiliza al iniciar un módulo de carga MVS, como SEZALOAD(FTP).

---

## Definiciones de puertos en PROFILE.TCPIP

No hay que confundir las definiciones de PORT (o PORTRANGE) en PROFILE.TCPIP con los puertos definidos en ETC.SERVICES, ya que estas definiciones tienen distintas finalidades. Los puertos definidos en PROFILE.TCPIP se utilizan en TCPIP para ver si el puerto está reservado para un determinado servicio. ETC.SERVICES se utiliza en INETD para correlacionar un puerto con un servicio.

INETD, cuando recibe una petición en un puerto supervisado, bifurca un proceso hijo (con el servicio solicitado) que se llama inetdx, donde inetd es el nombre de trabajo de INETD (depende del método de arranque) y x es un número de un solo dígito.

Esto complica la reserva de puertos, ya que si un puerto supervisado por INETD está reservado en PROFILE.TCPIP, conviene utilizar el nombre del procedimiento JCL iniciado para el espacio de direcciones de kernel z/OS UNIX para permitir que casi cualquier proceso se enlace con el puerto. Este nombre suele ser OMVS, a menos que se especifique explícitamente un nombre distinto en el parámetro STARTUP\_PROC del miembro parmlib BPXPRMxx.

En la lista que sigue se explica cómo determinar el nombre del trabajo, dado el entorno en el que se ejecuta la aplicación:

- Las aplicaciones que se ejecutan desde un proceso por lotes emplean el nombre del trabajo por lotes.
- Las aplicaciones que se inician desde la consola del operador de MVS emplean el nombre del procedimiento iniciado (STC) como nombre de trabajo.
- Las aplicaciones que se ejecutan desde un ID de usuario TSO emplean ese ID de usuario como nombre de trabajo.
- Las aplicaciones que se ejecutan desde la shell z/OS tienen normalmente un nombre de trabajo que es el ID de usuario que ha iniciado sesión más un sufijo de un solo carácter.
- Los usuarios autorizados pueden ejecutar aplicaciones desde la shell z/OS y utilizar la variable de entorno \_BPX\_JOBNAME para establecer el nombre del trabajo. En este caso, el valor especificado para la variable de entorno es el nombre del trabajo.
- El nombre del procedimiento JCL iniciado para el espacio de direcciones de kernel de los servicios de sistemas UNIX (USS) se puede usar para permitir que casi cualquier llamador de la API de socket bind() (exceptuando a los usuarios de la API de Pascal) se enlacen con el puerto. Este nombre suele ser OMVS, a menos que se especifique explícitamente un nombre distinto en el parámetro STARTUP\_PROC del miembro parmlib BPXPRMxx.
- Las aplicaciones z/OS UNIX iniciadas por INETD utilizan el nombre de trabajo del servidor INETD.



**Nota:** Aunque conviene no hacerlo, los puertos definidos en ETC.SERVICES pueden diferir del número de puerto reservado para el servicio en PROFILE.TCPIP.

---

## /etc/inetd.pid

El proceso INETD crea un archivo temporal, /etc/inetd.pid, que contiene el PID (ID de proceso) del daemon INETD que se ejecuta en este momento. Este valor de PID se emplea para identificar los registros de syslog que se originaron a partir del proceso INETD, y para proporcionar el valor de PID para los mandatos que lo necesiten, como el mandato kill. También se utiliza como mecanismo de bloqueo para impedir que haya más de un proceso INETD activo.

---

## Arranque

La implementación de INETD en z/OS UNIX se encuentra por defecto en /usr/sbin/inetd y admite dos parámetros de arranque opcionales que no dependen de la posición:

/usr/sbin/inetd [-d] [inetd.conf]

**-d** Opción de depuración. Los datos de salida de depuración se escriben en la salida de errores estándar (stderr), que el daemon syslogd puede encaminar a un archivo. En el manual *Communications Server IP Configuration Guide (SC31-8775)* hallará más información sobre syslogd. Fíjese en que, cuando se inicia de esta manera, INETD no bifurcará un proceso hijo para iniciar un servicio.

### inetd.conf

Archivo de configuración. El valor predeterminado es /etc/inetd.conf

Conviene iniciar INETD en tiempo de IPL. La manera más corriente de hacerlo es iniciarlo desde /etc/rc o /etc/inittab (solo en z/OS 1.8 o superior). También se puede iniciar desde un trabajo o una tarea iniciada mediante BPXBATCH o desde una sesión de shell de un usuario que posea la autorización pertinente.

## /etc/rc

Cuando se inicia desde el script de shell de inicialización de z/OS UNIX, /etc/rc, INETD emplea el orden de búsqueda de z/OS UNIX para localizar ETC.SERVICES. Se suministra un archivo /etc/rc de ejemplo como /samples/rc. Los mandatos de ejemplo que se pueden usar para iniciar INETD son los siguientes:

```
# Iniciar INETD
_BPX_JOBNAME='INETD' /usr/sbin/inetd /etc/inetd.conf &
sleep 5
```

## /etc/inittab

z/OS 1.8 o superior proporciona un método alternativo, /etc/inittab, para emitir mandatos durante la inicialización de z/OS UNIX. /etc/inittab permite definir el parámetro respawn, que reinicia el proceso automáticamente cuando finaliza (se envía un WTOR al operador para un segundo reinicio al cabo de 15 minutos). Cuando se inicia desde /etc/inittab, INETD emplea el orden de búsqueda de z/OS UNIX para localizar ETC.SERVICES. Se suministra un /etc/inittab de ejemplo como /samples/inittab. El mandato de ejemplo que se puede usar para iniciar INETD es el siguiente:

```
# Iniciar INETD
inetd::respfrk:/usr/sbin/inetd /etc/inetd.conf
```

**Nota:** Tenga en cuenta que el parámetro `respfrk` utilizado en el ejemplo transferirá el atributo `respawn` a todos los procesos bifurcados, incluido RSE. Cuando el cliente cierra la conexión, `respawn` la volverá a iniciar. El servidor RSE la finalizará de nuevo más tarde, debido al tiempo de espera excedido. En el manual *UNIX System Services Planning (GA22-7800)* obtendrá más información sobre `inittab`.

## BPXBATCH

El método de arranque BPXBATCH funciona para los trabajos de STC y de usuario. Observe que INETD es un proceso en segundo plano, por lo que el paso BPXBATCH para iniciar INETD finalizará unos segundos después del arranque. Cuando se inicia con BPXBATCH, INETD emplea el orden de búsqueda de z/OS UNIX para localizar ETC.SERVICES. El JCL que figura en: Figura 11 es un procedimiento de ejemplo para iniciar INETD (el paso KILL elimina un proceso INETD activo, si lo hay):

```
//INETD    PROC PRM=
//*
//KILL      EXEC PGM=BPXBATCH,
//          PARM='SH ps -e | grep inetd | cut -c 1-10 | xargs -n 1 kill'
//*
//INETD     EXEC PGM=BPXBATCH,TIME=NOLIMIT,REGION=0M,
//          PARM='PGM /usr/sbin/inetd &PRM'
//STDERR    DD PATH='/tmp/bpxbatch.start.inetd.stderr',
//          PATHOPTS=(OWRONLY,OCREAT,OTRUNC),
//          PATHMODE=SIRWXU
//* STDIN, STDOUT y STDENV toman por defecto el valor /dev/null
//* La salida del daemon INETD se puede controlar mediante los valores de syslogd
//*
```

Figura 11. JCL de arranque de INETD

**Nota:** STDIN, STDOUT y STDERR, si se asignan, deben ser archivos de z/OS UNIX. STDENV puede ser un conjunto de datos MVS o un archivo z/OS UNIX. En el manual *UNIX System Services Command Reference (SA22-7802)* obtendrá más información sobre BPXBATCH.

**Nota:** `inetd.conf` puede ser un conjunto de datos o miembro MVS cuando INETD se inicia mediante BPXBATCH. Para ello, codifique la sentencia PARM como en el ejemplo que sigue (utilice solo comillas simples (')):

```
// PARM='PGM /usr/sbin/inetd //'SYS1.TCPPARMS(INETCONF)'' &PRM'
```

## Sesión de shell

Cuando se inicia desde una sesión de shell, INETD emplea el orden de búsqueda de z/OS UNIX para localizar ETC.SERVICES. Los siguientes mandatos de ejemplo se pueden usar (si la persona que los emite tiene autorización suficiente) para detener e iniciar INETD (# es el indicador de z/OS UNIX):

```
1. # ps -e | grep inetd
   7 ?          0:00 /usr/sbin/inetd
2. # kill 7
3. # _BPX_JOBNAME='INETD' /usr/sbin/inetd
```

**Nota:** Este método no es aconsejable para el arranque inicial; los métodos `/etc/rc` o `/etc/inittab` son más apropiados porque se ejecutan cuando z/OS UNIX se inicializa.

---

## Seguridad

INETD es un proceso z/OS UNIX y, por lo tanto, exige definiciones de OMVS válidas en el software de seguridad para el ID de usuario asociado a INETD. Hay que establecer UID, HOME y PROGRAM para el ID de usuario, junto con el GID para el grupo predeterminado del usuario. Si INETD se inicia mediante `/etc/rc` o `/etc/inittab`, el ID de usuario se hereda del kernel z/OS UNIX, y su valor predeterminado es OMVSKERN.

```
ADDGROUP OMVSGRP OMVS(GID(1))
ADDUSER OMVSKERN DFLTGRP(OMVSGRP) NOPASSWORD +
      OMVS(UID(0) HOME('/') PROGRAM('/bin/sh'))
```

INETD es un daemon que necesita acceso a funciones como `setuid()`. Por lo tanto, el ID de usuario empleado para iniciar INETD debe tener acceso de lectura (READ) al perfil BPX.DAEMON, en la clase FACILITY. Si este perfil no está definido, el UID 0 es obligatorio.

```
PERMIT BPX.DAEMON CLASS(FACILITY) ACCESS(READ) ID(OMVSKERN)
```

El ID de usuario de INETD también debe tener permiso de ejecución (EXECUTE) para el programa `inetd` (`/usr/sbin/inetd`), acceso de lectura (READ) a sus archivos `inetd.conf` y `ETC.SERVICES`, y acceso de escritura (WRITE) a `/etc/inetd.pid`. Si desea ejecutar INETD sin el UID 0, puede otorgar acceso de control (CONTROL) al perfil SUPERUSER.FILESYS, en la clase UNIXPRIV, para proporcionar los permisos necesarios sobre los archivos z/OS UNIX.

Los programas que necesiten autorización de daemon deben estar controlados por programa si BPX.DAEMON está definido en la clase FACILITY. Esto ya se hace para el programa INETD predeterminado (`/usr/sbin/inetd`), pero se debe establecer manualmente si se propone utilizar una copia o una versión personalizada. Utilice el mandato **extattr +p** para hacer que un archivo z/OS UNIX esté controlado por programa. Utilice la clase PROGRAM de RACF para hacer que un módulo de carga MVS esté controlado por programa.

Los programadores del sistema que tengan que reiniciar INETD desde dentro de su sesión de shell iniciarán INETD utilizando sus permisos. Por lo tanto, deben tener la misma lista de permisos que el ID de usuario normal de INETD. Además, también necesitan permisos para listar y detener el proceso INETD. Esto se puede lograr de muchas maneras.

- UID 0

Esta configuración no está recomendada para los ID de usuario “human”, porque no hay restricciones relacionadas con z/OS UNIX.

- Acceso de lectura (READ) al perfil BPX.SUPERUSER de la clase FACILITY

Permite que el usuario se convierta en UID 0 con el mandato **su**. Esta es la configuración recomendada.

- Acceso a perfiles individuales que cubran los permisos necesarios

- Acceso de lectura (READ) a SUPERUSER.PROCESS.GETPSENT, en la clase UNIXPRIV (para el mandato **ps**)
- Acceso de lectura (READ) a SUPERUSER.PROCESS.KILL, en la clase UNIXPRIV (para el mandato **kill**)
- Acceso de lectura (READ) a BPX.JOBNAME, en la clase FACILITY (para la variable de entorno `_BPX_JOBNAME`)

En el manual *UNIX System Services Command Reference* (SA22-7802) obtendrá más información sobre los mandatos **extattr** y **su**. Consulte el manual *UNIX System*

*Services Planning (GA22-7800)* para obtener más información sobre la clase UNIXPRIV y los perfiles BPX.\* de la clase FACILITY. En el manual *Security Server RACF Security Administrator's Guide (SA22-7683)* hallará más información sobre las definiciones del segmento OMVS y la clase PROGRAM.

---

## Requisitos de Developer para System z

Developer para System z depende de INETD a la hora de poner a punto la conexión entre cliente y host. También impone requisitos adicionales sobre la configuración de INETD descrita más arriba.

### INETD

Los valores del entorno INETD, que se pasan al iniciar un proceso, y los permisos del ID de usuario de INETD deben estar debidamente establecidos para que INETD inicie el servidor RSE.

- Si INETD se inicia mediante JCL utilizando BPXBATCH, el tamaño de la región debe ser 0.
- Si INETD se inicia desde una sesión de shell TSO/OMVS, el tamaño de la región TSO debe ser igual o mayor que 2096128.
- Si INETD se inicia mediante /etc/rc o /etc/inittab, se utiliza el tamaño de región de SYS1.PROCLIB(BPX0INIT), que es 0 por defecto.
- Acceso de lectura y ejecución (READ y EXECUTE) a los directorios de instalación de Developer para System z, cuyo valor predeterminado es /usr/lpp/wd4z/\*.

### Daemon RSE

El daemon RSE que se inicia mediante INETD cuando un cliente se conecta al puerto 4035 se utiliza para la autenticación, para iniciar el servidor RSE y para devolver el número de puerto de nuevo al cliente para una comunicación ulterior. Para poder hacerlo, el ID de usuario asignado al daemon RSE (en inetd.conf) necesita tener los siguientes permisos:

- Definiciones de OMVS válidas en el software de seguridad; hay que establecer UID, HOME y PROGRAM, junto con el GID del grupo predeterminado del usuario
- Acceso de lectura (READ) al perfil BPX.DAEMON de la clase FACILITY
- Acceso de lectura y ejecución (READ y EXECUTE) a los directorios de instalación de Developer para System z, cuyo valor predeterminado es /usr/lpp/wd4z/\*
- Acceso de lectura y ejecución (READ y EXECUTE) al directorio de configuración de Developer para System z, cuyo valor predeterminado es /usr/lpp/wd4z/rse/lib/, pero conviene utilizar un directorio distinto, como /etc/wd4z/

---

## Apéndice E. Configurar SSL

Este apéndice se propone ayudarle a resolver algunos problemas comunes que pueden surgir al configurar la capa de sockets segura (SSL), o durante la tarea de comprobar y/o modificar una configuración existente.

Que una comunicación sea segura implica asegurarse de que su interlocutor sea la persona que afirma ser y transmitir información de tal manera que a las otras personas les resulte difícil interceptar los datos y leerlos. La capa de sockets segura (SSL) proporciona capacidad para ello en una red TCP/IP. Funciona utilizando certificados digitales para identificarse a sí mismo, y un protocolo de claves públicas para cifrar la comunicación. En el manual *Security Server RACF Security Administrator's Guide (SA22-7683)* hallará más información sobre los certificados digitales y el protocolo de claves públicas empleado por SSL.

Las acciones necesarias para configurar las comunicaciones SSL para Developer para System z variarán según el local, en función de las necesidades exactas, del método de comunicación RSE empleado y de lo que ya esté disponible en el local.

En ese apéndice clonaremos las definiciones actuales de RSE para poder tener una segunda conexión que utilice SSL. Ambos métodos de conexión, el de REXEC y el de daemon, se configurarán para SSL. También crearemos nuestros propios certificados de seguridad para que los utilicen diferentes componentes de la conexión RSE. Todo ello se realizará siguiendo estos pasos:

1. Clonar la configuración RSE existente
2. Determinar qué archivo(s) de claves hay que utilizar
3. Crear un almacén de claves con keytool
4. Crear una base de datos de claves (solo daemon) con RACF o con gskkyman
5. Activar SSL actualizando ssl.properties
6. Probar la conexión

**Nota:** Consulte el libro blanco *Setting up SSL for RSE Communication (SC23-5816)*, en la biblioteca Internet de Developer para System z, <http://www-306.ibm.com/software/awdtools/devzseries/library/>, para obtener más información sobre cómo utilizar un certificado firmado por una autoridad certificadora (CA) de confianza o sobre cómo configurar su propia CA.

A lo largo de este apéndice se utiliza un convenio de denominación uniforme:

- Certificado: wd4zrse
- Almacenamiento de claves y certificados: wd4zssl.\*
- Contraseña: rsessl

En la mayoría de las tareas que se describen a continuación, se espera que esté activo en z/OS UNIX. Para ello, emita el mandato TSO **OMVS**. Utilice el mandato **exit** para volver a TSO.

---

## Clonar la configuración RSE existente

En este paso se crea una nueva instancia del servidor RSE y del daemon RSE para ejecutarse en paralelo con la(s) existente(s). De este modo, la prueba SSL no entorpecerá las operaciones normales. Como ya se aconsejó en: “Guardar el archivo de configuración rsed.envvars en otro directorio” en la página 34, en los siguientes mandatos de ejemplo se espera que los archivos de configuración estén en /etc/wd4z/.

```
$ cd /etc/wd4z
$ mkdir ssl
$ cp * ssl
cp: FSUM6254 "ssl" es un directorio (no copiado)
$ ls ssl
rsecomm.properties  server.zseries      ssl.properties
rsed.envvars         setup.env.zseries
$ cd ssl
$ su
# oedit /etc/services

rsessl          4047/tcp          #Developer para System z RSE using SSL

añadir servicio rsessl, utilizando el puerto 4047
# oedit /etc/inetd.conf

rsessl stream tcp nowait OMVSKERN /usr/lpp/wd4z/rse/lib/fekfrsed rsed -d /etc/wd4z/ssl

añadir el daemon rsessl, utilizando el directorio de configuración /etc/wd4z/ssl
# ps -e | grep inetd
7 ?          0:00 /usr/sbin/inetd
# kill 7
# _BPX_JOBNAME='INETD' /usr/sbin/inetd
# exit
$ netstat | grep 4047
INETD4  00016619 0.0.0.0..4047          0.0.0.0..0          Escucha
```

Los mandatos que figuran más arriba crean un subdirectorio llamado ssl y los pueblan con los archivos de configuración obligatorios. No hace falta hacer cambios de configuración (todavía). Podemos compartir el directorio de instalación y los componentes MVS, ya que no son específicos de SSL. Sin embargo, hay que definir un nuevo daemon (rsessl) que utiliza los nuevos archivos de configuración. El puerto que se asigna al nuevo daemon es el 4047.

Para obtener más información sobre las acciones anteriores, consulte: Capítulo 4, “Activar componentes z/OS UNIX de Developer para System z”, en la página 33.

---

## Determinar qué archivo(s) de claves hay que utilizar

Los certificados de identidad y las claves de cifrado/descifrado que SSL emplea se almacenan en un archivo de claves. Existen distintas implementaciones de este archivo de claves, en función del tipo de aplicación.

El daemon RSE es una aplicación SSL del sistema y utiliza un archivo de base de datos de claves. Esta base de datos de claves puede ser un archivo físico creado por gskkyman o un anillo de claves gestionado por el software de seguridad (por ejemplo, por RACF). El servidor RSE (que se inicia mediante el daemon o REXEC) es una aplicación SSL Java y utiliza un archivo de almacén de claves creado por keytool. Actualmente, RACF no tiene ningún soporte inmediato para SSL Java.

Así que, para conectar por medio de REXEC, todo lo que necesitamos es el archivo de almacén de claves:

- almacén de claves (keytool)

Para conectar por medio del daemon, necesitamos ambos archivos, el de almacén de claves y el de base de datos de claves:

- almacén de claves (keytool)
- base de datos de claves (RACF o gskkyman)

En el manual *Security Server RACF Security Administrator's Guide (SA22-7683)* encontrará información sobre RACF y los certificados digitales. La documentación de gskkyman se encuentra en *Cryptographic Services System SSL Programming (SC24-5901)*. La documentación de keytool está disponible en <http://java.sun.com/j2se/1.4.2/docs/tooldocs/solaris/keytool.html>.

---

## Crear un almacén de claves con keytool

"keytool -genkey" genera un par de claves (una clave pública y una clave privada asociada). Luego envuelve la clave pública en un certificado X.509 v1 autofirmado, que se almacena como cadena de certificados de un solo elemento. Esta cadena de certificados y la clave privada se almacenen como una entrada (identificado por un alias) en un archivo de almacén de claves (nuevo).

**Nota:** Hay que incluir Java en los directorios de búsqueda de mandatos. Para poder ejecutar keytool, podría ser necesaria la siguiente sentencia (/usr/lpp/java/J1.4 es el directorio en el que está instalado Java):  
PATH=\$PATH:/usr/lpp/java/J1.4/bin

Toda la información se puede pasar como un parámetro, pero debido a las limitaciones de longitud que tiene la línea de mandatos, se necesita algo de interactividad.

```
$ keytool -genkey -alias wd4zrse -validity 3650 -keystore wd4zssl.jks -storepass
rsessl -keypass rsessl
¿Cuál es su nombre y su apellido?
[Desconocido]: wd4z rse ssl
¿Cuál es el nombre de su unidad de organización (OU)?
[Desconocido]: wd4z
¿Cuál es el nombre de su organización?
[Desconocido]: IBM
¿Cuál es el nombre de su ciudad o localidad?
[Desconocido]: Raleigh
¿Cuál es el nombre de su estado o provincia?
[Desconocido]: NC
¿Cuál es el código de dos letras de esta unidad?
[Desconocido]: US
¿Es correcto CN=wd4z rse ssl, OU=wd4z, O=IBM, L=Raleigh, ST=NC, C=US? (teclea "yes"
o "no")
[no]: yes
$ ls
rsecomm.properties  server.zseries      ssl.properties
rsed.envvars        setup.env.zseries   wd4zssl.jks
```

El certificado autofirmado creado más arriba es válido durante 10 años (sin contar los días bisiestos). Se almacena en /etc/wd4z/ssl/wd4zssl.jks utilizando el alias wd4zrse. Su contraseña (rsessl) es idéntica a la contraseña del almacén de claves, que es un requisito para RSE.

El resultado se puede verificar con la opción -list:



```
$ keytool -list -alias wd4zrse -keystore wd4zssl.jks -storepass rssl -v
Nombre de alias: wd4zrse
Fecha de creación: May 24, 2007
Tipo de entrada: keyEntry
Longitud de la cadena de certificados: 1
Certificado 1:
Propietario: CN=wd4z rse ssl, OU=wd4z, O=IBM, L=Raleigh, ST=NC, C=US
Emisor: CN=wd4z rse ssl, OU=wd4z, O=IBM, L=Raleigh, ST=NC, C=US
Número de serie: 46562b2b
Válido desde: 5/24/07 2:17 PM hasta: 5/21/17 2:17 PM
Huellas digitales del certificado
MD5: 9D:6D:F1:97:1E:AD:5D:B1:F7:14:16:4D:9B:1D:28:80
SHA1: B5:E2:31:F5:B0:E8:9D:01:AD:2D:E6:82:4A:E0:B1:5E:12:CB:10:1C
```

---

## Crear una base de datos de claves (solo daemon)

Como ya se ha mencionado, el daemon es una aplicación SSL del sistema que utiliza una base de datos de claves. Esta puede ser un archivo físico creado por gskkyman o bien un anillo de claves RACF. Los anillos de claves RACF son el método preferido para gestionar claves privadas y certificados de SSL del sistema.

**Nota:** SSL del sistema utiliza el recurso de servicio criptográfico integrado (ICSF) si está disponible. ICSF proporciona soporte criptográfico por hardware, que se utilizará en lugar de los algoritmos de software de SSL del sistema. Hallará más información al respecto en el manual *Cryptographic Services System SSL Programming (SC24-5901)*.

## Crear un anillo de claves con RACF

No realice este paso si utiliza gskkyman para SSL del sistema.

El mandato **RACDCERT** instala y mantiene claves privadas y certificados en RACF. RACF permite gestionar múltiples claves privadas y certificados en forma de grupo. Estos grupos se llaman anillos de claves.

En el manual *Security Server RACF Command Language Reference (SA22-7687)* encontrará los detalles del mandato **RACDCERT**.

```
RDEFINE FACILITY IRR.DIGTCERT.LIST UACC(NONE)
RDEFINE FACILITY IRR.DIGTCERT.LISTRING UACC(NONE)
PERMIT IRR.DIGTCERT.LIST CLASS(FACILITY) ACCESS(READ) ID(omvskern)
PERMIT IRR.DIGTCERT.LISTRING CLASS(FACILITY) ACCESS(READ) ID(omvskern)
SETOPTS RACLIST(FACILITY) REFRESH

RACDCERT ID(omvskern) GENCERT SUBJECTSDN(CN('wd4z rse ssl') +
OU('wd4z') O('IBM') L('Raleigh') SP('NC') C('US')) +
NOTAFTER(2017-05-21) WITHLABEL('wd4zrse') KEYUSAGE(HANDSHAKE)

RACDCERT ID(omvskern) ADDRING(wd4zssl.racf)
RACDCERT ID(omvskern) CONNECT(LABEL('wd4zrse') RING(wd4zssl.racf) +
DEFAULT USAGE(PERSONAL))
```

En el ejemplo anterior se empieza por crear los perfiles necesarios y permitir que el ID de usuario OMVSKERN tenga acceso a los anillos de claves. El ID de usuario que se utilice debe coincidir con el ID de usuario codificado en `/etc/inetd.conf` para el daemon RSE SSL. El próximo paso consiste en crear un nuevo certificado autofirmado con la etiqueta `wd4zrse`. No se necesita ninguna contraseña. Luego este certificado se añade a un anillo de claves recién creado (`wd4zssl.racf`). Igual que con el certificado, tampoco se necesita una contraseña para el anillo de claves. El tiempo de vigencia del certificado coincide con el del creado con `keytool`.



El resultado se puede verificar con la opción list:

RACDCERT ID(omvskern) LIST

Información de certificado para el usuario OMVSKERN:

```
Etiqueta: wd4zrse
ID de certificado: 2QjW10Xi0sXZ1aaEqZmihUBA
Estado: TRUST
Fecha inicial: 2007/05/24 00:00:00
Fecha final: 2017/05/21 23:59:59
Número de serie:
    >00<
Nombre del emisor:
    >CN=wd4z rse ssl.OU=wd4z.O=IBM.L=Raleigh.SP=NC.C=US<
Nombre del sujeto:
    >CN=wd4z rse ssl.OU=wd4z.O=IBM.L=Raleigh.SP=NC.C=US<
Tipo de clave privada: Non-ICSF
Tamaño de clave privada: 1024
Asociaciones de anillo:
    Propietario de anillo: OMVSKERN
Anillo:
    >wd4zssl.racf<
```

## Crear una base de datos de claves con gskkyman

No realice este paso si utiliza RACF para SSL del sistema.

gskkyman es un programa dirigido por menú y basado en la shell z/OS UNIX que crea, puebla y gestiona un archivo z/OS UNIX que contiene claves privadas, peticiones de certificado y certificados. Este archivo z/OS UNIX se llama base de datos de claves.

**Nota:** Las siguientes sentencias podrían ser necesarias para configurar el entorno de cara a gskkyman. Hallará más información al respecto en el manual System SSL Programming (SC24-5901).

```
PATH=$PATH:/usr/lpp/gskssl/bin
export NLSPATH=/usr/lpp/gskssl/lib/nls/msg/En_US.IBM-1047/%N:$NLSPATH
export STEPLIB=$STEPLIB:SYS1.SIEALNKE
```

\$ gskkyman

Menú de base de datos

- 1 - Crear base de datos nueva
- 2 - Abrir base de datos
- 3 - Cambiar contraseña de base de datos
- 4 - Cambiar longitud de registro de base de datos
- 5 - Suprimir base de datos
- 6 - Crear archivo de parámetros de clave
- 0 - Salir del programa

Entre el número de opción: 1

Entre el nombre de la base de datos de claves (pulse Intro para volver al menú): wd4zssl.kdb

Entre la contraseña de la base de datos (pulse Intro para volver al menú): rsessl

Vuelva a entrar la contraseña de la base de datos: rsessl

Entre el tiempo de caducidad de la contraseña en días (pulse Intro si no caduca):

Entre la longitud de registro de la base de datos (pulse Intro para utilizar 2500):

Se ha creado la base de datos de claves /etc/wd4z/ssl/wd4zssl.kdb.

Pulse Intro para continuar.

Menú de gestión de claves

Base de datos: /etc/wd4z/ssl/wd4zssl.kdb

- 1 - Gestionar claves y certificados
- 2 - Gestionar certificados
- 3 - Gestionar peticiones de certificado
- 4 - Crear petición de certificado nueva
- 5 - Recibir certificado solicitado o un certificado de renovación
- 6 - Crear un certificado autofirmado
- 7 - Importar un certificado
- 8 - Importar un certificado y una clave privada
- 9 - Mostrar la clave predeterminada
- 10 - Almacenar contraseña de base de datos
- 11 - Mostrar longitud de registro de base de datos

0 - Salir del programa

Entre el número de opción (pulse Intro para volver al menú anterior): 6

Tipo de certificado

- 1 - Certificado de CA con clave RSA de 1024 bites
- 2 - Certificado de CA con clave RSA de 2048 bites
- 3 - Certificado de CA con clave RSA de 4096 bites
- 4 - Certificado de CA con clave DSA de 1024 bites
- 5 - Certificado de usuario o servidor con clave RSA de 1024 bites
- 6 - Certificado de usuario o servidor con clave RSA de 2048 bites
- 7 - Certificado de usuario o servidor con clave RSA de 4096 bites
- 8 - Certificado de usuario o servidor con clave DSA de 1024 bites

Seleccione el tipo de certificado (pulse Intro para volver al menú): 5

Entre la etiqueta (pulse Intro para volver al menú): wd4zrse

Entre el nombre de sujeto del certificado

Nombre común (necesario): wd4z rse ssl

Unidad de organización (OU, opcional): wd4z

Organización (necesario): IBM

Ciudad/Localidad (opcional): Raleigh

Estado/Provincia (opcional): NC

País/Región (2 caracteres - necesario): US

Entre el número de días durante los que el certificado será válido (valor predeterminado, 365): 3650

Entre 1 para especificar nombres de sujetos alternativos o entre 0 para continuar: 0

Espere por favor .....

El certificado se ha creado.

Pulse Intro para continuar.

Menú de gestión de claves

Base de datos: /etc/wd4z/ssl/wd4zssl.kdb

- 1 - Gestionar claves y certificados
- 2 - Gestionar certificados
- 3 - Gestionar peticiones de certificado
- 4 - Crear petición de certificado nueva
- 5 - Recibir certificado solicitado o un certificado de renovación
- 6 - Crear un certificado autofirmado
- 7 - Importar un certificado
- 8 - Importar un certificado y una clave privada
- 9 - Mostrar la clave predeterminada
- 10 - Almacenar contraseña de base de datos
- 11 - Mostrar longitud de registro de base de datos

0 - Salir del programa

Entre el número de opción (pulse Intro para volver al menú anterior): 0

```

$ ls -l
total 152
-rwxr-xr-x 1 IBMUSER SYS1 333 May 24 13:52 rsecomm.properties
-rwxr-xr-x 1 IBMUSER SYS1 6067 May 24 13:52 rsed.envvars
-rwxr-xr-x 1 IBMUSER SYS1 332 May 24 13:52 server.zseries
-rwxr-xr-x 1 IBMUSER SYS1 645 May 24 13:52 setup.env.zseries
-rwxr-xr-x 1 IBMUSER SYS1 638 May 24 13:52 ssl.properties
-rw-r--r-- 1 IBMUSER SYS1 1224 May 24 14:17 wd4zssl.jks
-rw----- 1 IBMUSER SYS1 35080 May 24 14:24 wd4zssl.kdb
-rw----- 1 IBMUSER SYS1 80 May 24 14:24 wd4zssl.rdb
$ chmod 644 wd4zssl.kdb
$ chmod 644 wd4zssl.rdb
$ ls -l
total 152
-rwxr-xr-x 1 IBMUSER SYS1 333 May 24 13:52 rsecomm.properties
-rwxr-xr-x 1 IBMUSER SYS1 6067 May 24 13:52 rsed.envvars
-rwxr-xr-x 1 IBMUSER SYS1 332 May 24 13:52 server.zseries
-rwxr-xr-x 1 IBMUSER SYS1 645 May 24 13:52 setup.env.zseries
-rwxr-xr-x 1 IBMUSER SYS1 638 May 24 13:52 ssl.properties
-rw-r--r-- 1 IBMUSER SYS1 1224 May 24 14:17 wd4zssl.jks
-rw-r--r-- 1 IBMUSER SYS1 35080 May 24 14:24 wd4zssl.kdb
-rw-r--r-- 1 IBMUSER SYS1 80 May 24 14:24 wd4zssl.rdb

```

En el ejemplo anterior se empieza por crear una base de datos de claves llamada wd4zssl.kdb con la contraseña rsessl. Una vez creada, la base de datos se puebla creando un nuevo certificado autofirmado que se almacena bajo la etiqueta wd4zrse y que tiene la misma contraseña (rsessl) que la que se empleó para la base de datos de claves (este es un requisito de RSE).

gskkyman asigna la base de datos de claves con una máscara de bit de permiso 600 (muy seguro, el único que tiene acceso es el propietario). Los permisos se tienen que establecer para que sean menos restrictivos, a menos que el daemon utilice el mismo ID de usuario que el creador de la base de datos de claves. Las máscaras que se pueden usar para el mandato **chmod** son 640 (el propietario tiene acceso de lectura/escritura y el grupo del propietario tiene acceso de lectura) o 644 (el propietario tiene acceso de lectura/escritura y los demás tienen acceso de lectura).

El resultado se puede verificar seleccionando la opción **Mostrar información de certificado**, en el submenú **Gestionar claves y certificados**:

```
$ gskkyman
```

```
Menú de base de datos
```

- 1 - Crear base de datos nueva
- 2 - Abrir base de datos
- 3 - Cambiar contraseña de base de datos
- 4 - Cambiar longitud de registro de base de datos
- 5 - Suprimir base de datos
- 6 - Crear archivo de parámetros de clave

```
0 - Salir del programa
```

```
Entre el número de opción: 2
```

```
Entre el nombre de la base de datos de claves (pulse Intro para volver al menú): wd4zssl.kdb
```

```
Entre la contraseña de la base de datos (pulse Intro para volver al menú): rsessl
```

```
Menú de gestión de claves
```

```
Base de datos: /etc/wd4z/ssl/wd4zssl.kdb
```

- 1 - Gestionar claves y certificados
- 2 - Gestionar certificados
- 3 - Gestionar peticiones de certificado

- 4 - Crear petición de certificado nueva
- 5 - Recibir certificado solicitado o un certificado de renovación
- 6 - Crear un certificado autofirmado
- 7 - Importar un certificado
- 8 - Importar un certificado y una clave privada
- 9 - Mostrar la clave predeterminada
- 10 - Almacenar contraseña de base de datos
- 11 - Mostrar longitud de registro de base de datos

0 - Salir del programa

Entre el número de opción (pulse Intro para volver al menú anterior): 1

Lista de claves y certificados

Base de datos: /etc/wd4z/ssl/wd4zssl.kdb

1 - wd4zrse

0 - Volver al menú de selección

Entre el número de etiqueta (Intro para volver al menú de selección, p para lista anterior): 1

Menú de claves y certificados

Etiqueta: wd4zrse

- 1 - Mostrar información de certificado
- 2 - Mostrar información de clave
- 3 - Establecer clave como valor predeterminado
- 4 - Establecer estado de confianza de certificado
- 5 - Copiar certificado y clave en otra base de datos
- 6 - Exportar certificado a un archivo
- 7 - Exportar certificado y clave a un archivo
- 8 - Suprimir certificado y clave
- 9 - Cambiar etiqueta
- 10 - Crear un certificado firmado y una clave
- 11 - Crear una petición de renovación de certificado

0 - Salir del programa

Entre el número de opción (pulse Intro para volver al menú anterior): 1

Información de certificado

Etiqueta: wd4zrse

ID de registro: 14

ID de registro del emisor: 14

De confianza: Sí

Versión: 3

Número de serie: 45356379000ac997

Nombre del emisor: wd4z rse ssl

wd4z

IBM

Raleigh

NC

US

Nombre de sujeto: wd4z rse ssl

wd4z

IBM

Raleigh

NC

US

Fecha de efectividad: 2007/05/24

Fecha de caducidad: 2017/05/21

Algoritmo de clave pública: rsaEncryption

Tamaño de clave pública: 1024

Algoritmo de signatura: sha1WithRsaEncryption  
ID exclusivo del emisor: Ninguno  
ID exclusivo del sujeto: Ninguno  
Número de extensiones: 3

Entre 1 para visualizar las extensiones, entre 0 para volver al menú: 0

Menú de claves y certificados

Etiqueta: wd4zrse

- 1 - Mostrar información de certificado
- 2 - Mostrar información de clave
- 3 - Establecer clave como valor predeterminado
- 4 - Establecer estado de confianza de certificado
- 5 - Copiar certificado y clave en otra base de datos
- 6 - Exportar certificado a un archivo
- 7 - Exportar certificado y clave a un archivo
- 8 - Suprimir certificado y clave
- 9 - Cambiar etiqueta
- 10 - Crear un certificado firmado y una clave
- 11 - Crear una petición de renovación de certificado
  
- 0 - Salir del programa

Entre el número de opción (pulse Intro para volver al menú anterior): 0

---

## Activar SSL actualizando ssl.properties

Ahora que los certificados ya están a punto, RSE se puede iniciar utilizando SSL. En función de las definiciones que se hayan elegido en los pasos anteriores, habrá que proporcionar distintos valores en `ssl.properties`.

```
$ oedit ssl.properties
```

**enable\_ssl=true**

Los valores válidos son true y false (que es el predeterminado).

**daemon\_keydb\_file=wd4zssl.racf**

Nombre de la base de datos de claves de gskkyman o nombre del anillo de claves de RACF. Solo se necesita para utilización del daemon.

**daemon\_keydb\_password=**

Contraseña de la base de datos de claves de gskkyman o se deja en blanco para el anillo de claves de RACF. Solo se necesita para utilización del daemon.

**daemon\_key\_label=wd4zrse**

Etiqueta que se utiliza para gskkyman/RACF, si no se ha definido como etiqueta predeterminada. Si se utiliza el valor predeterminado, hay que quitar el carácter de comentario. Solo se necesita para utilización del daemon.

**server\_keystore\_file=wd4zssl.jks**

Nombre del almacén de claves de keytool.

**server\_keystore\_password=rsessl**

Contraseña del almacén de claves de keytool.

---

## Probar la conexión

Ahora, la configuración de host SSL ya se ha realizado y se puede probar conectando con el cliente Developer para System z. Dado que hemos creado una nueva configuración (clonando la existente) para que SSL la utilice, hay que definir una nueva conexión mediante estas características:

- REXEC: utilizando la vía de acceso `/etc/wd4z/ssl`
- daemon: utilizando el puerto 4047

**Nota:** Para poder ejecutar una aplicación SSL del sistema (conexión del daemon), SYS1.SIEALNKE debe estar en LINKLIST o en STEPLIB. Si prefiere el método de STEPLIB, añada la siguiente sentencia al final de `rsed.envvars`. Sin embargo, tenga presente que el hecho de utilizar STEPLIB en z/OS UNIX afecta negativamente al rendimiento, como se describe en: “Evitar el uso de STEPLIB” en la página 67.

- Si la última directiva STEPLIB definida anteriormente en `rsed.envvars` es igual a STEPLIB=NONE  
STEPLIB=SYS1.SIEALNKE
- Si la última directiva STEPLIB definida anteriormente en `rsed.envvars` no es igual a STEPLIB=NONE  
STEPLIB=\$STEPLIB:SYS1.SIEALNKE

Al conectarse, el host y el cliente se iniciarán con algún establecimiento de enlace para poder configurar una vía de acceso segura. Parte del establecimiento de enlace es el intercambio de certificados. El cliente Developer para System z, si no reconoce el certificado del host, preguntará al usuario si el certificado es de confianza.

*Figura 12. Importar certificado de host*

Con el botón Finalizar, el usuario puede aceptar este certificado como de confianza, después de lo cual continuará la inicialización de la conexión.

**Nota:** En la conexión del daemon se emplean 2 ubicaciones de certificado (SSL del sistema y SSL Java), produciendo 2 certificados distintos y, por consiguiente, 2 confirmaciones.

Una vez reconocido el certificado ante el cliente, este diálogo ya no vuelve a aparecer. La lista de certificados de confianza se puede gestionar seleccionando **Ventana > Preferencias... > Sistemas Remotos > SSL**, con lo cual aparece el diálogo:

*Figura 13. Preferencias*

Si la comunicación SSL falla, el cliente devolverá un mensaje de error. Hay más información en los distintos archivos de anotaciones (`home/.eclipse/RSE/USERID/*` y `/tmp/rsedaemon.log`), como se describe en: “Anotaciones de RSE” en la página 78.

---

## Apéndice F. Configurar APPC

Este apéndice se propone ayudarle a resolver algunos problemas comunes que pueden surgir al configurar APPC (comunicaciones avanzadas programa a programa) o durante la tarea de comprobar y/o modificar una configuración existente.

En los manuales *MVS Planning APPC/MVS Management (SA22-7599)* y *MVS Initialization and Tuning Reference (SA22-7592)* encontrará información adicional sobre la gestión de APPC y sobre los miembros parmlib que figuran a continuación.

Fíjese en que no se trata de describir una configuración completa de APPC, sino tan solo de resaltar algunos aspectos clave que podrían ser válidos para su local.

El miembro SYS1.SAMPLIB(ATBALL) contiene una lista (con sus descripciones) de todos los miembros (de ejemplo) relacionados con APPC en SYS1.SAMPLIB.

---

### VSAM

APPC/MVS almacena sus datos de configuración en miembros SYS1.PARMLIB y en dos conjuntos de datos VSAM:

- El conjunto de datos VSAM de programas de transacción (TP) (nombre predeterminado SYS1.APPCTP) contiene información de planificación y seguridad para los programas z/OS.
- El conjunto de datos VSAM de información complementaria (SI) (nombre predeterminado SYS1.APPCSI) contiene la traducción de los nombres de destino simbólicos utilizados por los TP locales de z/OS y los servidores APPC/MVS.

El TP es un programa de aplicación que emplea APPC para comunicarse con un TP en el mismo sistema o en otro con el fin de acceder a recursos. La configuración APPC de Developer para System z activa un nuevo TP que se llama FEKFRSRV, y al que nos referimos como servicio de mandatos TSO.

El trabajo de ejemplo que aparece en la figura 14 es una concatenación de miembros de ejemplo SYS1.SAMPLIB(ATBTPVSM) y SYS1.SAMPLIB(ATBSIVSM), y se puede usar para definir los VSAM de APPC.

```

//APPCVSAM JOB <parámetros del trabajo>
//*
/* PRECAUCIÓN: esto no es un procedimiento JCL ni un trabajo completo.
/* Antes de usar este ejemplo, tendrá que hacer las siguientes
/* modificaciones:
/* 1. Cambie los parámetros de trabajo para que respondan a los requisitos de su sistema.
/* 2. En lugar de *****, escriba el volumen en el que se pondrán los VSAM de APPC.
/*
//TP      EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
        DEFINE CLUSTER (NAME(SYS1.APPCTP) -
                        VOLUME(*****) -
                        INDEXED REUSE -
                        SHAREOPTIONS(3 3) -
                        RECORDSIZE(3824 7024) -
                        KEYS(112 0) -
                        RECORDS(300 150)) -
        DATA      (NAME(SYS1.APPCTP.DATA)) -
        INDEX      (NAME(SYS1.APPCTP.INDEX))
/*
//SI      EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
        DEFINE CLUSTER (NAME(SYS1.APPCSI) -
                        VOLUME(*****) -
                        INDEXED REUSE -
                        SHAREOPTIONS(3 3) -
                        RECORDSIZE(248 248) -
                        KEYS(112 0) -
                        RECORDS(50 25)) -
        DATA      (NAME(SYS1.APPCSI.DATA)) -
        INDEX      (NAME(SYS1.APPCSI.INDEX))
/*

```

Figura 14. JCL para crear los VSAM de APPC

## VTAM

APPC es una implementación del protocolo LU 6.2 de la arquitectura de red de sistemas (SNA). SNA proporciona formatos y protocolos que definen una gran variedad de componentes SNA físicos y lógicos, como la unidad lógica (LU). LU 6.2 es un tipo de unidad lógica que se ha diseñado específicamente para manejar las comunicaciones entre programas de aplicación.

Para poder usar SNA en MVS, debe instalar y configurar VTAM (método de acceso de telecomunicaciones virtuales). Para poder utilizar las tareas del sistema APPC, primero hay que activar VTAM.

La parte específica de APPC de la configuración de VTAM consta de tres pasos:

1. Defina el nombre de modalidad empleado (por ejemplo APPCHOST) en VTAM utilizando SYS1.SAMPLIB(ATBLJOB) para ensamblar y enlazar-editar SYS1.SAMPLIB(ATBLMODE) en su SYS1.VTAMLIB. Encontrará los detalles en el miembro SYS1.SAMPLIB(ATBLMODE).
2. Defina APPC/MVS como aplicación VTAM, copiando el miembro de ejemplo SYS1.SAMPLIB(ATBAPPL) en un conjunto de datos de la concatenación de SYS1.VTAMLST. Encontrará los detalles en el miembro SYS1.SAMPLIB(ATBAPPL).
3. Utilice el mandato de consola **v net,act,id=atbappl** para activar la aplicación que acaba de definir (donde net es igual al nombre de su STC de VTAM).



Utilice el mandato de consola **d net,appl**s para verificar que la aplicación está activa. Añada el nombre del miembro a SYS1.VTAMLST(ATCCONxx) si desea que se active al iniciarse VTAM.

El nombre-acb (ACBNAME) de MVSLU01 empleado en el miembro de ejemplo SYS1.SAMPLIB(ATBAPPL) se puede cambiar para que coincida con los estándares del local, pero debe coincidir con las definiciones existentes en el miembro SYS1.PARMLIB(APPCPMxx).

```

MVSLU01 APPL  ACBNAME=MVSLU01,      C
                APPC=YES,             C
                AUTOSES=0,            C
                DDRAINL=NALLOW,      C
                DLOGMOD=APPCHOST,     C
                DMINWNL=5,            C
                DMINWNR=5,            C
                DRESPL=NALLOW,       C
                DSESLIM=10,           C
                LMDENT=19,            C
                MODETAB=LOGMODES,     C
                PARSESS=YES,          C
                SECACPT=CONV,         C
                SRBEXIT=YES,          C
                VPACING=1              C

```

Figura 15. SYS1.SAMPLIB(ATBAPPL)

Consulte el manual *Communications Server bookshelf (F1A1BK61 for z/OS 1.7)* para obtener más información sobre cómo configurar VTAM.

---

## SYS1.PARMLIB(APPCPMxx)

Para habilitar y hacer posible el flujo de conversaciones entre sistemas, los locales deben definir unidades lógicas (LU) entre las que poder enlazar sesiones. Cada local debe definir como mínimo una LU para que pueda tener lugar el proceso de APPC/MVS, aun cuando el proceso de APPC permanezca en un solo sistema. Las LU son parte de las definiciones que se realizan en SYS1.PARMLIB(APPCPMxx).

El servicio de mandatos TSO exige que APPC esté configurado de tal manera que tenga un LU base que pueda manejar las peticiones de entrada y las de salida.

La definición de LU se tiene que añadir al miembro SYS1.PARMLIB(APPCPMxx) y debe incluir los parámetros BASE y SCHED(ASCH). El miembro APPCPMxx también especifica qué conjuntos de datos VSAM de perfil de transacción (TP) y de información complementaria (SI) se emplearán.

Figura 16 es un miembro SYS1.PARMLIB(APPCPMxx) de ejemplo que se puede usar para el servicio de mandatos TSO.

```

LUADD
  ACBNAME(MVSLU01)
  BASE
  SCHED(ASCH)
  TPDATA(SYS1.APPCTP)
  SIDEINFO DATASET(SYS1.APPCSI)

```

Figura 16. SYS1.PARMLIB(APPCPMxx)

Cuando un sistema tiene múltiples nombres de LU, habrá que hacer algunos cambios en función de qué LU seleccione el sistema para que sea la LU BASE. La LU BASE del sistema se determina mediante estos factores:

1. La LU base del sistema viene representada por la última sentencia LUADD que contiene ambos parámetros, NOSCHED y BASE. Este tipo de LU base del sistema permite que las peticiones de salida se procesen cuando no hay planificadores de transacciones activos.
2. Si no hay sentencias LUADD que contengan NOSCHED y BASE, la LU base del sistema viene representada por la última sentencia LUADD que contenga el parámetro BASE y especifique que ASCH es el planificador de transacciones APPC/MVS. Esto se puede hacer codificando SCHED(ASCH) o no codificando el parámetro SCHED en absoluto (ASCH es el valor predeterminado de SCHED).

Si su sistema tiene una LU con los parámetros BASE y NOSCHED, esta LU se emplearía como LU BASE, pero el servicio de mandatos TSO no funcionaría, porque esta LU no tiene un planificador de transacciones que maneje las peticiones a la transacción FEKFRSRV. Si no se puede cambiar esta LU con el fin de eliminar el parámetro NOSCHED, se puede establecer que la variable de entorno \_FEKFSCMD\_PARTNER\_LU de rsed.envvars sea la LU que tenga BASE y SCHED(ASCH), como en:

```
_FEKFSCMD_PARTNER_LU=MVSLU01
```

Para obtener más información sobre rsed.envvars, vea: “Personalizar rsed.envvars, el archivo de configuración de RSE” en la página 35.

---

## SYS1.PARMLIB(ASCHPMxx)

El planificador de transacciones APPC/MVS (cuyo nombre predeterminado es ASCH) inicia y planifica los programas de transacciones como respuesta a las peticiones de entrada que solicitan conversaciones. El miembro SYS1.PARMLIB(ASCHPMxx) controla su funcionamiento, por ejemplo, con definiciones de clase de transacción.

La clase de transacción APPC que se utilice para el servicio de mandatos TSO debe tener suficientes iniciadores APPC para permitir un iniciador para cada usuario de Developer para System z.

**Nota:** Existe una diferencia entre los iniciadores de APPC y z/OS (JES). Cuando un cliente Developer para System z se conecta al host, el servidor de mandatos TSO se inicia con un iniciador APPC. Developer para System z utiliza un iniciador z/OS (JES) cuando se construye un proyecto, cuando se realiza una comprobación de sintaxis remota o cuando se somete un trabajo.

Para el servicio de mandatos TSO, también hay que especificar las especificaciones predeterminadas en las secciones OPTIONS y TPDEFAULT.

Figura 17 en la página 123 es un miembro SYS1.PARMLIB(ASCHPMxx) de ejemplo que se puede usar para el servicio de mandatos TSO.

```
CLASSADD  
  CLASSNAME(A)  
  MAX(20)  
  MIN(1)  
  MSGLIMIT(200)
```

```
OPTIONS  
  DEFAULT(A)
```

```
TPDEFAULT  
  REGION(2M)  
  TIME(5)  
  MSGLEVEL(1,1)  
  OUTCLASS(X)
```

Figura 17. *SYS1.PARMLIB(ASCHPMxx)*

**Nota:** A efectos de depuración, el centro de soporte de IBM le podría pedir que aumente el valor de MSGLIMIT, para que se escriban más datos de salida en el archivo de anotaciones.

---

## Activar los cambios de APPC

Ahora se pueden activar los cambios de configuración documentados en los pasos anteriores. Hay distintas maneras de hacerlo, en función de las circunstancias:

- APPC todavía no está activo. Para iniciar APPC/MVS, entre los siguientes mandatos de consola (siendo xx los 2 últimos caracteres de los miembros SYS1.PARMLIB relacionados):

1. S APPC,SUB=MSTR,APPC=xx
2. S ASCH,SUB=MSTR,ASCH=xx

Añada estos mandatos a SYS1.PARMLIB(COMMNDxx) para que se inicien en el momento del arranque del sistema.

- APPC ya está activo. APPC puede recargar dinámicamente los miembros SYS1.PARMLIB mediante un mandato de consola SET (siendo xx los dos últimos caracteres de los miembros SYS1.PARMLIB relacionados):

1. SET APPC=xx
2. SET ASCH=xx

Para verificar la configuración de APPC, se pueden usar los mandatos de consola **D APPC** y **D ASCH**. En el manual *MVS System Commands (SA22-7627)* hallará más información sobre los mandatos de consola mencionados.

---

## Definir la transacción del servicio de mandatos TSO

Una vez que APPC/MVS esté activo, se puede definir el servicio de mandatos TSO de Developer para System z, como se describe en: “(Opcional) Definir una transacción APPC para el servicio de mandatos TSO” en la página 24.

La manera documentada de definir la transacción APPC consiste en personalizar y someter hlq.SFEKSAMP(FEKAPPCC), siendo hlq el calificador de alto nivel empleado durante la instalación de Developer para System z (el valor predeterminado es FEK).

La transacción APPC también se puede definir en modalidad interactiva mediante la interfaz ISPF de APPC, que viene documentada en un libro blanco. En este libro blanco también se describe cómo configurar la transacción APPC para que recoja información de contabilidad específica del usuario.

El libro blanco *APPC and WebSphere Developer para System z* (SC23-5885) está disponible en la biblioteca de Developer para System z en Internet, <http://www-306.ibm.com/software/awdtools/devzseries/library/>

**Nota:** El JCL del programa de transacción (TP) que APPC emplea para iniciar el servicio de mandatos TSO ha cambiado en la versión 7.1 de Developer para System z. Si sigue las indicaciones del libro blanco para definir el TP, debe añadir la palabra clave NESTMACS a la línea PARM; por ejemplo:

```
// PARM='ISPSTART CMD(%FEKFRSRV TIMEOUT=60) NEWAPPL(ISR) NESTMACS'
```

---

## Glosario

### A

**ID de acción.** Identificador numérico de una acción, entre 0 y 999

**Servidor de aplicaciones.**

1. Programa que maneja todas las operaciones de aplicación entre los sistemas basados en navegador y las aplicaciones o bases de datos de negocio de fondo de una organización. Hay una clase especial de servidores de aplicación basados en Java que cumplen el estándar J2EE. El código J2EE puede portarse fácilmente entre estos servidores de aplicaciones. Puede soportar JSP y servlets para contenido Web dinámico y EJB para transacciones y acceso a bases de datos.
2. Destino de una petición procedente de una aplicación remota. En el entorno DB2, la función de servidor de aplicaciones la proporciona el servicio de datos distribuidos, y sirve para acceder a datos de DB2 desde aplicaciones remotas.
3. En una red distribuida, programa servidor que proporciona el entorno de ejecución de un programa de aplicación.
4. Destino de una petición procedente de un peticionario de aplicación. El sistema de gestión de bases de datos (DBMS) en el sitio del servidor de aplicaciones proporciona los datos solicitados.
5. Software que gestiona la comunicación con el cliente que solicita un activo y consultas del gestor de contenido.

### B

**Bidireccional (bi-di).** Relativo a scripts como el árabe o el hebreo que generalmente van de derecha a izquierda, excepto los números, que van de izquierda a derecha. Esta definición pertenece al glosario de la Localization Industry Standards Association (LISA).

**atributo bidireccional.** Tipo de texto, orientación del texto, intercambio numérico e intercambio simétrico.

**Petición de construcción.** Petición procedente del cliente para realizar una transacción de construcción.

**Transacción de construcción.** Trabajo iniciado en MVS para realizar construcciones después haberse recibido una petición de construcción procedente del cliente.

### C

**Compilar.**

1. En los lenguajes Integrated Language Environment (ILE), convertir las sentencias fuente en módulos que luego se pueden enlazar en programas o programas de servicio.
2. Convertir todo o parte de un programa expresado en un lenguaje de alto nivel en un programa informático expresado en un lenguaje intermedio, ensamblador o lenguaje de máquina.

**Contenedor.**

1. En CoOperative Development Environment/400, objeto del sistema que contiene y organiza archivos fuente. Son ejemplos de contenedor una biblioteca de i5/OS o un conjunto de datos particionado MVS.
2. En J2EE, entidad que proporciona a los componentes servicios de gestión del ciclo de vida, de seguridad, de despliegue y de tiempo de ejecución. (Sun) Cada tipo de contenedor (EJB, Web, JSP, servlet, applet y cliente de aplicaciones) también proporciona servicios específicos del componente.
3. En los Servicios BRM, objeto físico utilizado para almacenar y mover medios, como una caja, un estuche o un bastidor.
4. En un servidor de cintas virtual (VTS), receptáculo en el que es posible almacenar uno o más volúmenes lógicos exportados (LVOL). Volumen apilado que contiene uno o más LVOL y que reside fuera de una biblioteca VTS y que se considera como contenedor de dichos volúmenes.
5. Ubicación de almacenamiento físico de los datos. Por ejemplo, un archivo, un directorio o un dispositivo.
6. Columna o fila que se utiliza para disponer el diseño de un portlet o de otro contenedor en una página.
7. Elemento de la interfaz de usuario que contiene objetos. En el gestor de carpetas, objeto que puede contener otras carpetas o documentos.

### D

**Base de datos.** Conjunto de elementos de datos interrelacionados o independientes, almacenados conjuntamente para servir a una o más aplicaciones.

**Vista de definición de datos.** Contiene una representación local de bases de datos y de sus objetos y proporciona características para manipular estos objetos y exportarlos a una base de datos remota

**Conjunto de datos.** Unidad principal de almacenamiento y recuperación de datos, que consiste en una colección de datos en una de varias

disposiciones prescritas y descritas por la información de control a la que tiene acceso el sistema.

**Depurar.** Detectar, diagnosticar y eliminar errores en programas.

**Sesión de depuración.** Actividades de depuración que tienen lugar entre el momento en que un desarrollador inicia un depurador y el momento en que el desarrollador sale de él.

## E

**Memoria intermedia de error.** Parte del almacenamiento utilizado para contener temporalmente la información de salida de errores.

## F

.

## G

**Pasarela.**

1. Componente middleware entre Internet y los entornos de intranet durante las invocaciones de servicios Web.
2. Software que proporciona servicios entre los puntos finales y el resto del entorno Tivoli.
3. Componente del protocolo de voz por Internet, que proporciona un puente entre VoIP y los entornos de circuitos conmutados.
4. Dispositivo o programa utilizado para conectar redes o sistemas con diferentes arquitecturas de red. Los sistemas pueden tener distintas características, como distintos protocolos de comunicaciones, distinta arquitectura de red o distintas políticas de seguridad, en cuyo caso la pasarela adquiere un rol de conversión, así como un rol de conexión.

## H

.

## I

**Interactive System Productivity Facility (ISPF).** Programa IBM bajo licencia que funciona como editor de pantalla completa y gestor de diálogos. Si se utiliza para escribir programas de aplicaciones, proporciona una manera de generar paneles de pantallas estándar y diálogos interactivos entre el programador de aplicaciones y el usuario del terminal. ISPF consta de cuatro componentes principales: DM, PDF, SCLM, y C/S. El componente DM es el gestor de diálogos, que proporciona servicios a los diálogos y usuarios finales. El componente PDF es el recurso de desarrollo de programas, que proporciona servicios para ayudar al

desarrollador de diálogos o de aplicaciones. El componente SCLM es el gestor de bibliotecas de configuraciones de software, que proporciona servicios a los desarrolladores de aplicaciones para gestionar sus bibliotecas de entorno de aplicaciones. El componente C/S es el cliente/servidor, que permite ejecutar ISPF en una estación de trabajo programable, para visualizar los paneles utilizando la función de visualización del sistema operativo de la estación de trabajo, y para integrar herramientas y datos de la estación de trabajo con las herramientas y datos del host.

**Intérprete.** Programa que convierte y ejecuta cada instrucción de un lenguaje de programación de alto nivel antes de convertir y ejecutar la siguiente instrucción.

**Isomórfico.** Cada elemento compuesto (en otras palabras, un elemento que contiene otros elementos) del documento de instancia XML que comienza desde la raíz tiene un y solo un elemento de grupo COBOL correspondiente cuya profundidad de anidación es idéntica a la profundidad de anidación de su equivalente XML. Cada elemento no compuesto (en otras palabras, un elemento que no contiene otros elementos) en el documento de instancia XML que comienza desde la parte superior tiene un y solo un elemento COBOL correspondiente cuya profundidad de anidación es idéntica a la profundidad de anidación de su equivalente XML y cuya dirección de memoria en tiempo de ejecución puede identificarse de forma inequívoca.

## J

.

## K

.

## L

**Sección de enlace.** Sección de la división de datos de una unidad activada (un programa al que se llame o un método invocado) que describe los elementos de datos disponibles de la unidad que lo activa (un programa o un método). A estos elementos de datos les puede hacer referencia la unidad activada y la unidad que activa.

**Biblioteca de carga.** Biblioteca que contiene módulos de carga.

**Acción de bloquear.** Bloquea un miembro

## M

.

## N

**Vista Navegador.** Vista jerárquica de los recursos que hay en el entorno de trabajo.

**No isomórfico.** Correlación simple de elementos COBOL y elementos XML que pertenecen a documentos XML y a grupos COBOL que no son idénticos en la forma (no isomórficos). También se puede crear una correlación no isomorfa entre elementos no isomórficos de estructuras isomórficas.

## O

**Vista Consola de salida.** Visualiza la salida de un proceso y permite proporcionar entrada de teclado para un proceso.

**Vista Salida.** Muestra los mensajes, parámetros y resultados relacionados con los objetos con los que se esté trabajando.

## P

**Perspectiva.** Grupo de vistas que muestran los diversos aspectos de los recursos del entorno de trabajo. El usuario del entorno de trabajo puede pasar de una perspectiva a otra, en función de la tarea que esté realizando, y personalizar la disposición de las vistas y editores dentro de la perspectiva.

## Q

.

## R

**RAM.** Gestor de acceso a repositorios

**Sistema de archivos remoto.** Sistema de archivos que reside en un servidor o sistema operativo independiente.

**Sistema remoto.** Cualquier otro sistema en la red con el que puede comunicarse su sistema.

**Perspectiva.** Proporciona una interfaz para gestionar sistemas remotos utilizando convenciones similares a ISPF.

**Repositorio.**

1. Área de almacenamiento para los datos. Cada repositorio tiene un nombre y un tipo de elemento de negocio asociado. Por defecto, el nombre será el mismo que el nombre del elemento de negocio. Por ejemplo, un repositorio de facturas se llamará Facturas. Hay dos tipos de repositorios de información: local (específico del proceso) y global (reutilizable).

2. Conjunto de datos VSAM en el que se almacenan los estado de los procesos BTS. Cuando un proceso no se está ejecutando bajo el control de BTS, su estado (y los estados de sus actividades subordinadas) se conservan escribiéndose en un conjunto de datos de repositorio. Los estados de todos los procesos de un tipo de proceso en particular (y los de sus instancias de actividad) se almacenan en el mismo conjunto de datos del repositorio. Es posible escribir registros de varios tipos de proceso en el mismo repositorio.
3. Área de almacenamiento persistente del código fuente y de otros recursos de las aplicaciones. En un entorno de programación en equipo, el repositorio compartido permite que varios usuarios accedan a los recursos de la aplicación.
4. Recopilación de información acerca de los gestores de cola que son miembros de un clúster. Esta información incluye nombres de gestores de colas, sus ubicaciones, sus canales, qué colas hospedan, etc.

**Instancia de repositorio.** Proyecto o componentes que existe en un SCM.

**Vista repositorios.** Muestra la ubicación de los repositorios CVS que se han añadido al entorno de trabajo.

**Archivo de respuestas.**

1. Archivo que contiene un conjunto de respuestas predefinidas a preguntas formuladas por un programa y que se utiliza en lugar de entrar dichos valores de uno en uno.
2. Archivo ASCII que se puede personalizar con los datos de instalación y configuración que automatizan una instalación. Durante una instalación interactiva, es necesario entrar los datos de instalación y configuración, pero si se utiliza un archivo de respuestas, la instalación puede continuar sin ningún tipo de intervención.

## S

**Vista Servidores.** Visualiza una lista de todos los servidores, así como las configuraciones asociadas a ellos.

**Shell.** Interfaz de software entre los usuarios y el sistema operativo, que interpreta mandatos e interacciones del usuario y los comunica al sistema operativo. Cada sistema puede tener varias capas de shells para los diversos niveles de interacción de los usuarios.

**Nombre de shell.** Nombre de la interfaz de shell.

**Script de shell.** Archivo que contiene mandatos que la shell puede interpretar. El usuario escribe el nombre del



archivo de script en el indicador de mandatos de la shell para hacer que la shell ejecute los mandatos del script.

**Sidedeck.** Biblioteca que publica las funciones de un programa DLL. Los nombre de entradas y de módulos se almacenan en la biblioteca una vez compilado el código fuente.

**Instalación silenciosa.** Instalación que no envía mensajes a la consola, sino que almacena los mensajes y los errores en archivos de anotaciones. Además, en la instalación silenciosa se pueden utilizar archivos de respuestas como entrada de datos.

**Desinstalación silenciosa.** Proceso de desinstalación que no envía mensajes a la consola, sino que almacena los mensajes y los errores en archivos de anotaciones después de haberse invocado el mandato de desinstalación.

## T

**Lista de tareas.** Lista de procedimientos que se pueden ejecutar mediante un único flujo de control.

## U

**URL.** Localizador uniforme de recursos.

## V

.

## W

.

## X

.

## Y

.

## Z

.



---

## Avisos

Esta información se ha desarrollado para los productos y los servicios que se ofrecen en Estados Unidos.

Es posible que IBM no ofrezca en otros países los productos, servicios o características que se describen en este documento. El representante local de IBM le puede informar acerca de los productos y servicios que actualmente están disponibles en su localidad. Las referencias hechas a productos, programas o servicios de IBM no pretenden afirmar ni dar a entender que únicamente puedan utilizarse dichos productos, programas o servicios de IBM. Puede utilizarse en su lugar cualquier otro producto, programa o servicio funcionalmente equivalente que no vulnere ninguno de los derechos de propiedad intelectual de IBM. No obstante, es responsabilidad del usuario evaluar y verificar el funcionamiento de cualquier producto, programa o servicio que no sea de IBM.

IBM puede tener patentes o solicitudes de patente pendientes de aprobación que cubran alguno de los temas tratados en este documento. La posesión de este documento no le confiere ninguna licencia sobre dichas patentes. Puede enviar consultas sobre licencias por escrito a:

*IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
Estados Unidos de América*

Para consultas sobre licencias relativas a la información de doble byte (DBCS), póngase en contacto con el departamento de propiedad intelectual de IBM en su país o envíe las consultas, por escrito, a:

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

**El párrafo que sigue no se aplica al Reino Unido ni a ningún otro país en el que tales disposiciones sean incompatibles con la legislación local:** INTERNATIONAL BUSINESS MACHINES CORPORATION PROPORCIONA ESTA PUBLICACIÓN "TAL CUAL", SIN GARANTÍA DE NINGUNA CLASE, YA SEA EXPLÍCITA O IMPLÍCITA, INCLUIDAS, PERO SIN LIMITARSE A ELLAS, LAS GARANTÍAS IMPLÍCITAS DE NO VULNERACIÓN, DE COMERCIALIZACIÓN O IDONEIDAD PARA UN PROPÓSITO DETERMINADO. Algunas legislaciones no contemplan la declaración de limitación de responsabilidad, ni implícitas ni explícitas, en determinadas transacciones, por lo que cabe la posibilidad de que esta declaración no se aplique en su caso.

Esta información puede contener imprecisiones técnicas o errores tipográficos. La información incluida en este documento está sujeta a cambios periódicos; estos cambios se incorporarán en nuevas ediciones de la publicación. IBM puede efectuar mejoras y/o cambios en los productos y/o programas descritos en esta publicación en cualquier momento y sin previo aviso.

Cualquier referencia hecha en esta información a sitios Web no de IBM se proporciona únicamente para su comodidad y no debe considerarse en modo alguno como promoción de dichos sitios Web. Los materiales de estos sitios Web no forman parte de los materiales de IBM para este producto, y el usuario será responsable del uso que se haga de estos sitios Web.

IBM puede utilizar o distribuir la información que usted le suministre del modo que IBM considere conveniente sin incurrir por ello en ninguna obligación para con usted.

Los licenciarios de este programa que deseen obtener información acerca de él con el fin de: (i) intercambiar la información entre los programas creados independientemente y otros programas (incluido este) y (ii) utilizar mutuamente la información que se ha intercambiado, deben ponerse en contacto con:

*IBM Corporation  
P.O. Box 12195, Dept. TL3B/B503/B313  
3039 Cornwallis Rd.  
Research  
Triangle Park, NC 27709-2195  
Estados Unidos de América*

Dicha información puede estar disponible, sujeta a los términos y condiciones apropiados, incluyendo en algunos casos el pago de una cantidad.

IBM proporciona el programa bajo licencia descrito en este documento, así como todo el material bajo licencia disponible, según los términos del Acuerdo de Cliente de IBM, del Acuerdo Internacional de Programas bajo Licencia de IBM o de cualquier otro acuerdo equivalente entre ambas partes.

Los datos de rendimiento que se indican en este documento se han obtenido en un entorno controlado. Por consiguiente, es posible que los resultados que se obtengan en otros entornos operativos sean notablemente distintos. Es posible que algunas mediciones se hayan tomado en sistemas de nivel de desarrollo y no existe ningún tipo de garantía de que dichas mediciones sean las mismas en sistemas disponibles para el público en general. Además, es posible que algunas mediciones se hayan estimado por extrapolación. Los resultados reales pueden variar. Los usuarios de este documento deberán verificar los datos aplicables para su entorno específico.

La información concerniente a productos no IBM se ha obtenido de los suministradores de dichos productos, de sus anuncios publicados o de otras fuentes de información pública disponibles. IBM no ha comprobado dichos productos y no puede afirmar la exactitud en cuanto a rendimiento, compatibilidad u otras características relativas a productos no IBM. Las consultas acerca de las posibilidades de los productos que no son de IBM deben dirigirse a las personas que los suministran.

Todas las declaraciones relacionadas con la dirección o intención futuras de IBM están sujetas a cambio o retirada sin previo aviso, y únicamente representan objetivos.

Esta información contiene ejemplos de datos e informes utilizados en operaciones comerciales diarias. Para ilustrarlos de la forma más completa posible, los ejemplos incluyen nombres de personas, empresas, marcas y productos. Todos estos nombres son ficticios y cualquier parecido con los nombres y direcciones utilizados por una empresa real es mera coincidencia.

#### LICENCIA DE COPYRIGHT:

Esta información contiene programas de aplicación de ejemplo en lenguaje fuente, que ilustra las técnicas de programación en diversas plataformas operativas. Puede copiar, modificar y distribuir los programas de ejemplo de cualquier forma, sin tener que pagar a IBM, con intención de desarrollar, utilizar, comercializar o distribuir programas de aplicación que estén en conformidad con la interfaz de programación de aplicaciones (API) de la plataforma operativa para la que están escritos los programas de ejemplo. Los ejemplos no se han probado minuciosamente bajo todas las condiciones. Por lo tanto, IBM no puede garantizar ni dar por sentada la fiabilidad, la facilidad de mantenimiento ni el funcionamiento de los programas. Usted puede copiar, modificar y distribuir los programas de ejemplo de cualquier forma, sin tener que pagar a IBM, con el fin de desarrollar, utilizar, comercializar o distribuir programas de aplicación que estén en conformidad con las interfaces de programación de aplicaciones (API) de IBM.

Cada copia o cada parte de los programas de ejemplo o de los trabajos que se deriven de ellos debe incluir un aviso de copyright como se indica a continuación:

© (nombre de la empresa) (año). Algunas partes de este código procede de los programas de ejemplo de IBM Corp. © Copyright IBM Corp. \_escriba el año o los años\_. Reservados todos los derechos.

---

## **Marcas registradas y marcas de servicio**

Los siguientes términos son marcas registradas de International Business Machines Corporation en Estados Unidos y/o en otros países:

- CICS
- CICSplex
- DB2
- IBM
- IMS
- MVS
- OS/390
- RACF
- Rational
- System z
- VTAM
- WebSphere
- z/OS
- zSeries

Java y todas las marcas y logotipos basados en Java son marcas registradas de Sun Microsystems, Inc., en Estados Unidos y en otros países.

Microsoft, Windows, Windows NT y el logotipo de Windows son marcas registradas de Microsoft Corporation en Estados Unidos y/o en otros países.

UNIX es una marca registrada de The Open Group.

Los nombres de otras empresas, productos y servicios que se hayan señalado con dos asteriscos (\*\*) pueden ser marcas registradas o marcas de servicio de terceros.



---

# Hoja de Comentarios

IBM Rational Developer para System z  
Guía de configuración de host  
Versión 7.1.1

Número de Publicación SC11-3660-01

Por favor, sírvase facilitarnos su opinión sobre esta publicación, tanto a nivel general (organización, contenido, utilidad, facilidad de lectura,...) como a nivel específico (errores u omisiones concretos). Tenga en cuenta que los comentarios que nos envíe deben estar relacionados exclusivamente con la información contenida en este manual y a la forma de presentación de ésta.

Para realizar consultas técnicas o solicitar información acerca de productos y precios, por favor diríjase a su sucursal de IBM, business partner de IBM o concesionario autorizado.

Para preguntas de tipo general, llame a "IBM Responde" (número de teléfono 901 300 000).

Al enviar comentarios a IBM, se garantiza a IBM el derecho no exclusivo de utilizar o distribuir dichos comentarios en la forma que considere apropiada sin incurrir por ello en ninguna obligación con el remitente.

Comentarios:

Gracias por su colaboración.

Para enviar sus comentarios:

- Envíelos por correo a la dirección indicada en el reverso.
- Envíelos por fax al número siguiente: 1-800-227-5088 (EE.UU. y Canadá)
- Envíelos por correo electrónico a: [kfrye@us.ibm.com](mailto:kfrye@us.ibm.com)

Si desea obtener respuesta de IBM, rellene la información siguiente:

Nombre

Dirección

Compañía

Número de teléfono

Dirección de e-mail

IBM Corporation  
Information Development  
Department G71A / Bldg. 503  
P.O. Box 12195  
Research Triangle Park, NC





Número de Programa: 5724-T07

Impreso en España

SC11-3660-01

