

**Rational** IBM Rational Developer for System z  
Versión 7.6.1

## *Guía de configuración de host*





**Rational** IBM Rational Developer for System z  
Versión 7.6.1

## *Guía de configuración de host*



**Nota**

Antes de utilizar este documento, lea la información general que figura en el apartado “Avisos de documentación de IBM Rational Developer for System z” en la página 353.

**Quinta edición (mayo de 2010)**

Esta edición corresponde al producto IBM Rational Developer para System z Versión 7.6.1 (número de programa 5724-T07) y a todos los releases y modificaciones subsiguientes mientras no 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

Puede enviar sus comentarios por fax a: 1-800-227-5088 (EE.UU. y Canadá)

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 duplicación o la divulgación están sujetos a las restricciones establecidas en el contrato GSA ADP Schedule Contract con IBM Corp.

© Copyright IBM Corporation 2005, 2010.

# Contenido

<b>Figuras . . . . .</b>	<b>ix</b>
--------------------------	-----------

<b>Tablas . . . . .</b>	<b>xi</b>
-------------------------	-----------

<b>Acerca de este documento . . . . .</b>	<b>xiii</b>
---	-------------

A quién va dirigido este documento . . . . .	xiii
Resumen de cambios . . . . .	xiii
Descripción del contenido del documento . . . . .	xiv
Planificación . . . . .	xiv
Personalización básica . . . . .	xiv
(Opcional) CARMA (Common Access Repository Manager) . . . . .	xiv
(Opcional) Gestor de despliegue de aplicaciones (ADM) . . . . .	xv
(Opcional) SCLM Developer Toolkit . . . . .	xv
(Opcional) Otras tareas de personalización . . . . .	xv
Verificación de la instalación . . . . .	xvi
Mandatos de operador . . . . .	xvi
Resolución de problemas de configuración . . . . .	xvi
Consideraciones relativas a la seguridad . . . . .	xvi
Qué es Developer for System z . . . . .	xvi
Consideraciones de WLM . . . . .	xvi
Consideraciones acerca de los ajustes . . . . .	xvii
Consideraciones sobre el rendimiento . . . . .	xvii
consideraciones de CICSTS . . . . .	xvii
Personalizar el entorno TSO . . . . .	xvii
Ejecutar varias instancias . . . . .	xvii
Guía de migración . . . . .	xvii
Configurar SSL y autenticación de X.509 . . . . .	xvii
Configurar TCP/IP . . . . .	xviii
Configurar INETD . . . . .	xviii
Configurar TCP/IP . . . . .	xviii
Requisitos . . . . .	xviii

## Parte 1. Personalización de Developer for System z . . . . . 1

### Capítulo 1. Planificación . . . . . 3

Consideraciones acerca de la migración . . . . .	3
Consideraciones de planificación . . . . .	3
Visión general del producto . . . . .	3
Requisitos de conocimientos técnicos . . . . .	4
Requisitos de tiempo . . . . .	4
Consideraciones relativas a la preinstalación . . . . .	4
Elección de la configuración . . . . .	5
Productos requisito . . . . .	5
Recursos necesarios . . . . .	5
Consideraciones relativas a la preconfiguración . . . . .	9
Gestión de cargas de trabajo (WLM) . . . . .	9
Uso de recursos y límites del sistema . . . . .	9
Configuración necesaria de los productos obligatorios . . . . .	9
Consideraciones sobre los ID de usuario . . . . .	9
Consideraciones sobre el servidor . . . . .	10

Método de configuración . . . . .	11
Consideraciones relativas al predespliegue . . . . .	12
Lista de comprobación del cliente . . . . .	13

### Capítulo 2. Personalización básica . . . 15

Requisitos y lista de comprobación . . . . .	15
Configuración de la personalización . . . . .	15
Cambios de PARMLIB . . . . .	16
Establecer los límites de z/OS UNIX en BPXPRMxx . . . . .	16
Añadir tareas iniciadas a COMMNDxx . . . . .	17
Definiciones LPA en LPALSTxx . . . . .	18
Autorizaciones APF en PROGxx . . . . .	18
Definiciones LINKLIST en PROGxx . . . . .	19
Definiciones de LINKLIST y LPA requisito . . . . .	20
Definiciones LINKLIST para otros productos . . . . .	21
Cambios de PROCLIB . . . . .	21
Supervisor de trabajos JES . . . . .	21
Daemon RSE . . . . .	22
Daemon de bloqueo . . . . .	23
Limitaciones del JCL para la variable PARM . . . . .	24
Procedimientos de construcción remota ELAXF* . . . . .	25
Definiciones de seguridad . . . . .	26
FEJCNFG, archivo de configuración del supervisor de trabajos JES . . . . .	27
rzed.envvars, archivo de configuración de RSE . . . . .	32
Definir el rango de puertos (PORTRANGE) disponibles para el servidor RSE . . . . .	41
Definir parámetros de inicio Java adicionales con _RSE_JAVAOPTS . . . . .	42
Definir parámetros de inicio Java adicionales con _RSE_CMDSERV_OPTS . . . . .	47
ISPF.conf, archivo de configuración de la Pasarela de cliente TSO/ISPF de ISPF . . . . .	47
Componentes opcionales . . . . .	48
Verificación de la instalación . . . . .	49

### Capítulo 3. (Opcional) CARMA (Common Access Repository Manager) 51

Requisitos y lista de comprobación . . . . .	51
Componentes de CARMA . . . . .	52
Notas de migración de VSAM de CARMA . . . . .	53
Interfaz RSE con CARMA . . . . .	53
Inicio del servidor CARMA mediante el sometimiento por lotes . . . . .	55
Ajustar CRASRV.properties . . . . .	56
Ajustar CRASUBMT . . . . .	56
(Opcional) Inicio alternativo del servidor CARMA mediante CRASTART . . . . .	57
Ajustar CRASRV.properties . . . . .	58
Ajustar crastart.conf . . . . .	58
(Opcional) Inicio alternativo del servidor CARMA mediante la Pasarela de cliente TSO/ISPF . . . . .	60
Ajustar CRASRV.properties . . . . .	60
Ajustar ISPF.conf . . . . .	61

(Opcional) Activar los RAM (Repository Access Managers) de ejemplo . . . . .	62
Activar los RAM de PDS . . . . .	63
Activar los RAM de SCLM . . . . .	63
Activar los RAM de esqueleto . . . . .	63
(Opcional) Activar CA Endevor SCM RAM . . . . .	63
Requisitos y lista de comprobación . . . . .	64
Definir CA Endevor SCM RAM . . . . .	64
Inicio de CA Endevor SCM RAM mediante el sometimiento por lotes . . . . .	65
Inicio de CA Endevor SCM RAM mediante CRASTART . . . . .	67
(Opcional) Personalizar CRANDVRA . . . . .	68
(Opcional) Personalizar CA Endevor SCM RAM . . . . .	69
(Opcional) Soportar varios RAM . . . . .	69
Ejemplo . . . . .	70
(Opcional) Comparación entre IRXJCL y CRAXJCL . . . . .	71
Crear CRAXJCL . . . . .	71

## Capítulo 4. (Opcional) Gestor de despliegue de aplicaciones (ADM) . . . 73

Requisitos y lista de comprobación . . . . .	73
Repositorio CRD . . . . .	74
Programa de utilidad administrativo de CICS . . . . .	74
RESTful versus Servicio Web . . . . .	75
Servidor CRD utilizando la interfaz RESTful . . . . .	75
Región de conexión primaria CICS . . . . .	75
Regiones de conexión no primarias CICS . . . . .	76
(Opcional) Personalizar ID de transacción del servidor CRD . . . . .	76
Servidor CRD utilizando la interfaz de servicios Web . . . . .	77
Manejador de mensajes de conducto . . . . .	77
Región de conexión primaria CICS . . . . .	78
Regiones de conexión no primarias CICS . . . . .	78
(Opcional) Repositorio de manifiesto . . . . .	79

## Capítulo 5. (Opcional) SCLM Developer Toolkit . . . . . 81

Requisitos y lista de comprobación . . . . .	81
Prerrequisitos . . . . .	82
Actualizaciones de ISPF.conf para SCLMDT . . . . .	82
Actualizaciones de rsed.envvars para SCLMDT . . . . .	83
(Opcional) Conversión de nombres largos/abreviados . . . . .	84
Crear LSTRANS.FILE, el VSAM de conversión de nombres largos/abreviados . . . . .	84
Actualizaciones de rsed.envvars para la conversión de nombres largos/abreviados . . . . .	86
(Opcional) Instalar y personalizar Ant . . . . .	86
Actualizaciones de SCLM para SCLMDT . . . . .	87
Eliminar archivos antiguos de WORKAREA . . . . .	88

## Capítulo 6. (Opcional) Otras tareas de personalización . . . . . 89

(Opcional) Procedimiento almacenado DB2 . . . . .	89
Cambios del gestor de carga de trabajo (WLM) . . . . .	89
Cambios de PROCLIB . . . . .	90
Cambios de DB2 . . . . .	90
(Opcional) Soporte de Enterprise Service Tools (EST) . . . . .	91

(Opcional) Soporte de idiomas bidireccionales CICS . . . . .	92
(Opcional) Diagnóstico de mensajes de error de IRZ . . . . .	93
(Opcional) Cifrado SSL de RSE . . . . .	93
(Opcional) Rastreo de RSE . . . . .	96
(Opcional) Grupos de propiedades basadas en host . . . . .	98
(Opcional) Proyectos basados en host . . . . .	99
(Opcional) Integración del gestor de archivos . . . . .	100
(Opcional) Caracteres no editables . . . . .	101
(Opcional) Utilizar REXEC (o SSH) . . . . .	102
Acciones remotas (basadas en host) para subproyectos z/OS UNIX . . . . .	103
Método de conexión RSE alternativo . . . . .	103
Configuración de REXEC (o SSH) . . . . .	103
(Opcional) Transacción APPC para el servicio de mandatos TSO . . . . .	104
Preparación . . . . .	104
Implementación . . . . .	106
Consideraciones relativas a la utilización de APPC . . . . .	107
(Opcional) Limpieza de WORKAREA . . . . .	108

## Capítulo 7. Verificación de la instalación . . . . . 109

Verificar tareas iniciadas . . . . .	109
JMON, Supervisor de trabajos JES . . . . .	109
LOCKD, Daemon de bloqueo . . . . .	109
RSED, daemon RSE . . . . .	109
Verificar servicios . . . . .	111
Inicialización de IVP . . . . .	112
Disponibilidad de puertos . . . . .	113
Configuración de TCP/IP . . . . .	114
Conexión del daemon RSE . . . . .	115
Conexión del Supervisor de trabajos JES . . . . .	115
Conexión del daemon de bloqueo . . . . .	116
Conexión de la Pasarela de cliente TSO/ISPF de ISPF . . . . .	116
(Opcional) Conexión del servicio de mandatos TSO mediante APPC . . . . .	118
(Opcional) Conexión SCLMDT . . . . .	118
(Opcional) Conexión REXEC . . . . .	120
(Opcional) Script de shell REXEC/SSH . . . . .	121

## Parte 2. Información de Developer for System z . . . . . 123

### Capítulo 8. Mandatos de operador . . . 125

Iniciar (S) . . . . .	125
Supervisor de trabajos JES . . . . .	125
Daemon RSE . . . . .	126
Daemon de bloqueo . . . . .	126
Modificar (F) . . . . .	127
Supervisor de trabajos JES . . . . .	127
Daemon RSE . . . . .	128
Daemon de bloqueo . . . . .	131
Detener (P) . . . . .	131
Mensajes de consola . . . . .	132
Supervisor de trabajos JES . . . . .	132
Daemon RSE, servidor de agrupaciones de hebras RSE y daemon de bloqueo . . . . .	132

Cómo leer un diagrama de sintaxis . . . . .	134
Símbolos . . . . .	134
Operandos . . . . .	134
Ejemplo de sintaxis . . . . .	135
Caracteres no alfanuméricos y espacios en blanco . . . . .	135
Seleccionar más de un operando . . . . .	135
Longitud superior a una línea . . . . .	135
Fragmentos de sintaxis . . . . .	135

## Capítulo 9. Resolución de problemas de configuración . . . . . 137

Anotar y configurar el análisis mediante FEKLOGS	137
Archivos de anotaciones . . . . .	138
Anotaciones del Supervisor de trabajos JES . . . . .	139
Anotaciones del daemon de bloqueo . . . . .	139
Daemon RSE y anotaciones de la agrupaciones de hebras . . . . .	139
Anotaciones de usuario de RSE . . . . .	140
Anotaciones de Integración del analizador de errores . . . . .	141
Anotaciones de Integración del gestor de archivos . . . . .	141
Anotaciones de SCLM Developer Toolkit . . . . .	142
Anotaciones de CARMA . . . . .	142
Anotaciones de transacción APPC (servicio de mandatos TSO) . . . . .	142
Anotaciones de prueba IVP de fekfivpi . . . . .	143
Anotaciones de prueba IVP de fekfivps . . . . .	143
Archivos de vuelco . . . . .	143
Vuelcos de MVS . . . . .	143
Vuelcos de Java . . . . .	144
Ubicaciones de vuelcos de z/OS UNIX . . . . .	145
Rastreo . . . . .	146
Rastreo del Supervisor de trabajos JES . . . . .	146
Rastreo de RSE . . . . .	146
Rastreo del daemon de bloqueo . . . . .	147
Rastreo de CARMA . . . . .	147
Rastreo de información de retorno de errores . . . . .	147
Bits de permiso de z/OS UNIX . . . . .	148
Atributo del sistema de archivos SETUID . . . . .	148
Autorización de control de programa . . . . .	149
Autorización de APF . . . . .	150
Sticky bit . . . . .	151
Puertos TCP/IP reservados . . . . .	151
Tamaño del espacio de direcciones . . . . .	153
Requisitos de JCL de inicio . . . . .	153
Limitaciones establecidas en SYS1.PARMLIB(BPXPRMxx) . . . . .	153
Limitaciones almacenadas en el perfil de seguridad . . . . .	153
Limitaciones aplicadas por la rutinas de salida del sistema . . . . .	153
Limitaciones para el direccionamiento de 64 bits . . . . .	154
Transacción APPC y el servicio de mandatos TSO . . . . .	154
Información miscelánea . . . . .	156
Límites del sistema . . . . .	156
Problemas conocidos de los requisitos . . . . .	156
Emulador de conexión de host . . . . .	157

## Capítulo 10. Consideraciones relativas a la seguridad . . . . . 159

Métodos de autenticación . . . . .	159
ID de usuario y contraseña . . . . .	160
ID de usuario y contraseña para una sola vez . . . . .	160
Certificado X.509 . . . . .	160
Autenticación del Supervisor de trabajos JES . . . . .	160
Seguridad de conexión . . . . .	161
Limitar la comunicación externa a puertos especificados . . . . .	161
Cifrado de comunicaciones mediante SSL . . . . .	161
Comprobación de Puerto de entrada . . . . .	162
Puertos TCP/IP . . . . .	162
Comunicación externa . . . . .	162
Comunicación interna . . . . .	163
Puertos CARMA y TCP/IP . . . . .	164
Utilizar PassTickets . . . . .	164
Anotaciones de auditoría . . . . .	165
Control de auditoría . . . . .	165
Datos de auditoría . . . . .	165
Seguridad de JES . . . . .	166
Acciones en trabajos - limitaciones de destino . . . . .	166
Acciones en trabajos - limitaciones de ejecución . . . . .	168
Acceso a los archivos de spool . . . . .	169
Comunicación cifrada con SSL . . . . .	170
Autenticación de cliente mediante certificados X.509 . . . . .	171
Validación de la autoridad certificadora (CA) (Opcional) Consulta en una lista de certificados revocados (CRL) . . . . .	172
Autenticación del software de seguridad . . . . .	172
Autenticación del daemon RSE . . . . .	174
Comprobación de puerto de entrada (POE) . . . . .	174
Seguridad de CICSTS . . . . .	175
Repositorio CRD . . . . .	175
Transacciones CICS . . . . .	175
Comunicación cifrada con SSL . . . . .	175
Seguridad de SCLM . . . . .	175
Archivos de configuración de Developer for System z . . . . .	176
Rastreo del daemon de bloqueo - FEJJCNFG . . . . .	176
RSE - rsed.envvars . . . . .	176
RSE - ssl.properties . . . . .	177
Definiciones de seguridad . . . . .	177
Requisitos y lista de comprobación . . . . .	178
Activar valores y clases de seguridad . . . . .	179
Definir un segmento OMVS para usuarios de Developer for System z . . . . .	180
Definir perfiles de conjunto de datos . . . . .	180
Definir las tareas iniciadas de Developer for System z . . . . .	183
Definir la seguridad de mandatos JES . . . . .	184
Definir RSE como servidor z/OS UNIX seguro . . . . .	186
Definir las bibliotecas controladas por programa MVS para RSE . . . . .	186
Definir la protección de aplicaciones para RSE . . . . .	187
Definir el soporte de PassTicket para RSE . . . . .	187
Definir los archivos controlados por programa z/OS UNIX para RSE . . . . .	188
Verificar valores de seguridad . . . . .	189

## Capítulo 11. Qué es Developer for System z . . . . . 191

Visión general de los componentes . . . . .	191
RSE como aplicación Java . . . . .	193
Propietarios de tareas. . . . .	194
Flujo de conexión . . . . .	196
Daemon de bloqueo . . . . .	198
Liberar un bloqueo . . . . .	199
Estructura de directorios de z/OS UNIX . . . . .	200
Privilegios de actualización para usuarios no administradores del sistema . . . . .	201

## Capítulo 12. Consideraciones de WLM 203

Clasificación de carga de trabajo . . . . .	203
Reglas de clasificación . . . . .	204
Establecimiento de objetivos . . . . .	205
Consideraciones para la selección de objetivos . . . . .	206
STC . . . . .	207
OMVS. . . . .	208
JES . . . . .	209
ASCH . . . . .	210
CICS . . . . .	210

## Capítulo 13. Consideraciones acerca de los ajustes . . . . . 213

Uso de recursos . . . . .	213
Visión general . . . . .	214
Recuento de espacios de direcciones. . . . .	215
Recuento de procesos. . . . .	218
Recuento de hebras . . . . .	221
Uso de almacenamiento . . . . .	224
Límite de tamaño de almacenamiento dinámico Java . . . . .	224
Límite de tamaño del espacio de direcciones . . . . .	225
Directrices de estimación de tamaño. . . . .	226
Análisis del uso de almacenamiento de ejemplo . . . . .	226
Uso de espacio del sistema de archivos de z/OS UNIX . . . . .	230
Definiciones de recursos clave. . . . .	233
/etc/rdz/rsed.envvars . . . . .	233
SYS1.PARMLIB(BPXPRMxx) . . . . .	234
Definiciones de varios recursos . . . . .	236
Tarjeta EXEC del servidor JCL. . . . .	236
FEK.#CUST.PARMLIB(FEJJCNF) . . . . .	237
SYS1.PARMLIB(IEASYSxx) . . . . .	237
SYS1.PARMLIB(IVTPRMxx) . . . . .	237
SYS1.PARMLIB(ASCHPMxx) . . . . .	237
Supervisión . . . . .	238
Supervisar RSE. . . . .	238
Supervisar z/OS UNIX . . . . .	239
Supervisar la red . . . . .	241
Supervisar los sistemas de archivos de z/OS UNIX . . . . .	241
Configuración de ejemplo . . . . .	241
Recuento de agrupaciones de hebras . . . . .	242
Determinar los límites mínimos . . . . .	242
Definir límites . . . . .	242
Utilización de recursos de supervisor . . . . .	244

## Capítulo 14. Consideraciones sobre el rendimiento . . . . . 245

Utilizar sistemas de archivos zFS . . . . .	245
Evitar el uso de STEPLIB . . . . .	245
Mejorar el acceso a las bibliotecas del sistema . . . . .	245
Bibliotecas de tiempo de ejecución de Language Environment (LE) . . . . .	246
Desarrollo de aplicaciones . . . . .	246
Mejorar el rendimiento de la comprobación de seguridad . . . . .	247
Gestión de cargas de trabajo (WLM). . . . .	247
Memoria dinámica Java de tamaño fijo . . . . .	247
Opción -Xquickstart de Java . . . . .	248
Compartimiento de clases entre las JVM . . . . .	248
Habilitar el compartimiento de clases . . . . .	249
Límites de tamaño de la memoria caché . . . . .	249
Seguridad de la memoria caché . . . . .	249
SYS1.PARMLIB(BPXPRMxx) . . . . .	249
Espacio de disco . . . . .	250
Utilidades para la gestión de cachés. . . . .	250

## Capítulo 15. consideraciones de CICSTS . . . . . 253

RESTful versus Servicio Web . . . . .	254
Comparación entre regiones de conexión primarias y no primarias . . . . .	254
Anotaciones de instalación de recursos CICS . . . . .	255
Seguridad del Gestor de despliegue de aplicaciones . . . . .	255
Seguridad del repositorio de CRD . . . . .	255
seguridad de conducto . . . . .	255
Seguridad de transacciones. . . . .	255
Comunicación cifrada con SSL. . . . .	256
Seguridad de recursos . . . . .	257
Programa de utilidad administrativo . . . . .	257
Notas de migración del programa de utilidad administrativo . . . . .	261
Mensajes del programa de utilidad administrativo . . . . .	262

## Capítulo 16. Personalizar el entorno TSO . . . . . 265

El servicio de mandatos TSO . . . . .	265
Métodos de acceso . . . . .	265
Utilizar el método de acceso de Pasarela de cliente TSO/ISPF . . . . .	266
Personalización básica – ISPF.conf . . . . .	266
Avanzado – Utilizar perfiles ISPF existentes . . . . .	266
Avanzado – Utilizar un exec de asignación . . . . .	267
Avanzado – Utilizar varios execs de asignación . . . . .	267
Avanzado – Varios archivos ISPF.conf con varias configuraciones de Developer for System z . . . . .	267
Utilizar el método de acceso APPC . . . . .	268
Personalización básica – JCL de transacción APPC . . . . .	268
Avanzado – Utilizar perfiles ISPF existentes . . . . .	269
Avanzado – Utilizar un exec de asignación . . . . .	269
Avanzado – Varias transacciones APPC con varias configuraciones de Developer for System z . . . . .	270

## Capítulo 17. Ejecutar varias instancias 273

Configuración idéntica en todo un sysplex . . . . .	273
Archivos de configuración diferentes con idéntico nivel de software . . . . .	274
Todas las demás situaciones . . . . .	275

## Capítulo 18. Guía de migración. . . . 279

Consideraciones acerca de la migración. . . . .	279
Hacer copia de seguridad de archivos configurados con anterioridad. . . . .	279
Notas de migración de la versión 7.6.1 . . . . .	281
Migrar desde la versión 7.5 a la versión 7.6 . . . . .	282
IBM Rational Developer for System z, FMID HHOP760 . . . . .	282
Archivos configurables . . . . .	284
Migrar desde la versión 7.1 a la versión 7.5 . . . . .	288
IBM Rational Developer for System z, FMID HHOP750 . . . . .	288
Archivos configurables . . . . .	289
Migrar desde la versión 7.0 a la versión 7.1 . . . . .	291
IBM Rational Developer for System z, FMID HHOP710 . . . . .	291
IBM CARMA (Common Access Repository Manager), FMID HCMA710 . . . . .	292
Archivos configurables . . . . .	292

## Apéndice A. Configurar SSL y autenticación de X.509 . . . . . 295

Decida dónde desea almacenar los certificados y claves privadas. . . . .	296
Crear un anillo de claves con RACF. . . . .	297
(Opcional) Uso de un certificado firmado . . . . .	298
Clonar la configuración RSE existente . . . . .	299
Actualizar rsed.envvars para habilitar la coexistencia . . . . .	299
Actualizar ssl.properties para habilitar la SSL. . . . .	299
Activar la SSL creando un daemon RSE nuevo . . . . .	300
Probar la conexión. . . . .	300
(Opcional) Añadir soporte de autorización al cliente de X.509. . . . .	303
(Opcional) Crear una base de datos de claves con gskkyman . . . . .	303
(Opcional) Crear un almacén de claves con keytool . . . . .	306

## Apéndice B. Configurar TCP/IP. . . . 309

Dependencia del nombre de host. . . . .	309
Qué son los resolvientes. . . . .	310
Qué es el orden de búsqueda de la información de configuración . . . . .	310
Orden de búsqueda utilizado en el entorno z/OS UNIX . . . . .	311
Archivos de configuración resolvientes base . . . . .	311
Tablas de conversión . . . . .	312
Tablas de hosts locales . . . . .	312
Cómo se aplica esta información de configuración a Developer for System z . . . . .	313
La dirección del host no se resuelve correctamente . . . . .	315

## Apéndice C. Configurar INETD . . . . 317

inetd.conf . . . . .	317
ETC.SERVICES. . . . .	318
Orden de búsqueda utilizado en el entorno z/OS UNIX . . . . .	319
Orden de búsqueda utilizado en el entorno MVS nativo . . . . .	319
Definiciones de puertos en PROFILE.TCPIP . . . . .	320
/etc/inetd.pid . . . . .	321
Arranque. . . . .	321
/etc/rc . . . . .	321
/etc/inittab . . . . .	321
BPXBATCH . . . . .	322
Sesión de shell . . . . .	322
Seguridad . . . . .	323
Requisitos de Developer for System z . . . . .	324
INETD . . . . .	324
REXEC (o SSH). . . . .	324

## Apéndice D. Configurar APPC . . . . 325

VSAM. . . . .	325
VTAM. . . . .	326
SYS1.PARMLIB(APPCCPMxx) . . . . .	327
SYS1.PARMLIB(ASCHPMxx) . . . . .	328
Activar los cambios de APPC . . . . .	329
Definir la transacción del servicio de mandatos TSO . . . . .	329
(Opcional) Opciones de configuración alternativas . . . . .	330
Nombre de transacción alternativo . . . . .	330
Varias LU . . . . .	330
Seguridad de LU . . . . .	330

## Apéndice E. Requisitos . . . . . 333

Prerrequisitos del host z/OS . . . . .	333
z/OS . . . . .	333
SMP/E . . . . .	334
SDK for z/OS Java 2 Technology Edition . . . . .	335
Correquisitos de host z/OS. . . . .	335
z/OS . . . . .	336
COBOL compiler . . . . .	338
PL/I compiler . . . . .	338
Debug Tool for z/OS. . . . .	339
CICS Transaction Server. . . . .	339
IMS . . . . .	340
DB2 for z/OS . . . . .	341
Rational Team Concert para System z . . . . .	341
Gestor de archivos. . . . .	342
Analizador de errores . . . . .	342
REXX . . . . .	342
Ported tools . . . . .	343
Ant. . . . .	343
Endevor . . . . .	343

## Bibliografía . . . . . 345

Publicaciones a las que se hace referencia . . . . .	345
Publicaciones informativas . . . . .	347

## Glosario . . . . . 349

## Avisos de documentación de IBM Rational Developer for System z . . . 353

Licencia de Copyright . . . . .	354
Reconocimientos de marcas registradas. . . . .	355

<b>Índice. . . . .</b>	<b>357</b>
------------------------	------------

## Figuras

1.	JMON - Tarea iniciada del Supervisor de trabajos JES . . . . .	22	30.	FMIEXT.properties – archivo de configuración del gestor de archivos. . . . .	101
2.	RSED - Tarea iniciada del daemon RSE . . . . .	23	31.	uchars.settings - Configuración de caracteres no editables . . . . .	102
3.	LOCKD - Tarea iniciada del daemon de bloqueo . . . . .	23	32.	REXX para paneles ISPF APPC . . . . .	105
4.	RSED - Inicio alternativo de daemon RSE . . . . .	24	33.	Mandato de operador START JMON . . . . .	125
5.	rsed.stdin.sh - inicio alternativo de daemon RSE . . . . .	24	34.	Mandato de operador START RSED . . . . .	126
6.	FEJCNFG, archivo de configuración del supervisor de trabajos JES. . . . .	28	35.	Mandato de operador START LOCKD . . . . .	126
7.	rsed.envvars - Archivo de configuración de RSE . . . . .	34	36.	Mandato de operador MODIFY JMON . . . . .	127
8.	(continuación). . . . .	35	37.	Mandato de operador MODIFY RSED . . . . .	128
9.	ISPF.conf - Archivo de configuración de ISPF . . . . .	48	38.	Mandato de operador MODIFY LOCKD . . . . .	131
10.	CRASRV.properties – archivo de configuración de CARMA . . . . .	54	39.	Mandato de operador STOP . . . . .	131
11.	CRASRV.properties - Inicio de CARMA mediante sometimiento por lotes . . . . .	56	40.	Puertos TCP/IP. . . . .	162
12.	CRASUBMT - Inicio de CARMA mediante sometimiento por lotes. . . . .	57	41.	Visión general de los componentes . . . . .	191
13.	CRASRV.properties - Inicio alternativo de CARMA con *CRASTART. . . . .	58	42.	RSE como aplicación Java . . . . .	193
14.	crastart.conf - Inicio alternativo de CARMA con *CRASTART. . . . .	59	43.	Propietarios de tareas. . . . .	195
15.	CRASRV.properties - Inicio alternativo de CARMA con *ISPF . . . . .	61	44.	Flujo de conexión . . . . .	196
16.	ISPF.conf - Inicio alternativo de CARMA con *ISPF. . . . .	62	45.	Flujo de daemon de bloqueo . . . . .	198
17.	Figura x1. CRASRV.properties - Inicio de CA Endeavor SCM RAM mediante el sometimiento por lotes . . . . .	65	46.	Estructura de directorios de z/OS UNIX . . . . .	200
18.	Figura x2. CRASUBMT - Inicio de CA Endeavor SCM RAM mediante el sometimiento por lotes . . . . .	66	47.	Clasificación de WLM. . . . .	203
19.	Figura x3. CRASRV.properties - Inicio de CA Endeavor SCM RAM mediante CRASTART . . . . .	67	48.	Número máximo de espacios de direcciones . . . . .	217
20.	crastart.conf - Inicio de CA Endeavor SCM RAM mediante CRASTART . . . . .	68	49.	Número de espacios de direcciones por cliente . . . . .	218
21.	Actualizaciones de ISPF.conf para SCLMDT . . . . .	83	50.	Número máximo de procesos . . . . .	219
22.	Actualizaciones de rsed.envvars para SCLMDT . . . . .	84	51.	Número de procesos por cliente . . . . .	220
23.	FLM02LST - JCL de configuración de la conversión de nombres largos/abreviados . . . . .	85	52.	Número máximo de hebras de la agrupación de hebras RSE . . . . .	223
24.	ELAXMSAM - Tarea de procedimiento almacenado DB2. . . . .	90	53.	Número máximo de hebras del Supervisor de trabajos JES . . . . .	223
25.	ELAXMJCL – Definición de procedimiento almacenado DB2. . . . .	91	54.	Uso de recursos con 5 inicios de sesión . . . . .	227
26.	ssl.properties – Archivo de configuración SSL . . . . .	95	55.	Uso de recursos con 5 inicios de sesión (continuación) . . . . .	228
27.	rscomm.properties – archivo de configuración de anotaciones . . . . .	97	56.	Uso de recursos al editar un miembro PDS . . . . .	229
28.	propertiescfg.properties - archivo de configuración de grupos de propiedades basadas en host . . . . .	99	57.	uso de espacio del sistema de archivos de z/OS UNIX . . . . .	231
29.	projectcfg.properties – archivo de configuración de proyectos basados en host . . . . .	100	58.	Utilización de recursos de configuración de ejemplo . . . . .	244
			59.	ADNJSPAU - programa de utilidad administrativo de CICSTS . . . . .	259
			60.	FEKAPPCC - crear una segunda transacción APPC . . . . .	271
			61.	RSESSL - Trabajos de usuario del daemon RSE para SSL . . . . .	300
			62.	Diálogo Importar certificado de host. . . . .	301
			63.	Diálogo Preferencias - SSL . . . . .	302
			64.	JCL de arranque de INETD . . . . .	322
			65.	JCL para crear los VSAM de APPC . . . . .	326
			66.	SYS1.SAMPLIB(ATBAPPL) . . . . .	327
			67.	SYS1.PARMLIB(APPCPMxx) . . . . .	327
			68.	SYS1.PARMLIB(ASCHPMxx) . . . . .	329



# Tablas

1.	Recursos necesarios . . . . .	6	25.	Mecanismos de almacenamiento de certificados de SSL. . . . .	170
2.	Recursos opcionales . . . . .	6	26.	Variables de configuración de seguridad	178
3.	Administradores necesarios para tareas obligatorias . . . . .	7	27.	Mandatos de operador del Supervisor de trabajos JES2. . . . .	185
4.	Administradores necesarios para tareas opcionales . . . . .	8	28.	Mandatos de operador del Supervisor de trabajos JES3. . . . .	185
5.	Lista de comprobación del cliente - Partes obligatorias . . . . .	13	29.	Subsistemas de punto de entrada de WLM	204
6.	Lista de comprobación del cliente - Partes opcionales . . . . .	13	30.	Calificadores de trabajo de WLM . . . . .	205
7.	Procedimientos ELAXF* de ejemplo . . . . .	25	31.	Cargas de trabajo WLM . . . . .	206
8.	Lista de comprobación de calificadores de alto nivel de ELAXF*. . . . .	26	32.	Cargas de trabajo WLM - STC . . . . .	207
9.	Matriz de permisos de mandato LIMIT_COMMANDS . . . . .	30	33.	Cargas de trabajo WLM - OMVS . . . . .	208
10.	variables de crastart.conf . . . . .	59	34.	Cargas de trabajo WLM - JES . . . . .	209
11.	ID de transacción predeterminados del servidor CRD . . . . .	76	35.	Cargas de trabajo WLM - ASCH . . . . .	210
12.	ID de transacción predeterminados del servidor CRD . . . . .	77	36.	Cargas de trabajo WLM - CICS . . . . .	210
13.	Lista de comprobación del administrador de SCLM . . . . .	87	37.	Uso de recursos comunes . . . . .	214
14.	Mecanismos de almacenamiento de certificados de SSL . . . . .	94	38.	Uso de recursos requisito específicos del usuario . . . . .	214
15.	Tipos de almacenes de claves válidos . . . . .	96	39.	Uso de recursos específicos del usuario	215
16.	Lista de comprobación para la transacción APPC . . . . .	105	40.	Recuento de espacios de direcciones . . . . .	215
17.	IVPs para servicios. . . . .	112	41.	Límites de espacios de direcciones . . . . .	218
18.	Estado de error de la agrupación de hebras	129	42.	Recuento de procesos . . . . .	218
19.	Mensajes de consola de RSE . . . . .	132	43.	Límites de procesos . . . . .	221
20.	Variables de JAVA_DUMP_TDUMP_PATTERN. . . . .	144	44.	Recuento de hebras . . . . .	221
21.	Mandatos de la consola del supervisor de trabajos JES . . . . .	167	45.	Límites de hebras . . . . .	224
22.	Matriz de permisos de mandato LIMIT_COMMANDS . . . . .	167	46.	Directivas de salidas de anotaciones . . . . .	232
23.	Perfiles JESSPOOL ampliados . . . . .	167	47.	Personalizaciones de la versión 7.6 . . . . .	284
24.	matriz de permisos de examen de LIMIT_VIEW . . . . .	169	48.	Personalizaciones de la versión 7.5 . . . . .	289
			49.	Personalizaciones de la versión 7.1 . . . . .	293
			50.	Mecanismos de almacenamiento de certificados de SSL. . . . .	296
			51.	Definiciones locales disponibles para el resolvente . . . . .	315
			52.	Publicaciones a las que se hace referencia	345
			53.	Sitios Web a los que se hace referencia	347
			54.	Publicaciones informativas . . . . .	347



---

## Acerca de este documento

En este documento se describe la configuración de las funciones de IBM Rational Developer for System z. Incluye instrucciones que explican cómo configurar IBM Rational Developer for System z Versión 7.6.1 en su sistema de hospedaje z/OS.

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

- *IBM Rational Developer for System z* se denominará *Developer for System z*.
- *Common Access Repository Manager* se denominará *CARMA*.
- *Software Configuration and Library Manager Developer Toolkit* se denominará *SCLM Developer Toolkit*, cuya abreviatura es *SCLMDT*.
- *z/OS UNIX® System Services* se denomina *z/OS UNIX*.
- *Customer Information Control System Transaction Server* se denomina *CICSTS* y se abrevia como *CICS*.

En el caso de releases anteriores, incluidos los de IBM WebSphere Developer for System z, IBM WebSphere Developer for zSeries e IBM® 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 dirigido este documento

Este documento está destinado a los programadores de sistemas que instalan y configuran IBM Rational Developer for System z Versión 7.6.1, FMID HHOP760, en su sistema host z/OS.

Lista con detalle los diversos pasos necesarios para realizar una configuración completa del producto, incluidos algunos escenarios que no son predeterminados. Para utilizar esta documentación, debe estar familiarizado con z/OS UNIX System Services y con los sistemas de hospedaje MVS.

---

## Resumen de cambios

En esta sección se resumen los cambios de la *Guía de configuración de host de Rational Developer for System z Versión 7.6.1*, SC11-3660-04 (actualizado en mayo de 2010).

Los cambios técnicos o las adiciones al texto y las ilustraciones se indican mediante una línea vertical situada a la izquierda del cambio.

Este documento contiene la información presentada anteriormente en la *Guía de configuración de host de Rational Developer for System z Versión 7.6*, SC11-3660-03.

Información nueva:

- Se incorporan las correcciones y la información adicional presentada en *En las Notas de release de configuración de host de Rational Developer for System z v7.6*.
- Notas de migración específicas de la versión 7.6.1. Consulte “Notas de migración de la versión 7.6.1” en la página 281.
- Se ha añadido una visión general del documento. Consulte “Descripción del contenido del documento” en la página xiv.

- Soporte de Java™ de 64 bits. Consulte Apéndice E, “Requisitos”, en la página 333.
- Configuración del producto a través de paneles ISPF. Consulte “Consideraciones relativas a la preconfiguración” en la página 9.
- Directivas nuevas en rsed.envvars. Consulte “rsed.envvars, archivo de configuración de RSE” en la página 32.
- Miembros proclib nuevos. Consulte “Procedimientos de construcción remota ELAXF\*” en la página 25.
- Diseño VSAM de CARMA nuevo. Consulte “Notas de migración de VSAM de CARMA” en la página 53.
- Soporte de varios RAM de CARMA. Consulte “(Opcional) Soportar varios RAM” en la página 69.
- Opciones nuevas del Gestor de despliegue de aplicaciones. Consulte “Programa de utilidad administrativo” en la página 257.
- Diseño VSAM del Gestor de despliegue de aplicaciones nuevo. Vea: “Notas de migración del programa de utilidad administrativo” en la página 261
- Mandatos de operador nuevos. Consulte Capítulo 8, “Mandatos de operador”, en la página 125.
- Mensajes de consola nuevos. Vea: “Mensajes de consola” en la página 132
- Información de gestión de cargas de trabajo. Consulte Capítulo 12, “Consideraciones de WLM”, en la página 203.

---

## Descripción del contenido del documento

En esta sección se resume la información presentada en este documento.

### Planificación

Utilice la información de este capítulo, para planificar la instalación y el despliegue de Developer for System z.

### Personalización básica

Los pasos de configuración que se indican a continuación corresponden a una configuración básica de Developer for System z:

- Configuración de la personalización
- Cambios de PARMLIB
- Cambios de PROCLIB
- Definiciones de seguridad
- FEJCNFG, archivo de configuración del supervisor de trabajos JES
- rsed.envvars, archivo de configuración de RSE
- ISPF.conf, archivo de configuración de la Pasarela de cliente TSO/ISPF de ISPF

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

CARMA (Common Access Repository Manager) es una ayuda de productividad para los desarrolladores que crean RAM (Repository Access Managers). Un RAM es una interfaz de programación de aplicaciones (API) para SCM (Software Configuration Managers) basados en z/OS.

A su vez, las aplicaciones escritas por usuario pueden iniciar un servidor CARMA que cargue los RAM y suministre una interfaz estándar para acceder al SCM.

La interfaz de IBM® Rational® Developer for System z con el Gestor de configuraciones de software de CA Endevor® proporciona a los clientes de Developer for System z acceso directo a CA Endevor® SCM.

## **(Opcional) Gestor de despliegue de aplicaciones (ADM)**

Developer for System z utiliza determinadas funciones del Gestor de despliegue de aplicaciones como procedimiento de despliegue común para varios componentes. La personalización opcional permite tener más características de Application Deployment Manager y puede añadir los servicios siguientes a Developer for System z:

- El Explorador CICS de IBM proporciona una infraestructura basada en Eclipse para poder visualizar y manipular recursos CICS, además de permitir una mayor integración entre las herramientas CICS.
- El cliente y el servidor de Definición de recursos de CICS (CRD) proporcionan las funciones siguientes:
  - Editor de Definiciones de recursos CICS
  - Permitir a los desarrolladores de aplicaciones definir los recursos CICS de manera segura, controlada y limitada.
  - 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.
  - Ayudas de desarrollo CICS variadas
  - Ayudas de desarrollo de servicio web CICS variadas

## **(Opcional) SCLM Developer Toolkit**

SCLM Developer Toolkit suministra las herramientas necesarias para ampliar las prestaciones de SCLM para el cliente. SCLM en sí es un gestor de código fuente basado en host-based que se suministra como parte de ISPF.

SCLM Developer Toolkit contiene un plug-in de Eclipse-based 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 el SCLM del sistema central, incluidas las tareas de construir, ensamblar y desplegar el código J2EE desde el sistema central.

## **(Opcional) Otras tareas de personalización**

Esta sección combina diversas tareas de personalización opcionales. Siga las instrucciones de la sección adecuada para configurar el servicio que desee.

- procedimiento almacenado DB2
- Soporte de Enterprise Service Tools (EST)
- Soporte de idiomas bidireccionales CICS
- Diagnóstico de mensajes de error de IRZ
- Cifrado SSL de RSE
- Rastreo RSE
- Grupos de propiedades basadas en Host
- Proyectos basados en host
- Integración del gestor de archivos
- Caracteres no editables

- Utilizar REXEC (o SSH)
- Transacción APPC para el servicio de mandatos TSO
- Borrado de WORKAREA

## Verificación de la instalación

Después de completar la personalización del producto, puede utilizar los Programas de verificación de la instalación (IVP) descritos en este capítulo para verificar la configuración satisfactoria de componentes de productos clave.

## Mandatos de operador

Este capítulo proporciona una visión general de los mandatos del operador (o la consola) disponibles para Developer for System z.

## Resolución de problemas de configuración

Este capítulo se propone ayudarle a resolver algunos problemas comunes que pueden surgir durante la configuración de Developer for System z, y tiene las secciones siguientes:

- Anotar y configurar el análisis mediante FEKLOGS
- Archivos de anotaciones
- Archivos de vuelco
- Rastrear
- Bites de permiso de z/OS UNIX
- Puertos TCP/IP reservados
- Tamaño del espacio de direcciones
- Transacción APPC y el servicio de mandatos TSO
- Información variada

## Consideraciones relativas a la seguridad

Developer for System z proporciona a los usuarios acceso al sistema central en una estación de trabajo que no es del sistema central. Algunos aspectos importantes de la configuración del producto son: validar las solicitudes de conexión, proporcionar una comunicación segura entre el host y la estación de trabajo, y autorizar y auditar la actividad.

## Qué es Developer for System z

El host de Developer for System z está formado por varios componentes que interactúan para proporcionar al cliente acceso a los servicios y datos del host. Comprender el diseño de estos componentes puede ayudarle a tomar las decisiones de configuración correctas.

## Consideraciones de WLM

Al contrario que las aplicaciones z/OS tradicionales, Developer for System z no es una aplicación monolítica que se pueda identificar fácilmente para el Gestor de carga de trabajo (WLM). Developer for System z está formado por varios componentes que interactúan para proporcionar al cliente acceso a los servicios y datos del host. Algunos de estos servicios están activos en diferentes espacios de direcciones, lo que resulta en diferentes clasificaciones WLM.

## Consideraciones acerca de los ajustes

RSE (Explorador de Sistemas remotos) es el núcleo de Developer for System z. Para gestionar las conexiones y cargas de trabajo de los clientes, RSE está formado por un espacio de direcciones de daemon, que controla los espacios de direcciones de agrupaciones de hebras. El daemon actúa como punto focal a efectos de conexión y gestión, mientras que las agrupaciones de hebras procesan las cargas de trabajo del cliente.

Ello hace que RSE sea el destino principal para ajustar la configuración de Developer for System z. Sin embargo, mantener a cientos de usuarios, cada uno de los cuales utiliza 16 hebras o más, una cantidad determinada de almacenamiento y, posiblemente, one o más espacios de direcciones, requiere que la configuración de Developer for System z y z/OS.

## 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 for System z.

## consideraciones de CICSTS

Este capítulo contiene información útil para un administrador de CICS Transaction Server.

## Personalizar el entorno TSO

Este capítulo le ayuda a emular un procedimiento de inicio de sesión TSO añadiendo sentencias DD y conjuntos de datos al entorno TSO en Developer for System z.

## Ejecutar varias instancias

En algunas ocasiones le interesará tener múltiples instancias de Developer for 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 capítulo para planificar la coexistencia de distintas instancias de Developer for System z, y después podrá usar esta guía de configuración para personalizarlas.

## Guía de migración

En este apartado se resaltan los cambios de instalación y configuración comparados con los releases anteriores del producto. También se ofrecen algunas directrices generales para la migración a este release. Para obtener más información, consulte las secciones relacionadas de este manual.

## Configurar SSL y autenticación de X.509

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 o modificar una configuración existente. Este apéndice también facilita una configuración de ejemplo para admitir que los usuarios se autenticquen con un certificado X.509.

## **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 o modificar una configuración existente.

## **Configurar INETD**

Este apéndice se propone ayudarle a resolver algunos problemas comunes que pueden surgir al configurar INETD, o durante la tarea de comprobar o modificar una configuración existente. Developer for System z utiliza INETD para la función REXEC/SSH.

## **Configurar TCP/IP**

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 o modificar una configuración existente.

## **Requisitos**

Este apéndice enumera los prerrequisitos y correquisitos del host para esta versión de Developer for System z.

---

## **Parte 1. Personalización de Developer for System z**



---

## Capítulo 1. Planificación

Utilice la información de este capítulo, junto con la información de Apéndice E, “Requisitos”, en la página 333, para planificar la instalación y el desarrollo de Developer for System z. Se describen los siguientes temas:

- “Consideraciones acerca de la migración”
- “Consideraciones de planificación”
- “Consideraciones relativas a la preinstalación” en la página 4
- “Consideraciones relativas a la preconfiguración” en la página 9
- “Consideraciones relativas al predespliegue” en la página 12
- “Lista de comprobación del cliente” en la página 13

---

### Consideraciones acerca de la migración

Capítulo 18, “Guía de migración”, en la página 279 describe los cambios de instalación y configuración comparados con los releases anteriores del producto. Utilice esta información para planificar la migración al release actual de Developer for System z.

#### Notas:

1. Si es usted un usuario anterior de IBM Rational Developer for System z, IBM WebSphere Developer for System z, IBM WebSphere Developer for zSeries o IBM WebSphere Studio Enterprise Developer, le recomendamos que guarde los archivos personalizados relacionados ANTES de instalar IBM Rational Developer for System z Versión 7.6.1. Consulte Capítulo 18, “Guía de migración”, en la página 279 para obtener una visión general de los archivos que requieren personalización.
2. Lea la sección Capítulo 17, “Ejecutar varias instancias”, en la página 273 si tiene previsto ejecutar varias instancias de Developer for System z.

---

### Consideraciones de planificación

#### Visión general del producto

Developer for System z está formado por un cliente, instalado en el sistema personal del usuario, y un servidor, instalado en uno o varios hosts. Esta documentación se centra en el host como sistema z/OS. Sin embargo, también están soportados otros sistemas operativos como AIX y Linux® on System z.

El cliente proporcionar desarrolladores para un entorno de desarrollo basado en Eclipse que facilita al host una interfaz gráfica uniforme y que, entre otras cosas, puede descargar trabajo del host en el cliente, guardando los recursos en el host.

La parte del host está formada por varias tareas permanentemente activas y por tareas que se inician con un fin. Estas tareas permiten al cliente trabajar con los distintos componentes del host z/OS, tales como conjuntos de datos MVS, mandatos TSO, archivos y comandos z/OS UNIX, sometimientos de trabajos y salidas de trabajos.

Developer for System z también puede interactuar con subsistemas y otro software de aplicaciones en el host, tales como CICS, Debug Tool y Software Configuration

Managers (SCM), si Developer for system z está configurado para ello, y si los productos (correquisito) están disponibles.

Consulte Capítulo 11, “Qué es Developer for System z”, en la página 191 para obtener los conocimientos básicos del diseño de Developer for System z.

Consulte el sitio web de Developer for System z, <http://www-01.ibm.com/software/awdtools/rdz/>, o a su representante de IBM local para obtener más información acerca de las funciones que ofrece Developer for System z.

## Requisitos de conocimientos técnicos

Los conocimientos técnicos de SMP/E que se necesitan para la instalación de un host de Developer for System z son mínimos.

Para la configuración de Developer para System z se necesita más que los conocimientos técnicos y permisos normales, de manera que es posible que sea necesaria una ayuda por parte de otras personas. Tabla 3 en la página 7 y Tabla 4 en la página 8 enumeran a los administradores necesarios para las tareas de personalización tanto obligatorias como opcionales.

## Requisitos de tiempo

El tiempo que se necesita par instalar y configurar los componentes de host de Developer for System z depende de varios factores, tales como:

- la configuración actual de z/OS UNIX y TCP/IP
- la disponibilidad del software y el mantenimiento prerequisite
- si los segmentos OMVS están o no definidos para los usuarios de Developer for System z
- la disponibilidad de un usuario, que haya instalado satisfactoriamente el cliente, para probar la instalación y notificar los problemas que puedan producirse

La experiencia ha demostrado que para completar el proceso de instalación y configuración del host de Developer for System z se necesita entre uno y cuatro días. Este requisito de tiempo se calcula para una instalación limpia realizada por un programador de sistemas con experiencia. Si se encuentran problemas, o si no se dispone de los conocimientos técnicos necesarios, la configuración podría llevar más tiempo.

---

## Consideraciones relativas a la preinstalación

Consulte la publicación *Directorio de programa para IBM Rational Developer for System z*, GI11-8627 (GI11-8298), para obtener instrucciones detalladas acerca de la instalación SMP/E del producto.

**Nota:** El sistema de archivos (HFS o zFS) en el que se instala Developer for System z debe estar montado con el bit de permiso SETUID (este el valor predeterminado del sistema). El hecho de montar el sistema de archivos con el parámetro NOSETUID impedirá que Developer for System z cree el entorno de seguridad del usuario y la solicitud de conexión del cliente fallará.

Consulte la sección Capítulo 17, “Ejecutar varias instancias”, en la página 273 si tiene previsto ejecutar varias instancias de Developer for System z.

## Elección de la configuración

Developer for System z ofrece una opción relativa a cómo acceder al servicio de mandatos TSO. La elección realizada aquí influye sobre la configuración necesaria de los requisitos previos. Debe seleccionarse y configurarse uno de los métodos siguientes:

- Pasarela de cliente TSO/ISPF de ISPF, que requiere un nivel de servicio mínimo de ISPF. Este es el método predeterminado utilizado en los ejemplos suministrados.
- Una transacción APPC (como en los releases anteriores a la versión 7.1)

**Nota:** SCLM Developer Toolkit y, opcionalmente, un método alternativo de inicio para Common Access Repository Manager (CARMA) también utilizan la Pasarela de cliente TSO/ISPF de ISPF.

## Productos requisito

Apéndice E, “Requisitos”, en la página 333 tiene una lista del software prerequisite que hay que instalar y debe estar operativo para que Developer for System z funcione. También hay una lista del software corequisite para poder utilizar las características específicas de Developer for System z. Estos requisitos deben estar instalados y ser operativos en tiempo de ejecución para que la correspondiente característica funcione como es debido.

Consulte los *Requisitos previos de Rational Developer for System z* (SC23-7659) en la biblioteca en línea del Developer for System z en <http://www-01.ibm.com/software/awdtools/rdz/library/> si desea obtener una lista actualizada de los productos prerequisite y corequisite para su versión de Developer for System z. Planee por adelantado la disponibilidad de estos productos requisito, ya que puede tardar algún tiempo en función de las políticas del sitio. Los requisitos clave para una configuración básica son los siguientes:

- z/OS 1.8 o superior
- ISPF APAR OA29489 (Pasarela de cliente TSO/ISPF)
- Java 5.0 o superior

**Nota:** Es necesario aplicar el PTF para Developer for System z APAR PM07305 cuando se utiliza una versión de 64 bits de Java. El PTF está disponible a través de la página de servicio recomendada de Developer for System z, <http://www-01.ibm.com/support/docview.wss?rs=2294&context=SS2QJ2&uid=swg27006335>.

## Recursos necesarios

Developer for System z requiere la asignación de los recursos del sistema indicados en la sección Tabla 1 en la página 6. Los recursos listados en Tabla 2 en la página 6 son necesarios para los servicios opcionales. Prevea la disponibilidad de estos recursos ya que puede llevar algún tiempo conseguirlos, según las políticas de su sitio.

**Nota:** Developer for System z está formado por varias tareas que se comunican entre ellas y con el cliente. Estas tareas utilizan varios temporizadores para detectar la pérdida de comunicación con el interlocutor o interlocutores. Ello implica que pueden surgir emisiones de tiempo de espera excedido (debido a la falta de tiempo de CPU durante la ventana de tiempo de espera

excedido) en sistemas con una carga de CPU pesada o con valores de gestión de la carga de trabajo (WLM) incorrectos para Developer for system z.

*Tabla 1. Recursos necesarios*

Recurso	Valor predeterminado	Información
conjunto de datos autorizado APF	FEK.SFEKAUTH	"Autorizaciones APF en PROGxx" en la página 18
tarea iniciada	JMON, RSED y LOCKD	"Consideraciones sobre el servidor" en la página 10
puerto para uso exclusivo del host	6715	"FEJJCNFG, archivo de configuración del supervisor de trabajos JES" en la página 27
puerto para uso exclusivo del host	4036	"rsed.envvars, archivo de configuración de RSE" en la página 32
puerto para la comunicación cliente-host	4035	"Cambios de PROCLIB" en la página 21
rango de puertos para la comunicación cliente-host	se utiliza cualquier puerto disponible	"Definir el rango de puertos (PORTRANGE) disponibles para el servidor RSE" en la página 41
Definición de seguridad de aplicación	READ de acceso universal para FEKAPPL	"Definir la protección de aplicaciones para RSE" en la página 187
definiciones de seguridad PassTicket	no hay valor predeterminado	"Definir el soporte de PassTicket para RSE" en la página 187

*Tabla 2. Recursos opcionales*

Recurso	Valor predeterminado	Información
conjunto de datos LINKLIST	FEK.SFEKAUTH y FEK.SFEKLOAD	Capítulo 5, "(Opcional) SCLM Developer Toolkit", en la página 81
conjunto de datos LPA	FEK.SFEKLPA	Capítulo 3, "(Opcional) CARMA (Common Access Repository Manager)", en la página 51
rango de puertos para uso exclusivo del host	5227-5326 (100 puertos)	Capítulo 3, "(Opcional) CARMA (Common Access Repository Manager)", en la página 51
puertos para uso exclusivo del host	se utiliza cualquier puerto disponible	"(Opcional) Transacción APPC para el servicio de mandatos TSO" en la página 104
puerto para la comunicación cliente-host	no hay valor predeterminado	Capítulo 4, "(Opcional) Gestor de despliegue de aplicaciones (ADM)", en la página 73

Tabla 2. Recursos opcionales (continuación)

Recurso	Valor predeterminado	Información
Actualización CSD de CICS	varios valores	Capítulo 4, “(Opcional) Gestor de despliegue de aplicaciones (ADM)”, en la página 73
Actualización JCL de CICS	FEK.SFEKLOAD	<ul style="list-style-type: none"> <li>Capítulo 4, “(Opcional) Gestor de despliegue de aplicaciones (ADM)”, en la página 73</li> <li>“(Opcional) Soporte de idiomas bidireccionales CICS” en la página 92</li> </ul>

Para la configuración de Developer for System z se necesita más que los conocimientos técnicos y permisos normales, de manera que es posible que sea necesaria una mínima ayuda por parte de otras personas. Tabla 3 y Tabla 4 en la página 8 enumeran a los administradores necesarios para las tareas de personalización tanto obligatorias como opcionales.

Tabla 3. Administradores necesarios para tareas obligatorias

Administrador	Tarea	Información
Sistema	Son necesarias acciones típicas del programador de sistemas para todas las tareas de personalización	N/A
Seguridad	<ul style="list-style-type: none"> <li>Definir un segmento OMVS para usuarios de Developer for System z</li> <li>Definir perfiles de conjunto de datos</li> <li>Definir tareas iniciadas</li> <li>Definir la seguridad de mandatos del operador</li> <li>Definir perfiles de servidor z/OS UNIX</li> <li>Definir seguridad de aplicación</li> <li>Definir el soporte de PassTicket</li> <li>Definir conjuntos de datos controlados por programa</li> <li>Definir archivos z/OS UNIX controlados por programa</li> </ul>	Capítulo 10, “Consideraciones relativas a la seguridad”, en la página 159
TCP/IP	Definir nuevos puertos TCP/IP	“Puertos TCP/IP” en la página 162
WLM	Asignar objetivos de tarea iniciada a los servidores y los procesos hijo	Capítulo 12, “Consideraciones de WLM”, en la página 203

Tabla 4. Administradores necesarios para tareas opcionales

Administrador	Tarea	Información
Sistema	Son necesarias acciones típicas del programador de sistemas para todas las tareas de personalización	N/A
Seguridad	<ul style="list-style-type: none"> <li>Definir perfiles de conjunto de datos</li> <li>Definir conjuntos de datos controlados por programa</li> <li>Definir permiso para someter trabajos xxx*</li> <li>Definir seguridad de transacciones CICS</li> <li>Añadir certificado para SSL</li> <li>Definir el soporte de los certificados de cliente X.509</li> </ul>	<ul style="list-style-type: none"> <li>Capítulo 10, "Consideraciones relativas a la seguridad", en la página 159</li> <li>Capítulo 15, "consideraciones de CICSTS", en la página 253</li> <li>Apéndice A, "Configurar SSL y autenticación de X.509", en la página 295</li> </ul>
TCP/IP	Definir nuevos puertos TCP/IP	"Puertos TCP/IP" en la página 162
SCLM	<ul style="list-style-type: none"> <li>Definir conversores de lenguaje SCLM para soporte JAVA/J2EE</li> <li>Definir tipos SCLM para soporte JAVA/J2EE</li> </ul>	Capítulo 5, "(Opcional) SCLM Developer Toolkit", en la página 81
CICS TS	<ul style="list-style-type: none"> <li>Actualizar JCL de región CICS</li> <li>Actualizar CSD de región CICS</li> <li>Definir grupo CICS</li> <li>Definir nombres de transacción CICS</li> <li>Definir un programa en CICS</li> </ul>	<ul style="list-style-type: none"> <li>Capítulo 4, "(Opcional) Gestor de despliegue de aplicaciones (ADM)", en la página 73</li> <li>"(Opcional) Soporte de idiomas bidireccionales CICS" en la página 92</li> </ul>
DB2	Definir un procedimiento almacenado DB2	"(Opcional) Procedimiento almacenado DB2" en la página 89
WLM	<ul style="list-style-type: none"> <li>Asignar objetivos a un procedimiento almacenado DB2</li> <li>Asignar objetivos de tipo TSO a una transacción APPC</li> </ul>	<ul style="list-style-type: none"> <li>"(Opcional) Procedimiento almacenado DB2" en la página 89</li> <li>Capítulo 12, "Consideraciones de WLM", en la página 203</li> </ul>
APPC	Define una transacción APPC	"(Opcional) Transacción APPC para el servicio de mandatos TSO" en la página 104

---

## Consideraciones relativas a la preconfiguración

### Gestión de cargas de trabajo (WLM)

Al contrario que las aplicaciones z/OS tradicionales, Developer for System z no es una aplicación monolítica que se pueda identificar fácilmente para el Gestor de carga de trabajo (WLM). Developer for System z está formado por varios componentes que interactúan para proporcionar al cliente acceso a los servicios y datos del host. Consulte la sección Capítulo 12, “Consideraciones de WLM”, en la página 203 para planificar la configuración de WLM en consecuencia.

### Uso de recursos y límites del sistema

Cuando se está utilizando, Developer for System z utilizará un número variable de recursos del sistema, como espacios de direcciones, y procesos y hebras z/OS UNIX. La disponibilidad de estos recursos está limitada por varias definiciones de sistema. Consulte Capítulo 13, “Consideraciones acerca de los ajustes”, en la página 213 para estimar el uso de los recursos clave, de manera que pueda planificar acorde la configuración del sistema.

### Configuración necesaria de los productos 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 continuación se indican algunas tareas de personalización obligatorias que es fácil pasar por alto:

- Todos los usuarios de Developer for System z deben tener acceso de lectura y ejecución (READ y EXECUTE) sobre los directorio de Java.
- Todos los usuarios de Developer for System z deben tener acceso de lectura, grabación y ejecución (READ, WRITE y EXECUTE) sobre el directorio /tmp/.
- Las acciones remotas (basadas en host) para subproyectos z/OS UNIX requieren que la versión de REXEC o SSH de z/OS UNIX esté activa en el host.

### Consideraciones sobre los ID de usuario

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

- Acceso TSO (con un tamaño de región normal).

**Nota:** El usuario que ejecuta los programas de verificación de instalación (IVP) necesita un tamaño de región grande, ya que se ejecutarán funciones que necesitan mucha memoria (tales como Java). Debe establecer el tamaño de región en 131072 kilobytes (128 megabytes) o superior.

- Un segmento OMVS definido en el sistema de seguridad (por ejemplo, RACF), para el ID de usuario y para el grupo predeterminado.
  - El campo HOME debe hacer referencia a un directorio inicial asignado para el usuario (con acceso de escritura, lectura y ejecución, WRITE, READ y EXECUTE).
  - 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  
PROCUSERMAX= NONE  
THREADSMAX= NONE  
MMAPAREAMAX= 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 y archivos de instalación y configuración de Developer for System z cuyos valores predeterminados son /usr/lpp/rdz/\*, /etc/rdz/\* y /var/rdz/\*.
- Acceso de lectura, grabación y ejecución (READ, WRITE y EXECUTE) al directorio WORKAREA de Developer for System z, cuyo valor predeterminado es /var/rdz/WORKAREA.
- Acceso de lectura (READ) a los conjuntos de datos de instalación de Developer for System z, cuyo valor predeterminado es FEK.SFEK\*.

## Consideraciones sobre el servidor

Developer for System z consta de 3 servidores permanentemente activos, que pueden ser tareas iniciadas o trabajos de usuario. Estos servidores proporcionan por sí mismos los servicios solicitados o inician otros servidores (como hebras z/OS UNIX o trabajos de usuario) para suministrar el servicio.

- El Supervisor de trabajos JES (JMON) suministra todos los servicios relacionados con JES.
- El daemon de bloqueo (LOCKD) proporciona servicios de rastreo para bloqueos de conjuntos de datos.
- El Explorador de sistemas remotos (RSE) proporciona servicios del núcleo como los de conectar el cliente al host e iniciar otros servidores para servicios específicos. El RSE consta de 2 entidades lógicas:
  - El daemon RSE (RSED), que gestiona la configuración de conexiones y es responsable de la ejecución en modalidad de servidor único.
  - El servidor RSE, que maneja las solicitudes de clientes individuales.

El Supervisor de trabajos JES (JMON) suministra todos los servicios relacionados con JES.

- Los mecanismos de seguridad utilizados por el supervisor de trabajos JES se basan en que los conjuntos de datos en los que reside sean seguros. Esto implica que sólo los administradores del sistema que sean de confianza puedan actualizar las bibliotecas y archivos de configuración.

Explorador de sistemas remotos RSE es el componente de Developer for System z que proporciona servicios centrales como los de conectar el cliente al host.

- A partir de la versión 7.5, el RSE ya no es un proceso gestionado por INETD, sino una tarea iniciada.
- A partir de la versión 7.5, el servidor RSE utiliza un modelo de servidor único, mientras que, en las versiones anteriores, cada conexión cliente-host tenía un servidor RSE privado.
- El RSE da soporte a varios niveles de seguridad de comunicaciones:
  - La comunicación externa (cliente-host) puede limitarse a puertos especificados. Esta característica está inhabilitada por omisión.
  - La comunicación externa (cliente-host) puede cifrarse mediante SSL. Esta característica está inhabilitada por omisión.
  - Puede utilizarse la comprobación de puerto de entrada (POE) para permitir el acceso sólo a las direcciones TCP/IP de confianza. Esta característica está inhabilitada por omisión.
- El RSE también soporta varios métodos de autenticación de clientes:
  - ID de usuario y contraseña
  - ID de usuario y contraseña para una sola vez
  - certificado X.509
- Los mecanismos de seguridad utilizados por el RSE se basan en que el sistema de archivos en el que reside sea seguro. Esto implica que sólo los administradores del sistema que sean de confianza puedan actualizar las bibliotecas y archivos de configuración.

Como se describe en “Puertos TCP/IP” en la página 162, determinados 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 for System z tienen tráfico solo de host. A continuación se indican los puertos necesarios para una configuración básica de Developer for System z.

- Daemon RSE para la configuración de la comunicación cliente-host (mediante el protocolo tcp), puerto predeterminado 4035.
- Servidor RSE para la comunicación cliente-host (mediante el protocolo tcp). Por omisión, utiliza cualquier puerto disponible, pero puede estar limitado a un rango especificado.

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

## Método de configuración

A partir de la versión 7.6.1, Developer for System z proporciona un método alternativo, mediante una aplicación de panel ISPF, para configurar el lado de host del producto. Esto le permite elegir entre los métodos siguientes:

- Utilizar la aplicación de panel ISPF. Esta aplicación le guía a través de los pasos de personalización necesarios y de los pasos de personalización opcionales seleccionados. Para obtener más información, consulte el libro blanco *Host Configuration Utility*, disponible en la biblioteca de internet de Developer for System z, <http://www-306.ibm.com/software/awdtools/rdz/library/>.
- Utilizar la *Guía de inicio rápido de configuración de host*. Esto le lleva por los pasos de personalización necesarios. El ámbito de esta guía está limitado a una configuración básica.
- Utilizar la *Guía de configuración de host*. Esto le lleva por los pasos de personalización necesarios y todos los pasos de personalización opcionales

seleccionados. En esta guía se tratan todas las opciones configurables, incluyendo algunos casos prácticos no predeterminados

---

## Consideraciones relativas al predespliegue

Developer for System z permite 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, este archivo debe sustituir a su equivalente en las listas que figuran más abajo.

**Nota:** La lista que sigue no cubre las necesidades de despliegue del software prerequisite y corequisite.

- FEK.SFEKAUTH(\*)
- FEK.SFEKLOAD(\*)
- FEK.SFEKPROC(\*)
- FEK.#CUST.PARMLIB(\*)
- FEK.#CUST.PROCLIB(\*)
- /usr/lpp/rdz/\*
- /etc/rdz/\*
- /var/rdz/\* (sólo estructura de directorios)
- componentes opcionales:
  - FEK.SFEKLPA(\*)
  - FEK.#CUST.CNTL(\*)
  - definiciones, conjuntos de datos, archivos y directorios resultantes de los trabajos de personalización de FEK.#CUST.JCL

### Notas:

1. FEK y /usr/lpp/rdz son el calificador de alto nivel y la vía de acceso que se emplean durante la instalación del producto. FEK.#CUST, /etc/rdz y /var/rdz son las ubicaciones predeterminadas utilizadas durante la personalización del producto (consulte la sección “Configuración de la personalización” en la página 15 para obtener más información).
2. Debe instalar Developer for System z en un sistema de archivos privado (HFS o zFS) para desplegar fácilmente los componentes z/OS UNIX del producto.
3. Si no puede utilizar un sistema de archivos privado, debe utilizar una herramienta de archivado como el mandato tar de z/OS UNIX para transportar los directorios de z/OS UNIX de un sistema a otro. Eso permite conservar los atributos (como por ejemplo el control de programa) de los archivos y directorios de Developer for System z.

Consulte la publicación *UNIX System Services Command Reference* (SA22-7802) para obtener más información acerca de los siguientes mandatos de ejemplo para archivar y restaurar el directorio de instalación de Developer for System z.

- Archivar: cd /SYS1/usr/lpp/rdz; tar -cSf /u/userid/rdz.tar
- Restaurar: cd /SYS2/usr/lpp/rdz; tar -xSf /u/userid/rdz.tar

## Lista de comprobación del cliente

Los usuarios del cliente Developer for 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 estas listas de comprobación para reunir la información que necesite.

La lista de comprobación de Tabla 5 indica los resultados necesarios de los pasos de personalización obligatorios. Tabla 6 indica los resultados necesarios de los pasos de personalización opcionales.

*Tabla 5. Lista de comprobación del cliente - Partes obligatorias*

Personalización	Valor
Número de puerto del servidor del supervisor de trabajos JES (el valor predeterminado es 6715 ):  Consulte SERV_PORT en "FEJJCNFG, archivo de configuración del supervisor de trabajos JES" en la página 27.	
Número de puerto TCP/IP del daemon RSE (el valor predeterminado es 4035):  Consulte "Daemon RSE" en la página 22.	

*Tabla 6. Lista de comprobación del cliente - Partes opcionales*

Personalización	Valor
Ubicación de los procedimientos ELAXF* si no están en la biblioteca de procedimientos del sistema:  Vea la nota de JCLLIB en: "Procedimientos de construcción remota ELAXF*" en la página 25.	
Nombres de procedimiento o paso de los procedimientos ELAXF*, si se han cambiado  Vea la nota que indica cómo cambiarlos, en: "Procedimientos de construcción remota ELAXF*" en la página 25.	
Nombre de procedimiento almacenado DB2 (el valor predeterminado es ELAXMSAM):  Vea información sobre los procedimientos almacenados DB2, en: Capítulo 17, "Ejecutar varias instancias", en la página 273.	
Ubicación del procedimiento almacenado DB2 si no está en una biblioteca de procedimientos del sistema:  Vea: "(Opcional) Procedimiento almacenado DB2" en la página 89.	
(correquisito) Número de puerto TN3270 para el Emulador de conexión de host (valor predeterminado 23).  Consulte: Capítulo 10, "Consideraciones relativas a la seguridad", en la página 159	
(correquisito) Número de puerto REXEC o SSH (valor predeterminado 512 o 22, respectivamente):  Vea: "(Opcional) Utilizar REXEC (o SSH)" en la página 102.	

*Tabla 6. Lista de comprobación del cliente - Partes opcionales (continuación)*

<b>Personalización</b>	<b>Valor</b>
Ubicación del archivo <code>server.zseries</code> si se utiliza el método de conexión REXEC/SSH (valor predeterminado <code>/etc/rdz</code> ).  Vea: “(Opcional) Utilizar REXEC (o SSH)” en la página 102.	
Ubicación del JCL de <code>CRA#ASLM</code> para las asignaciones de conjuntos de datos RAM de SCLM de CARMA (valor predeterminado <code>FEK.#CUST.JCL</code> ):  Vea la nota de <code>CRA#ASLM</code> , en: “Activar los RAM de SCLM” en la página 63.	

---

## Capítulo 2. Personalización básica

Los pasos de configuración que se indican a continuación corresponden a una configuración básica de Developer for System z. Consulte los capítulos dedicados a los componentes opcionales para conocer sus requisitos de personalización.

---

### Requisitos y lista de comprobación

Necesitará ayuda de un administrador de seguridad y de un administrador de TCP/IP para realizar esta tarea de personalización, que requiere los siguientes recursos y tareas de personalización especiales:

- conjunto de datos autorizado APF
- Varias actualizaciones de PARMLIB
- Varias actualizaciones de software de seguridad
- Varios puertos TCP/IP para la comunicación interna y cliente-host

A fin de verificar la instalación y empezar a utilizar Developer for System z en su sitio, debe realizar las tareas siguientes. A menos que se indique de otro modo, todas las tareas son obligatorias.

1. Crear copias personalizables de ejemplos y crear el entorno de trabajo para Developer for system z. Para obtener más detalles, consulte “Configuración de la personalización”.
2. Actualizar los límites del sistema z/OS UNIX, iniciar las tareas iniciales, definir conjuntos de datos de LINKLIST y autorizado APF y, de forma opcional, los conjuntos de datos LPA. Para obtener más detalles, consulte “Cambios de PARMLIB” en la página 16.
3. Crear procedimientos de tareas iniciadas y procedimientos de compilación/enlace. Para obtener más detalles, consulte “Cambios de PROCLIB” en la página 21.
4. Actualizar las definiciones de seguridad. Para obtener más detalles, consulte “Definiciones de seguridad” en la página 26. Debe tener en cuenta y comprender cómo se utilizan PassTickets para establecer la seguridad de las hebras. Los detalles están en: “Utilizar PassTickets” en la página 164.
5. Archivos de configuración para la personalización de Developer for System z. Para obtener más detalles, consulte:
  - “FEJJCNFG, archivo de configuración del supervisor de trabajos JES” en la página 27
  - “rsed.envvars, archivo de configuración de RSE” en la página 32
  - “ISPF.conf, archivo de configuración de la Pasarela de cliente TSO/ISPF de ISPF” en la página 47

---

### Configuración de la personalización

Developer for System z se suministra con varios archivos de configuración y JCL de ejemplo. Para evitar la necesidad de sobrescribir las personalizaciones al aplicar el mantenimiento, debe copiar todos estos miembros y los archivos de z/OS UNIX en una ubicación diferente y personalizar la copia.

Algunas funciones de Developer for System z también requieren la existencia de determinados directorios en z/OS UNIX, que deben crearse durante la

personalización del producto. Para facilitar el proceso de instalación, se suministra un trabajo de ejemplo, FEKSETUP, destinado a crear las copias y los directorios necesarios.

Personalice y someta el miembro de ejemplo FEKSETUP del conjunto de datos FEK.SFEKSAMP para crear copias personalizables de los archivos y el JCL de configuración, y para crear los directorios z/OS UNIX necesarios. Los pasos de personalización necesarios se describen dentro del miembro.

Este trabajo realiza las tareas siguientes:

- Crea FEK.#CUST.PARMLIB y lo llena con archivos de configuración de ejemplo.
- Crea FEK.#CUST.PROCLIB y lo llena con miembros SYS1.PROCLIB de ejemplo.
- Crea FEK.#CUST.JCL y lo llena con JCL de configuración de ejemplo.
- Crea FEK.#CUST.CNTL y lo llena con scripts de inicio de servidor de ejemplo.
- Crea FEK.#CUST.ASM y lo llena con código fuente assembler de ejemplo.
- Crea FEK.#CUST.COBOLE y lo llena con código fuente COBOL de ejemplo.
- Crea /etc/rdz/\* y lo llena con archivos de configuración de ejemplo.
- Crea /var/rdz/\* como directorios de trabajo para diversas funciones de Developer for System z.

#### Notas:

1. A menos que se indique lo contrario, los pasos de configuración de esta publicación utilizan las ubicaciones de miembro/archivo creadas por el trabajo FEKSETUP. Los ejemplos originales, que no deben actualizarse, se encuentran en FEK.SFEKSAMP y /usr/lpp/rdz/samples/.
2. Si desea conservar todos los archivos de z/OS UNIX de Developer for System z en el mismo sistema de archivos (HFS o zFS), pero también desea colocar los archivos de configuración en /etc/rdz, puede utilizar enlaces simbólicos para resolver este problema. Los siguientes mandatos de ejemplo de z/OS UNIX crean un directorio en el sistema de archivos existente (/usr/lpp/rdz/cust) y definen un enlace simbólico (/etc/rdz) que lleva a él:

```
mkdir /usr/lpp/rdz/cust
ln -s /usr/lpp/rdz/cust /etc/rdz
```

---

## Cambios de PARMLIB

Consulte la publicación *MVS Initialization and Tuning Reference* (SA22-7592) para obtener más información acerca de las definiciones de PARMLIB listadas más abajo. Consulte la publicación *MVS System Commands* (SA22-7627) para obtener más información sobre los mandatos de consola de ejemplo.

## Establecer los límites de z/OS UNIX en BPXPRMxx

El Explorador de sistemas remotos (RSE), que proporciona servicios del núcleo como los de conectar el cliente al host, es un proceso basado en z/OS UNIX. Por ello, es importante establecer los valores correctos para los límites del sistema z/OS UNIX en BPXPRMxx, basándose en la cantidad de usuarios de Developer for System z que están activos simultáneamente y en su carga de trabajo media.

Consulte Capítulo 13, “Consideraciones acerca de los ajustes”, en la página 213 para obtener más información sobre los distintos límites de BPXPRMxx y su impacto sobre Developer for System z.

MAXASSIZE especifica el tamaño máximo de la región del espacio de direcciones (proceso). Establezca MAXASSIZE en SYS1.PARMLIB(BPXPRMxx) en 2G. Es el valor máximo permitido. Este es un límite a escala del sistema y, por ello, está activo para todos los espacios de direcciones z/OS UNIX. Si no desea este límite, puede establecer el límite únicamente para Developer for System z en el software de seguridad, tal como se describe en “Definir las tareas iniciadas de Developer for System z” en la página 183.

MAXTHREADS especifica el número máximo de hebras activas para un único proceso. Establezca MAXTHREADS en SYS1.PARMLIB(BPXPRMxx) en 1500 o más. Este es un límite a escala del sistema y, por ello, está activo para todos los espacios de direcciones z/OS UNIX. Si no desea este límite, puede establecer el límite únicamente para Developer for System z en el software de seguridad, tal como se describe en “Definir las tareas iniciadas de Developer for System z” en la página 183.

MAXTHREADTASKS especifica el número máximo de tareas de MVS activas para un único proceso. Establezca MAXTHREADTASKS en SYS1.PARMLIB(BPXPRMxx) en 1500 o más. Este es un límite a escala del sistema y, por ello, está activo para todos los espacios de direcciones z/OS UNIX. Si no desea este límite, puede establecer el límite únicamente para Developer for System z en el software de seguridad, tal como se describe en “Definir las tareas iniciadas de Developer for System z” en la página 183.

MAXPROCUSER especifica el número máximo de procesos que un solo ID de usuario de z/OS UNIX puede tener activos simultáneamente. Establezca MAXPROCUSER en SYS1.PARMLIB(BPXPRMxx) en 50 o más. Este valor pretende ser un límite a nivel del sistema, ya que debe estar activo para cada cliente que utiliza Developer for System z.

Estos valores pueden comprobarse y establecerse dinámicamente (hasta la próxima IPL) con los siguientes mandatos de consola:

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

#### Notas:

1. 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 153.
2. El valor MAXPROCUSER utilizado anteriormente se basa en usuarios que tengan un ID de usuario de z/OS UNIX (UID) exclusivo. Aumente este valor si los usuarios comparten el mismo UID.
3. Asegúrese de que otros valores de BPXPRMxx, como los correspondientes a MAXPROCSYS y MAXUIDS, sean suficientes para manejar la cantidad esperada de usuarios de Developer for System z activos simultáneamente. Encontrará más detalles en: Capítulo 13, “Consideraciones acerca de los ajustes”, en la página 213.

## Añadir tareas iniciadas a COMMNDxx

Añada mandatos de inicio para los servidores RSED, LOCKD y JMON de Developer for System z a SYS1.PARMLIB(COMMANDxx) para que se inicien automáticamente en la próxima IPL.

Una vez definidos y configurados los servidores, podrán iniciarse dinámicamente (hasta la próxima IPL) con los siguientes mandatos de consola:

- S RSED
- S LOCKD
- S JMON

**Nota:** El daemon de bloqueo debería iniciarse antes de que los usuarios de Developer for System z inicien sesión en el daemon RSE. El objetivo de ellos es que el daemon de bloqueo pueda rastrear las peticiones de bloqueo de conjuntos de datos de estos usuarios. Por ello, debería iniciar el daemon de bloqueo en el momento de inicio del sistema.

## Definiciones LPA en LPALSTxx

El servicio CARMA (Common Access Repository Manager) (opcional) da soporte a métodos alternativos de inicio del servidor que no requieren la utilización de un iniciador JES. La más flexible de estas alternativas requiere que el módulo CRSTART de la biblioteca de carga FEK.SFEKLPA se encuentre en la LPA (Área de paquete de enlace).

Los conjuntos de datos LPA se definen en SYS1.PARMLIB(LPALSTxx).

Las definiciones LPA pueden establecerse dinámicamente (hasta la próxima IPL) con los siguientes mandatos de consola:

- SETPROG LPA,ADD,DSN=FEK.SFEKLPA

## Autorizaciones APF en PROGxx

Para que el supervisor de trabajos JES pueda acceder a los archivos de spool JES, el módulo FEJJMON de la biblioteca de carga FEK.SFEKAUTH y las bibliotecas de tiempo de ejecución de Language Environment (LE) (CEE.SCEERUN\*) deben estar autorizadas por APF.

Para que SCLM Developer Toolkit (opcional) funcione, el módulo BWBTSOW de la biblioteca de carga FEK.SFEKAUTH y las bibliotecas de tiempo de ejecución de REXX (REXX.SEAGLPA) deben estar autorizadas por APF.

Para que ISPF pueda crear la Pasarela de cliente TSO/ISPF, el módulo ISPZTS0 de SYS1.LINKLIB debe estar autorizado por APF. El servicio de mandatos TSO de Developer for System z, SCLM Developer Toolkit y opcionalmente CARMA utilizan la Pasarela de cliente TSO/ISPF.

Las autorizaciones de APF se definen en SYS1.PARMLIB(PROGxx), si su local siguió las recomendaciones de IBM.

Las autorizaciones APF pueden establecerse dinámicamente (hasta la próxima IPL) con los siguientes mandatos de consola, donde volser es el volumen en el que reside el conjunto de datos si no está gestionado por SMS:

- SETPROG APF,ADD,DSN=FEK.SFEKAUTH,SMS
- SETPROG APF,ADD,DSN=CEE.SCEERUN,VOL=volser
- SETPROG APF,ADD,DSN=CEE.SCEERUN2,VOL=volser
- SETPROG APF,ADD,DSN=REXX.V1R4M0.SEAGLPA,VOL=volser
- SETPROG APF,ADD,DSN=SYS1.LINKLIB,VOL=volser

**Notas:**

1. Si utiliza la biblioteca alternativa para el paquete de producto REXX, el nombre predeterminado de la biblioteca de tiempo de ejecución de REXX es REXX.\*.SEAGALT en lugar de REXX.\*.SEAGLPA (como en el ejemplo anterior).
2. Las bibliotecas de LPA, como REXX.\*.SEAGLPA, se tienen la autorización APF automáticamente cuando están ubicadas en LPA y, por ello, no requieren definiciones explícitas.
3. Algunos de los productos correquisito, como por ejemplo IBM Debug Tool, también requieren autorización APF. Consulte las guías de personalización relacionadas para obtener más información acerca de este aspecto.

## Definiciones LINKLIST en PROGxx

Las definiciones LINKLIST de Developer for System z se puede agrupar en 3 categorías:

- Bibliotecas de carga de Developer for System z necesarias para las funciones de Developer for System z. Estas definiciones se describen en esta sección.
- Bibliotecas de carga requisito necesarias para las funciones de Developer for System z. Estas definiciones se describen en “Definiciones de LINKLIST y LPA requisito” en la página 20.
- Bibliotecas de carga de Developer for System z necesarias para otros productos. Estas definiciones se describen en “Definiciones LINKLIST para otros productos” en la página 21.

Para que SCLM Developer Toolkit (opcional) funcione, todos los módulos BWB\* de las bibliotecas de carga FEK.SFEKAUTH y FEK.SFEKLOAD deben estar disponibles por medio de STEPLIB o LINKLIST.

Si opta por utilizar STEPLIB, debe definir las bibliotecas no disponibles por medio de LINKLIST en la directiva STEPLIB de rsed.envvars, el archivo de configuración de RSE. Sin embargo, tenga en cuenta lo siguiente:

- el hecho de utilizar STEPLIB en z/OS UNIX afecta negativamente al rendimiento.
- Si una biblioteca de STEPLIB tiene autorización APF, todas deben tener autorización. Las bibliotecas pierden su autorización APF si se mezclan con bibliotecas sin autorización en STEPLIB.

Los conjuntos de datos de LINKLIST se definen en SYS1.PARMLIB(PROGxx), si su sitio ha seguido las recomendaciones de IBM.

Las definiciones necesarias serán parecidas a la siguiente, donde listname es el nombre del conjunto LINKLIST que se activara, y volser es el nombre volumen en el que reside el conjunto de datos si no está catalogado en el catálogo maestro:

- LNKLIST ADD NAME(listname) DSN=FEK.SFEKAUTH VOLUME(volser)
- LNKLIST ADD NAME(listname) DSN=FEK.SFEKLOAD

Las definiciones LINKLIST pueden crearse dinámicamente (hasta la próxima IPL) con el siguiente grupo de mandatos de consola, donde listname es el nombre del conjunto LINKLIST actual y volser es el volumen en el que reside el conjunto de datos si no está catalogado en el catálogo maestro:

1. LNKLIST DEFINE,NAME=LLTMP,COPYFROM=CURRENT
2. LNKLIST ADD NAME=LLTMP,DSN=FEK.SFEKAUTH,VOL=volser
3. LNKLIST ADD NAME=LLTMP,DSN=FEK.SFEKLOAD

4. LNKLIST ACTIVATE,NAME=LLTMP
5. LNKLIST UNDEFINE,NAME=listname
6. LNKLIST UPDATE,JOB=\*

## Definiciones de LINKLIST y LPA requisito

El Explorador de sistemas remotos (RSE) es un proceso z/OS UNIX que requiere acceso a las bibliotecas de carga de MVS. Las bibliotecas siguientes (prerrequisito) deben estar disponibles por medio de STEPLIB o LINKLIST/LPALIB:

- Biblioteca de carga del sistema
  - SYS1.LINKLIB
- Tiempo de ejecución de Language Environment
  - CEE.SCEERUN
  - CEE.SCEERUN2
- Biblioteca de clases DLL C++
  - CBC.SCLBDLL
- Pasarela de cliente TSO/ISPF de ISP
  - ISP.SISPLoad
  - ISP.SISPLPA

Las bibliotecas adicionales siguientes deben estar disponibles por medio de STEPLIB o LINKLIST/LPALIB, para dar soporte a la utilización de servicios opcionales. Esta lista no incluye los conjuntos de datos específicos de un producto con el que interactúa Developer for System z, como IBM Debug Tool:

- Biblioteca de tiempo de ejecución REXX (para SCLM Developer Toolkit)
  - REXX.\*.SEAGLPA
- Biblioteca de carga del sistema (para cifrado SSL)
  - SYS1.SIEALNKE
- Biblioteca de carga de TCP/IP (al utilizar APPC para el servicio de mandatos TSO)
  - TCPIP.SEZALOAD

### Notas:

1. Si utiliza la biblioteca alternativa para el paquete de producto REXX, el nombre predeterminado de la biblioteca de tiempo de ejecución de REXX es REXX.\*.SEAGALT en lugar de REXX.\*.SEAGLPA (como en el ejemplo anterior).
2. Las bibliotecas diseñadas para colocación en LPA, como REXX.\*.SEAGLPA, pueden requerir control de programa adicional y/o autorizaciones APF si se accede a ellas por medio de LINKLIST o STEPLIB.
3. Algunos de los productos correquisito, como por ejemplo IBM Debug Tool, también requieren definiciones STEPLIB o LINKLIST/LPALIB. Consulte las guías de personalización relacionadas para obtener más información acerca de este aspecto.
4. Si CEE.SCEELKED está en LINKLIST o STEPLIB, TCPIP.SEZALOAD debe colocarse antes de CEE.SCEELKED. Si no lo hace, se producirá una terminación anormal del sistema 0C1 para las llamadas de socket REXX de TCP/IP.

Los conjuntos de datos de LINKLIST se definen en SYS1.PARMLIB(PROGxx), si su sitio ha seguido las recomendaciones de IBM. Los conjuntos de datos LPA se definen en SYS1.PARMLIB(LPALSTxx).

Si opta por utilizar STEPLIB, debe definir las bibliotecas no disponibles por medio de LINKLIST/LPALIB en la directiva STEPLIB de rsed.envvars, el archivo de configuración de RSE. Sin embargo, tenga en cuenta lo siguiente:

- El hecho de utilizar STEPLIB en z/OS UNIX afecta negativamente al rendimiento.
- Si una biblioteca de STEPLIB tiene autorización APF, todas deben tener autorización. Las bibliotecas pierden su autorización APF si se mezclan con bibliotecas sin autorización en STEPLIB.
- Las bibliotecas añadidas al STEPLIB DD en un JCL no se propagan a los procesos z/OS UNIX iniciados por el JCL.

## Definiciones LINKLIST para otros productos

El cliente de Developer for System z tiene un componente de generación de código denominado Enterprise Service Tools (EST). Para que el código generado emita mensajes de error de diagnóstico, todos los módulos IRZ\* y IIRZ\* de la biblioteca de carga FEK.SFEKLOAD deben estar disponibles a través de STEPLIB o LINKLIST.

Los conjuntos de datos de LINKLIST se definen en SYS1.PARMLIB(PROGxx), si su sitio ha seguido las recomendaciones de IBM.

Si opta por utilizar STEPLIB, debe definir las bibliotecas no disponibles por medio de LINKLIST en la directiva STEPLIB de la tarea que ejecuta el código (IMS o trabajo por lotes). Sin embargo, debe tener en cuenta lo siguiente:

- Si una biblioteca de STEPLIB tiene autorización APF, todas deben tener autorización. Las bibliotecas pierden su autorización APF si se mezclan con bibliotecas sin autorización en STEPLIB.

---

## Cambios de PROCLIB

Los procedimientos de tarea iniciada y construcción remota indicados más abajo deben residir en una biblioteca de procedimientos del sistema definida en el subsistema JES. En las instrucciones que siguen, se utiliza la biblioteca de procedimientos predeterminada de IBM, SYS1.PROCLIB.

## Supervisor de trabajos JES

Personalice el miembro de tarea iniciada de ejemplo FEK.#CUST.PROCLIB(JMON), tal como se describe dentro del miembro, y cópielo en SYS1.PROCLIB. Como se muestra en el ejemplo de código que figura a continuación, debe suministrar lo siguiente:

- El calificador de alto nivel de la biblioteca de carga (autorizada), que por omisión es FEK
- El archivo de configuración del supervisor de trabajos JES, que por omisión es FEK.#CUST.PARMLIB(FEJJCNFG)

```

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

```

Figura 1. JMON - Tarea iniciada del Supervisor de trabajos JES

#### Notas:

1. Para obtener más información sobre los parámetros de inicio, consulte: Capítulo 8, “Mandatos de operador”, en la página 125.
2. Este JCL de ejemplo se suministra inicialmente como FEK.SFEKSAMP(FEJJJCL) y se redenomina como FEK.#CUST.PROCLIB(JMON) en “Configuración de la personalización” en la página 15.
3. El rastreo también se puede controlar mediante mandatos de la consola, como se describe en la sección Capítulo 8, “Mandatos de operador”, en la página 125.
4. Esta tarea debe estar asignada a SYSSTC o a un objetivo equivalente en el Gestor de carga de trabajo (WLM).
5. La variable de entorno LE \_CEE\_ENVFILE\_S requiere z/OS 1.8 o superior. La variable se puede sustituir por \_CEE\_ENVFILE en niveles de z/OS más antiguos pero, debido a un error en el tiempo de ejecución de C, la variable TZ del archivo de configuración del Supervisor de trabajos JES (FEJJCNFG) no se interpreta correctamente.

## Daemon RSE

Personalice el miembro de tarea iniciada de ejemplo FEK.#CUST.PROCLIB(RSED), tal como se describe dentro del miembro, y cópielo en SYS1.PROCLIB. Como se muestra en el ejemplo de código que figura a continuación, debe suministrar lo siguiente:

- El puerto del daemon RSE, que por omisión es el 4035.
- El directorio inicial en el que se ha instalado Developer for System z, que por omisión es /usr/lpp/rdz.
- La ubicación de los archivos de configuración, que por omisión es /etc/rdz.

```

/**
/** DAEMON RSE
/**
//RSED      PROC IVP='',                                * 'IVP' para realizar una prueba IVP
//          PORT=4035,
//          HOME='/usr/lpp/rdz',
//          CNFG='/etc/rdz'
/**
//RSE       EXEC PGM=BPXBATSL,REGION=0M,TIME=NOLIMIT,
//          PARM='PGM &HOME/bin/rsed.sh &IVP &PORT &CNFG'
//STDERR    DD SYSOUT=*
//STDOUT    DD SYSOUT=*
//          PEND
/**

```

Figura 2. RSED - Tarea iniciada del daemon RSE

**Nota:**

- Para obtener más información sobre los parámetros de inicio, consulte: Capítulo 8, “Mandatos de operador”, en la página 125.
- Este JCL de ejemplo se suministra inicialmente como FEK.SFEKSAMP(FEKRSED) y se redenomina como FEK.#CUST.PROCLIB(RSED) en “Configuración de la personalización” en la página 15.
- Limite la longitud del nombre del trabajo a 7 o menos caracteres. Los mandatos de operador **modificar** y **detener** fallarán devolviendo el mensaje "IEE342I MODIFY REJECTED-TASK BUSY" en caso que se utilice un nombre de 8 caracteres. Ello es debido al diseño de z/OS UNIX para los procesos hijo.
- Esta tarea, además de los procesos hijo que crea, debe estar asignada a SYSSTC o a un objetivo equivalente en el Gestor de carga de trabajo (WLM). Los procesos hijo tienen el mismo nombre que la tarea padre, RSED, con un número de 1 dígito aleatorio añadido, por ejemplo RSED8.

## Daemon de bloqueo

Personalice el miembro de tarea iniciada de ejemplo FEK.#CUST.PROCLIB(LOCKD), tal como se describe dentro del miembro, y cópielo en SYS1.PROCLIB. Como se muestra en el ejemplo de código que figura a continuación, debe suministrar lo siguiente:

- El directorio inicial en el que se ha instalado Developer for System z, que por omisión es /usr/lpp/rdz.
- La ubicación de los archivos de configuración, que por omisión es /etc/rdz.
- El nivel de detalle inicial de las anotaciones; el valor predeterminado es 1.

```

/**
/** RSE LOCK DAEMON
/**
//LOCKD     PROC HOME='/usr/lpp/rdz',
//          CNFG='/etc/rdz',
//          LOG=1
/**
//LOCKD     EXEC PGM=BPXBATSL,REGION=0M,TIME=NOLIMIT,
//          PARM=PGM &HOME./bin/lockd.sh &CNFG &LOG'
//STDOUT    DD SYSOUT=*
//STDERR    DD SYSOUT=*
//          PEND
/**

```

Figura 3. LOCKD - Tarea iniciada del daemon de bloqueo

#### Notas:

1. Para obtener más información sobre los parámetros de inicio, consulte: Capítulo 8, “Mandatos de operador”, en la página 125.
2. Este JCL de ejemplo se suministra inicialmente como FEK.SFEKSAMP(FEKLOCKD) y se redenomina como FEK.#CUST.PROCLIB(LOCKD) en “Configuración de la personalización” en la página 15.
3. Esta tarea debe estar asignada a SYSSTC o a un objetivo equivalente en el Gestor de carga de trabajo (WLM).

## Limitaciones del JCL para la variable PARM

La longitud máxima para la variable PARM es de 100 caracteres, lo cual puede causar problemas si utiliza nombres de directorio personalizados. Para evitar este problema, puede:

- Utilizar enlaces simbólicos

Los enlaces simbólicos pueden utilizarse como taquigrafía de un nombre de directorio largo. El siguiente mandato z/OS UNIX de ejemplo define un enlace simbólico (/usr/lpp/rdz) a otro directorio (/long/directory/name/usr/lpp/rdz).

```
ln -s /long/directory/name/usr/lpp/rdz /usr/lpp/rdz
```

- Utilizar STDIN

Si el campo PARM está vacío, **BPXBATSL** iniciará una shell z/OS UNIX y ejecutará el script de shell suministrado por STDIN. Tenga en cuenta que STDIN debe ser un archivo z/OS UNIX (asignado como ORDONLY) y que al utilizar STDIN se inhabilita la utilización de las variables PROC para el puerto, etc. Tenga también en cuenta que la shell ejecutará los scripts de inicio de sesión de shell /etc/profile y \$HOME/.profile.

Para utilizar este método, primero de actualizar el JCL de inicio para que coincida con un código parecido al del ejemplo siguiente:

```
/*  
/* DAEMON RSE - UTILIZACIÓN DE STDIN  
/*  
//RSED      PROC CNFG='/etc/rdz'  
/*  
//RSE       EXEC PGM=BPXBATSL,REGION=0M,TIME=NOLIMIT  
//STDOUT    DD SYSOUT=*  
//STDERR    DD SYSOUT=*  
//STDIN     DD PATHOPTS=(ORDONLY),PATH='&CNFG./rsed.stdin.sh'  
//STDENV    DD PATHOPTS=(ORDONLY),PATH='&CNFG./rsed.envvars'  
//          PEND  
/*
```

*Figura 4. RSED - Inicio alternativo de daemon RSE*

En segundo lugar, debe crear el script de shell (/etc/rdz/rsed.stdin.sh en este ejemplo) que iniciará el daemon RSE. El contenido de este script será como el del ejemplo siguiente:

```
/long/directory/name/usr/lpp/rdz/bin/rsed.sh 4035 /etc/rdz
```

*Figura 5. rsed.stdin.sh - inicio alternativo de daemon RSE*

**Nota:** Debe asignar rsed.envvars a STDENV en el JCL de inicio del daemon, ya que define algunas directivas de z/OS UNIX que ayudan a guardar recursos del sistema al utilizar este método de inicio.

## Procedimientos de construcción remota ELAXF\*

Developer for 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 a las instalaciones aplicar sus propios estándares y garantizará que los desarrolladores utilicen los mismos procedimientos con las mismas opciones de compilador y los mismos niveles de compilador.

Los procedimientos de ejemplo y sus funciones se listan 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).
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 muestra para hacer preproceso DB2 de sentencias EXEC SQL incluidas en programas COBOL.
ELAXFCOT	Procedimiento de ejemplo para hacer conversión CICS de sentencias CICS EXEC embebidas en programas COBOL.
ELAXFCPC	Procedimiento de muestra para hacer compilaciones en C.
ELAXFCPP	Procedimiento de muestra para hacer compilaciones en C++.
ELAXFCP1	Procedimiento de ejemplo para compilaciones COBOL con sentencias de preprocesador SCM (-INC e ++INCLUDE).
ELAXFDCL	Procedimiento de ejemplo para ejecutar un programa en modalidad TSO.
ELAXFGO	Procedimiento de ejemplo para el paso GO.
ELAXFLNK	Procedimiento de ejemplo para enlazar programas C/C++, COBOL, PLI y Assembler de alto nivel.
ELAXFMFS	Procedimiento de ejemplo para crear pantallas MFS IMS.
ELAXFPLP	Procedimiento de muestra para hacer preproceso DB2 de sentencias EXEC SQL incluidas 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.
ELAXFPP1	Procedimiento de ejemplo para compilaciones PL/I con sentencias de preprocesador SCM (-INC e ++INCLUDE).
ELAXFTSO	Procedimiento de ejemplo para ejecutar/depurar código DB2 generado en modalidad TSO.
ELAXFUOP	Procedimiento de ejemplo para generar el paso UOPT al construir programas que se ejecutan en subsistemas CICS o IMS.

Los nombres de los procedimientos y los nombres de los pasos de cada procedimiento coinciden con las propiedades predeterminadas que se suministran

con el cliente Developer for System z. Si decide cambiar el nombre de un procedimiento o el nombre de un paso del procedimiento, también hay que actualizar el correspondiente archivo de propiedades en todos los clientes. Le recomendamos que no cambie el nombre de los procedimientos ni de los pasos.

Personalice los miembros de procedimiento de construcción de ejemplo FEK.#CUST.PROCLIB(ELAXF\*), como se describe dentro de los miembros, y cópielos en SYS1.PROCLIB. Debe suministrar los calificadores de alto nivel correctos para las diversas bibliotecas de producto, como se describe en Tabla 8.

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

Producto	HLQ predeterminado	Valor
Developer for System z	FEK	
CICS	CICSTS32.CICS	
DB2	DSN910	
IMS	IMS	
COBOL	IGY.V4R1M0	
PL/I	IBMZ.V3R8M0	
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, solicite a los usuarios de Developer for System z que añadan una tarjeta JCLLIB (justo después de la tarjeta JOB) a las propiedades del trabajo en el cliente.

```
//MYJOB JOB <parámetros del trabajo>
//PROCS JCLLIB ORDER=(FEK.#CUST.PROCLIB)
```

## Definiciones de seguridad

Personalice y someta el miembro de ejemplo FEKRACF del conjunto de datos FEK.#CUST.JCL para crear las definiciones de seguridad para Developer for System z. El usuario que somete este trabajo debe tener privilegios de administrador de seguridad, como SPECIAL de RACF.

### Nota:

- Para los sitios que utilizan CA ACF2™ para z/OS, consulte el enlace siguiente, <https://support.ca.com/irj/portal/kbtech?ipLogNrow=0&docid=492389&searchID=TEC492389>, para obtener detalles sobre los mandatos de seguridad necesarios para configurar adecuadamente Developer for System z.
- Para los sitios que utilizan CA Top Secret® para z/OS, consulte la página del producto de la página de soporte de CA (<https://support.ca.com>) y compruebe el Documento de conocimiento de Developer for System z relacionado. Este Documento de conocimiento contiene detalles sobre los mandatos de seguridad necesarios para configurar correctamente Developer for System z.

La lista siguiente de definiciones obligatorias relacionadas con la seguridad para Developer for System z se describen con detalle en el Capítulo 10, “Consideraciones relativas a la seguridad”, en la página 159. Este capítulo también describe los aspectos generales relacionados con la seguridad de Developer for System z, que incluyen los aspectos de seguridad de los productos obligatorios que no se cubren en el trabajo de ejemplo FEKRACF.

- Activar valores y clases de seguridad
- Definir un segmento OMVS para usuarios de Developer for System z
- Definir perfiles de conjunto de datos
- Definir las tareas iniciadas JMON, RSED y LOCKD
- Definir la seguridad de mandatos JES
- Definir RSE como servidor z/OS UNIX seguro
- Definir las bibliotecas controladas por programa MVS para RSE
- Definir seguridad de aplicación para RSE
- Definir el soporte de PassTicket para RSE
- Definir los archivos controlados por programa z/OS UNIX para RSE

**Nota:** El trabajo FEKRACF no contiene sólo mandatos RACF. El último paso de las definiciones de seguridad consiste en convertir un archivo de z/OS UNIX en controlado por programa. Dependiendo de las políticas de su sitio, esta puede ser una tarea del programador del sistema y no del administrador de seguridad.

<b>Atención:</b> La solicitud de conexión del cliente fallará si la seguridad de la aplicación los PassTickets no están configurados correctamente.
---

---

## FEJJC�FG, archivo de configuración del supervisor de trabajos JES

El Supervisor de trabajos JES (JMON) suministra todos los servicios relacionados con JES. El comportamiento del Supervisor de trabajos JES puede controlarse con las definiciones de FEJJC�FG.

FEJJC�FG\* se encuentra en FEK.#CUST.PARMLIB, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15.

Personalice el miembro de configuración del supervisor de trabajos JES de ejemplo FEJJC�FG, como se muestra en el ejemplo siguiente. Las líneas de comentarios empiezan por el signo de almohadilla (#) cuando se utiliza una página de códigos de EE.UU. Las líneas de datos solamente pueden tener una directiva y su valor asignado; los comentarios no pueden estar en la misma línea.

**Nota:** Es necesario reiniciar la tarea iniciada JMON para recoger los cambios que haya realizado.

```

SERV_PORT=6715
TZ=EST5EDT
#_BPXK_SETIBMOPT_TRANSPORT=TCPIP
#APPLID=FEKAPPL
#AUTHMETHOD=SAF
#CODEPAGE=UTF-8
#CONCHAR=$
#CONSOLE_NAME=JMON
#GEN_CONSOLE_NAME=OFF
#HOST_CODEPAGE=IBM-1047
#LIMIT_COMMANDS=NOLIMIT
#LIMIT_VIEW=USERID
#LISTEN_QUEUE_LENGTH=5
#MAX_DATASETS=32
#MAX_THREADS=200
#TIMEOUT=3600
#TIMEOUT_INTERVAL=1200
#SUBMITMETHOD=TSO
#TSO_TEMPLATE=FEK.#CUST.CNTL(FEJTSO)

```

Figura 6. FEJCNFG, archivo de configuración del supervisor de trabajos JES

## 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 servidor y los clientes Developer for System z, deben estar configurados con el mismo número de puerto. Si cambia el número de puerto del servidor, todos los clientes 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**.
- Cuando se utiliza un cliente de la versión 7.1 o superior, toda comunicación en este puerto está confinada a la máquina de hospedaje z/OS.

**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 la publicación *UNIX System Services Command Reference* (SA22-7802).

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

## \_BPXK\_SETIBMOPT\_TRANSPORT

Especifica el nombre de la pila TCPIP que debe utilizarse. El valor predeterminado es TCPIP. Descoméntelo y cámbielo por 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 DD SYSTCPD en el JCL del servidor no establece la afinidad de pila solicitada.
- Cuando esta directiva no está activa, el Supervisor de trabajos JES enlaza con cada pila disponible del sistema (BIND INADDRANY).

## APPLID

Especifica el identificador de aplicación utilizado para la identificación del Supervisor de trabajos JES en su software de seguridad. El valor predeterminado es FEKAPPL. Descoméntelo y cámbielo por el ID de aplicación deseado.

**Nota:** Este valor debe coincidir con el ID de aplicación establecido para RSE en el archivo de configuración `rsed.envvars`. Si estos dos valores son distintos, RSE no puede conectar el cliente al Supervisor de trabajos JES.

## AUTHMETHOD

El valor predeterminado es SAF, y significa que se utiliza la interfaz de seguridad SAF (Recurso de autorización del sistema). No lo cambie, a menos que así se lo indique el centro de soporte de IBM.

## 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 tenga que descomentar la directiva y 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.

## CONCHAR

Especifica el carácter del mandato de consola JES. CONCHAR toma por defecto el valor CONCHAR=\$ para JES2, o el valor CONCHAR=\* para JES3. Descoméntelo y cámbielo por el carácter de mandato solicitado.

## CONSOLE\_NAME

Especifica el nombre de la consola de EMCS utilizada para emitir mandatos en los trabajos (Retener, Liberar, Cancelar y Depurar). El valor predeterminado es JMON. Descoméntelo y cámbielo por el name de consola deseado, utilizando las siguientes directrices.

- CONSOLE\_NAME debe ser un nombre de consola que puede tener de 2 hasta 8 caracteres alfanuméricos, o bien '&SYSUID' (sin comillas).
- Si se especifica un nombre de consola, se utiliza una única consola con ese nombre para todos los usuarios. Si la consola con ese nombre está en uso, el mandato emitido por el cliente fallará.
- Si se especifica &SYSUID, se utiliza el ID de usuario de cliente como nombre de la consola. Por ello, se utiliza una consola distinta para cada usuario. Si la consola con ese nombre está en uso (por ejemplo, el usuario está utilizando SDSF ULOG), puede que el mandato emitido por el cliente falle, dependiendo del valor de GEN\_CONSOLE\_NAME.

Independientemente de qué nombre de consola se utilice, se utiliza el ID de usuario del cliente que solicita el mandato como la LU de la consola, dejando un rastro en los mensajes de syslog IEA630I y IEA631I.

```
IEA630I OPERATOR console NOW ACTIVE,  SYSTEM=sysid, LU=id  
IEA631I OPERATOR console NOW INACTIVE, SYSTEM=sysid, LU=id
```

## GEN\_CONSOLE\_NAME

Habilita o inhabilita la generación automática de nombres de consola alternativos. El valor predeterminado es OFF. Descoméntelo y cámbielo a ON para habilitar los nombre de consola alternativos.

Esta directiva solamente se utiliza cuando CONSOLE\_NAME iguala &SYSUID, y el ID de usuario no está disponible como nombre de consola.

Si GEN\_CONSOLE\_NAME=ON, se genera un nombre de consola alternativo añadiendo un único dígito al ID de usuario. Se intentan todos los dígitos desde el 0 hasta el 9. En caso de que no se encuentre ninguna consola disponible, el mandato emitido por el cliente fallará.

Si GEN\_CONSOLE\_NAME=OFF, el mandato emitido por el cliente fallará.

**Nota:** Los únicos valores válidos son ON y OFF.

## HOST\_CODEPAGE

Página de códigos del host. El valor predeterminado es IBM-1047. Quite el carácter de comentario y cámbielo para que coincida con la página de códigos del host.

A partir de la versión 7.6.1, los clientes de Developer for System z ignoran el valor HOST\_CODEPAGE especificado aquí y utilizan la página de códigos especificada localmente en las propiedades del subsistema “Archivos MVS”.

**Nota:** Incluso para clientes recientes, el Supervisor de trabajos JES utilizará la página de códigos de host especificada en HOST\_CODEPAGE durante la configuración de comunicaciones del cliente.

## LIMIT\_COMMANDS

Define los trabajos en los que el usuario puede emitir mandatos de operador JES seleccionados (Mostrar JCL, Retener, Liberar, Cancelar y Depurar). El valor predeterminado (LIMIT\_COMMANDS=USERID) limita los mandatos a los trabajos propiedad del usuario. Descomente esta directiva y especifique LIMITED o NOLIMIT para permitir que el usuario emita mandatos ante todos los archivos de spool, si el producto de seguridad lo permite.

Tabla 9. Matriz de permisos de mandato LIMIT\_COMMANDS

LIMIT_COMMANDS	Propietario del trabajo	
	Usuario	Otros
USERID (valor predeterminado)	Permitido	No permitido
LIMITED	Permitido	Permitido sólo si lo permiten explícitamente los perfiles de seguridad
NOLIMIT	Permitido	Permitido si lo permiten los perfiles de seguridad o cuando la clase JESSPOOL no está activa

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

## LIMIT\_VIEW

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 lo permite el producto de seguridad. Descomente esta directiva y especifique USERID para limitar la vista a los datos de salida que sean propiedad del usuario.

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

### **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 número máximo de conjuntos de datos de salida en spool que el supervisor de trabajos JES devolverá al cliente (por ejemplo, SYSOUT, SYSPRINT, SYS00001, etc.). El valor predeterminado es 32. El valor máximo es 2147483647.

### **MAX\_THREADS**

Número máximo de usuarios que pueden utilizar un supervisor de trabajos JES en un momento dado. El valor predeterminado es 200. El valor máximo es 2147483647. Si aumenta este número, es posible que también deba aumentar el tamaño del espacio de direcciones del supervisor de trabajos JES.

### **TIMEOUT**

Tiempo, en segundos, que debe transcurrir antes de desactivar una hebra debido a que carece de interacción con el cliente. El valor predeterminado es 3600 (1 hora). El valor máximo es 2147483647. El valor TIMEOUT=0 inhabilita la función.

### **TIMEOUT\_INTERVAL**

El número de segundos entre comprobaciones de tiempo de espera. El valor predeterminado es 1200. El valor máximo es 2147483647.

### **SUBMITMETHOD=TSO**

Someter trabajos mediante TSO. El valor predeterminado (SUBMITMETHOD=JES) hace que los trabajos se sometan directamente a JES. Descomente esta directiva y especifique TSO para someter el trabajo por medio del mandato **SUBMIT** de TSO. Este método permite invocar rutinas de salida de TSO; sin embargo, afecta negativamente al rendimiento y, por tanto, no conviene utilizarlo.

#### **Nota:**

- Los únicos valores válidos son TSO y JES.
- Si se especifica SUBMITMETHOD=TSO, también hay que definir TSO\_TEMPLATE.

### **TSO\_TEMPLATE**

JCL de envoltura para someter el trabajo por medio de TSO. El valor predeterminado es FEK.#CUST.CNTL(FEJTS0). Esta sentencia hace referencia al nombre de miembro totalmente calificado del JCL que se debe usar como envoltura para el sometimiento TSO. Consulte la sentencia SUBMITMETHOD para obtener más información.

#### **Nota:**

- En FEK.#CUST.CNTL(FEJTS0) se proporciona un trabajo de envoltura de ejemplo. Consulte este miembro para obtener más información sobre la personalización que se necesita.
- TSO\_TEMPLATE no tiene efecto si no se especifica también SUBMITMETHOD=TSO.

---

## rsed.envvars, archivo de configuración de RSE

Los procesos del daemon de bloqueo RSE y del servidor RSE (daemon RSE, agrupación de hebras RSE y servidor RSE) utilizan las definiciones de `rsed.envvars`. Los servicios opcionales de Developer for System z y de terceros pueden utilizar también este archivo de configuración para definir variables de entorno para su uso.

El Explorador de sistemas remotos (RSE) proporciona servicios del núcleo como los de conectar el cliente al host e iniciar otros servidores para servicios específicos. El daemon de bloqueo proporciona servicios de rastreo para bloqueos de conjuntos de datos.

`rsed.envvars` se encuentra en `/etc/rdz/`, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP (FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15. Puede editar el archivo con el mandato TSO **OEDIT**.

Consulte el siguiente archivo `rsed.envvars` de ejemplo, que debe personalizarse para que coincida con el entorno del sistema. Las líneas de comentarios empiezan por el signo de almohadilla (#) cuando se utiliza una página de códigos de EE.UU. Las líneas de datos solamente pueden tener una directiva y su valor asignado; los comentarios no pueden estar en la misma línea. Las continuaciones de línea y los espacios alrededor del signo igual (=) no están soportados.

**Nota:** Es necesario reiniciar las tareas iniciadas RSED y LOCKD para recoger los cambios que haya realizado.



```

#=====
# (1) definiciones obligatorias
JAVA_HOME=/usr/lpp/java/J5.0
RSE_HOME=/usr/lpp/rdz
_RSE_LOCKD_PORT=4036
_RSE_HOST_CODEPAGE=IBM-1047
TZ=EST5EDT
LANG=C
PATH=/bin:/usr/sbin
_CEE_DMPTARG=/tmp
STEPLIB=NONE
#STEPLIB=$STEPLIB:CEE.SCEERUN:CEE.SCEERUN2:CBC.SCLBDLL
_RSE_SAF_CLASS=/usr/include/java_classes/IRRRacf.jar
_RSE_JAVAOPTS=""
_RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Xms1m -Xmx256m"
_RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Ddaemon.log=/var/rdz/logs"
_RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Duser.log=/var/rdz/logs"
_RSE_JAVAOPTS="$ _RSE_JAVAOPTS -DDSTORE_LOG_DIRECTORY="
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Dmaximum.clients=60"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Dmaximum.threads=1000"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Dminimum.threadpool.process=1"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Dmaximum.threadpool.process=100"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Dipv6=true"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Dkeep.last.log=true"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Denable.standard.log=true"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Denable.port.of.entry=true"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Denable.certificate.mapping=false"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Denable.automount=true"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Denable.audit.log=true"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Daudit.cycle=30"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Daudit.retention.period=0"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Ddeny.nonzero.port=true"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Dsingle.logon=false"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Dprocess.cleanup.interval=0"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -DAPPLID=FEKAPPL"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -DDENY_PASSWORD_SAVE=true"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Dhide_zos_unix=true"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -DDSTORE_IDLE_SHUTDOWN_TIMEOUT=3600000"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -DDSTORE_TRACING_ON=true"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -DDSTORE_MEMLOGGING_ON=true"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -DTSO_SERVER=APPC"
#=====
# (2) definiciones obligatorias para la Pasarela de cliente TSO/ISPF
_CMDSERV_BASE_HOME=/usr/lpp/ispf
_CMDSERV_CONF_HOME=/etc/rdz
_CMDSERV_WORK_HOME=/var/rdz
#STEPLIB=$STEPLIB:ISP.SISPLPA:SYS1.LINKLIB
_RSE_CMDSERV_OPTS=""
# _RSE_CMDSERV_OPTS="$ _RSE_CMDSERV_OPTS&ISPPROF=&SYSUID..ISPPROF"
#=====
# (3) definiciones obligatorias para SCLM Developer Toolkit
_SCLMDT_CONF_HOME=/var/rdz/sclmdt
#STEPLIB=$STEPLIB:FEK.SFEKAUTH:FEK.SFEKLOAD
# _SCLMDT_TRANTABLE=FEK.#CUST.LSTRANS.FILE
#ANT_HOME=/usr/lpp/Apache/Ant/apache-ant-1.7.1
#=====
# (4) definiciones opcionales
# _RSE_PORTRANGE=8108-8118
# _BPXK_SETIBMOPT_TRANSPORT=TCPIP
# _FEKFSCMD_TP_NAME=FEKFRSRV
# _FEKFSCMD_PARTNER_LU=nombre_lu
#GSK_CRL_SECURITY_LEVEL=HIGH
#GSK_LDAP_SERVER=ldap_server_url
#GSK_LDAP_PORT=ldap_server_port
#GSK_LDAP_USER=ldap_userid
#GSK_LDAP_PASSWORD=ldap_server_password
#=====

```

Figura 7. rsed.envvars - Archivo de configuración de RSE

```

# (5) no lo cambie a menos que así se lo indique el centro de soporte de IBM
_CEE_RUNOPTS="ALL31(ON) HEAP(32M,32K,ANYWHERE,KEEP,,) TRAP(ON)"
_BPX_SHAREAS=YES
_BPX_SPAWN_SCRIPT=YES
JAVA_PROPAGATE=NO
RSE_LIB=$RSE_HOME/lib
PATH=.:$JAVA_HOME/bin:$RSE_HOME/bin:$CMDSESV_BASE_HOME/bin:$PATH
LIBPATH=$JAVA_HOME/bin:$JAVA_HOME/bin/classic:$RSE_LIB:$RSE_LIB/icuc
LIBPATH=.:usr/lib:$LIBPATH
CLASSPATH=$RSE_LIB:$RSE_LIB/dstore_core.jar:$RSE_LIB/clientserver.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/dstore_extra_server.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/zosserver.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/dstore_miners.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/universalminers.jar:$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-core-2.3.2.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/cdtparser.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/wdzbidi.jar:$RSE_LIB/fmiExtensions.jar
CLASSPATH=$CLASSPATH:$RSE_SAF_CLASS
CLASSPATH=.:$CLASSPATH
_RSE_CMDSESV_OPTS="&SESSION=SPAWN$ RSE_CMDSESV_OPTS"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DISPF_OPTS='$_RSE_CMDSESV_OPTS'"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DA_PLUGIN_PATH=$RSE_LIB"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Xbootclasspath/p:$RSE_LIB/bidiTools.jar"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Dfile.encoding=$_RSE_HOST_CODEPAGE"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Dconsole.encoding=$_RSE_HOST_CODEPAGE"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DDSTORE_SPIRIT_ON=true"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DSPIRIT_EXPIRY_TIME=6"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DSPIRIT_INTERVAL_TIME=6"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Dcom.ibm.cacheLocalHost=true"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Duser.home=$HOME"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Dclient.username=$RSE_USER_ID"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Dlow.heap.usage.ratio=15"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Dmaximum.heap.usage.ratio=40"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DDSTORE_KEEPALIVE_ENABLED=true"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DDSTORE_KEEPALIVE_RESPONSE_TIMEOUT=30000"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DDSTORE_IO_SOCKET_READ_TIMEOUT=90000"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DRSECOMM_LOGFILE_MAX=0"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Dlock.daemon.port=$_RSE_LOCKD_PORT"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Dlock.daemon.cleanup.interval=1440"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -showversion"
_RSE_SERVER_CLASS=org.eclipse.dstore.core.server.Server
_RSE_DAEMON_CLASS=com.ibm.etools.zos.server.RseDaemon
_RSE_POOL_SERVER_CLASS=com.ibm.etools.zos.server.ThreadPoolProcess
_RSE_LOCKD_CLASS=com.ibm.ftt.rse.mvs.server.miners.MVSLockDaemon
_RSE_SERVER_TIMEOUT=120000
_SCLMDT_BASE_HOME=$RSE_HOME
_SCLMDT_WORK_HOME=$CMDSESV_WORK_HOME
CGI_DTWORK=$_SCLMDT_WORK_HOME
#=====
# (6) variables de entorno adicionales

```

Figura 8. (continuación)

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

Las definiciones que se necesitan son las siguientes:

**JAVA\_HOME**

Directorio inicial Java. El valor predeterminado es `/usr/lpp/java/J5.0`. Cámbielo para que coincida con su instalación de Java.

**RSE\_HOME**

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

**\_RSE\_LOCKD\_PORT**

Número de puerto del daemon de bloqueo RSE. El valor predeterminado es 4036. Se puede cambiar, si lo desea.

**Nota:**

- Antes de seleccionar un puerto, verifique que el puerto está disponible en su sistema con los mandatos **TSO NETSTAT** y **NETSTAT PORTL**.
- Toda la comunicación en este puerto está confinada a la máquina de hospedaje z/OS.

**\_RSE\_HOST\_CODEPAGE**

Página de códigos del host. El valor predeterminado es IBM-1047. Cámbielo para que coincida con la página de códigos del host.

**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 la publicación *UNIX System Services Command Reference* (SA22-7802).

**LANG**

Especifica el nombre del entorno local predeterminado. El valor predeterminado es C. C especifica el entorno local de POSIX (por ejemplo, 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 lo desea.

**\_CEE\_DMPTARG**

Ubicación de vuelcos de Language Environment (LE) z/OS UNIX utilizada por la máquina virtual Java (JVM). El valor predeterminado es `/tmp`.

**STEPLIB**

Acceso a conjuntos de datos MVS que no se encuentran en LINKLIST/LPALIB. El valor predeterminado es NONE.

Puede pasar por alto la necesidad de contar con bibliotecas prerequisite en LINKLIST/LPALIB descomentando y personalizando una o varias de las siguientes directivas STEPLIB. Consulte la sección “Cambios de PARMLIB” en la página 16 para obtener más información acerca de la utilización de las bibliotecas indicadas a continuación:

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

**Nota:**

- El hecho de utilizar STEPLIB en z/OS UNIX afecta negativamente al rendimiento.
- Si una biblioteca de STEPLIB tiene autorización APF, todas deben tener autorización. Las bibliotecas pierden su autorización APF si se mezclan con bibliotecas sin autorización en STEPLIB.
- Las bibliotecas diseñadas para colocación en LPA pueden requerir control de programa adicional y autorizaciones APF si se accede a ellas por medio de LINKLIST o STEPLIB.
- El hecho de codificar una sentencia STEPLIB DD en el JCL del servidor no establece la concatenación STEPLIB solicitada.

**RSE\_SAF\_CLASS**

Especifica la interfaz Java del producto de seguridad. El valor predeterminado es `/usr/include/java_classes/IRRRacf.jar`. Cámbielo para que coincida con su configuración de software de seguridad.

**Nota:** A partir de z/OS 1.10, `/usr/include/java_classes/IRRRacf.jar` forma parte de SAF, que se suministra con el producto base z/OS, por lo que también está disponible para los clientes no RACF.

**RSE\_JAVAOPTS**

Opciones Java específicas de RSE. . Para obtener más información sobre esta definición, vea: “Definir parámetros de inicio Java adicionales con `_RSE_JAVAOPTS`” en la página 42.

Por omisión, Developer for System z utiliza la Pasarela de cliente TSO/ISPF de ISPF para el servicio de mandatos TSO. Se utiliza una transacción APPC cuando la siguiente opción `_RSE_JAVAOPTS` no está comentada:

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

Las definiciones siguientes son obligatorias si se utiliza la Pasarela de cliente TSO/ISPF de ISPF para el servicio de mandatos TSO, SCLM Developer Toolkit o CARMA.

**`_CMDSERV_BASE_HOME`**

Directorio inicial para el código ISPF que suministra el servicio de Pasarela de cliente TSO/ISPF. El valor predeterminado es `/usr/lpp/ispf`. Cámbielo para que coincida con su instalación de ISPF. Esta directiva sólo es obligatoria si se utiliza la Pasarela de cliente TSO/ISPF de ISPF.

**`_CMDSERV_CONF_HOME`**

Directorio de configuración base de ISPF. El valor predeterminado es `/etc/rdz`. Cámbielo para que coincida con la ubicación de `ISPF.conf`, el archivo de personalización de la Pasarela de cliente TSO/ISPF de ISPF. Esta directiva sólo es obligatoria si se utiliza la Pasarela de cliente TSO/ISPF de ISPF.

**`_CMDSERV_WORK_HOME`**

Directorio de trabajo base de ISPF. El valor predeterminado es `/var/rdz`. Cámbielo para que coincida con la ubicación del directorio `WORKAREA` utilizado por la Pasarela de cliente TSO/ISPF. Esta directiva sólo es obligatoria si se utiliza la Pasarela de cliente TSO/ISPF de ISPF.

**Notas:**

- La Pasarela de cliente TSO/ISPF añadirá `/WORKAREA` a la vía de acceso especificada en `_CMDSERV_WORK_HOME`. No la añada usted.

- Si no ha utilizado el trabajo de ejemplo SFEKSAMP(FEKSETUP) para crear el entorno personalizable, debe verificar que el directorio WORKAREA existe en la vía de acceso especificada en \_CMDSERV\_WORK\_HOME. Los bits de permiso del directorio deben ser 777.

### **STEPLIB**

STEPLIB se ha descrito anteriormente en la sección dedicada a las definiciones obligatorias.

### **RSE\_CMDSERV\_OPTS**

Opciones Java adicionales específicas de la Pasarela de cliente TSO/ISPF. El valor predeterminado es "". Para obtener más información sobre esta definición, vea: "Definir parámetros de inicio Java adicionales con \_RSE\_CMDSERV\_OPTS" en la página 47. Esta directiva sólo es obligatoria si se utiliza la Pasarela de cliente TSO/ISPF de ISPF.

Las definiciones siguientes son necesarias si se utiliza SCLM Developer Toolkit.

### **SCLMDT\_CONF\_HOME**

Directorio de configuración base de SCLM Developer Toolkit. El valor predeterminado es /var/rdz/scldmt. Cámbielo para que coincida con la ubicación del directorio CONFIG utilizado por SCLMDT para almacenar información de proyectos SCLM. Esta directiva sólo es obligatoria si se utiliza SCLMDT.

**Nota:** SCLMDT añadirá /CONFIG y /CONFIG/PROJECT a la vía de acceso especificada en SCLMDT\_CNF\_HOME. No la añada usted.

### **STEPLIB**

STEPLIB se ha descrito anteriormente en la sección dedicada a las definiciones obligatorias.

### **\_SCLMDT\_TRANTABLE**

Nombre del VSAM de conversión de nombres largos/abreviados. El valor predeterminado es FEK.#CUST.LSTRANS.FILE. Descoméntelo y cámbielo para que coincida con el nombre utilizado en el trabajo de ejemplo de SCLM ISP.SISPSAMP(FLM02LST). Esta directiva solo es necesaria cuando se utiliza la conversión de nombres largos/abreviados en SCLM Developer Toolkit.

### **ANT\_HOME**

Directorio inicial de la instalación Ant. El valor predeterminado es /usr/lpp/apache/Ant/apache-ant-1.7.1. Cámbielo para que coincida con su instalación de Ant. Esta directiva sólo es necesaria cuando se utiliza el soporte de construcción JAVA/J2EE con SCLM Developer Toolkit.

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: "Definir el rango de puertos (PORTRANGE) disponibles para el servidor RSE" en la página 41. Esta directiva es opcional.

### **\_BPXK\_SETIBMOPT\_TRANSPORT**

Especifica el nombre de la pila TCP/IP que debe utilizarse. El valor predeterminado es TCPIP. Descoméntelo y cámbielo por el nombre de la pila TCP/IP solicitada, tal como se define en la sentencia TCPIPJOBNAME del TCPIP.DATA relacionado. Esta directiva es opcional.

**Nota:**

- El hecho de codificar una sentencia DD SYSTCPD en el JCL del servidor no establece la afinidad de pila solicitada.
- Cuando esta directiva no está activa, RSE enlaza con cada pila disponible del sistema (BIND INADDRANY).

**\_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. Esta directiva es opcional.

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

Obligar al servidor RSE a que utilice esta LU de asociado de APPC. El valor predeterminado es la LU base especificada durante la configuración de APPC. Esta directiva es opcional.

**GSK\_CRL\_SECURITY\_LEVEL**

Especifica el nivel de seguridad que utilizarán las aplicaciones SSL al contactar con los servidores LDAP para buscar certificados revocados en las CRL durante la validación de certificados. El valor predeterminado es MEDIUM. Quite el carácter de comentario y realice cambios para aplicar el uso del valor especificado. Esta directiva es opcional. Los siguientes valores son válidos:

- BAJ0 - La validación del certificado no fallará si no se puede contactar con el servidor LDAP.
- MEDIO - Para la validación del certificado es necesario poder contactar con el servidor LDAP, pero no es necesario que haya definida ninguna CRL. Este es el valor predeterminado.
- ALT0 - Para la validación del certificado es necesario poder contactar con el servidor LDAP y que haya una CRL definida.

**Nota:** Esta directiva requiere z/OS 1.9 o superior.

**GSK\_LDAP\_SERVER**

Especifica uno o varios nombres de host del servidor LDAP separados por espacios. Quite el carácter de comentario y realice cambios para aplicar el uso de los servidores LDAP especificados para obtener su CRL. Esta directiva es opcional.

El nombre de host puede ser una dirección TCP/IP o un URL. Cada nombre de host puede contener un número de puerto opcional separado del nombre de host por dos puntos (:).

**GSK\_LDAP\_PORT**

Especifica el puerto del servidor LDAP. El valor predeterminado es 389. Quite el carácter de comentario y realice cambios para aplicar el uso del valor especificado. Esta directiva es opcional.

**GSK\_LDAP\_USER**

Especifica el nombre distinguido que se debe utilizar al conectarse con el servidor LDAP. Quite el carácter de comentario y realice cambios para aplicar el uso del valor especificado. Esta directiva es opcional.

**GSK\_LDAP\_PASSWORD**

Especifica la contraseña que se debe utilizar al conectarse con el servidor LDAP. Quite el carácter de comentario y realice cambios para aplicar el uso del valor especificado. Esta directiva es opcional.

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

**\_CEE\_RUNOPTS**

Opciones de tiempo de ejecución de Language Environment (LE). El valor predeterminado es "ALL31(ON) HEAP(32M,32K,ANYWHERE,KEEP,,) TRAP(ON)". 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.

**JAVA\_PROPAGATE**

Propaga el contexto de seguridad y carga de trabajo durante la creación de hebras (sólo Java versión 1.4 y posteriores). El valor predeterminado es NO. No lo modifique.

**RSE\_LIB**

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

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

**LIBPATH**

Vía de acceso de la biblioteca. El valor predeterminado es demasiado largo para repetirlo. 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 específicas de 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 org.ibm.etools.eclipse.dstore.core.server.Server. No lo modifique.

**\_RSE\_DAEMON\_CLASS**

Clase Java para el daemon RSE. El valor predeterminado es com.ibm.etools.zos.server.RseDaemon. No lo modifique.

**\_RSE\_POOL\_SERVER\_CLASS**

Clase Java para la agrupación de hebras RSE. El valor predeterminado es com.ibm.etools.zos.server.ThreadPoolProcess. No lo modifique.

**\_RSE\_LOCKD\_CLASS**

Clase Java para el daemon de bloqueo RSE. El valor predeterminado es com.ibm.ftt.rse.mvs.server.miners.MVSLockDaemon. 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.

#### **`SCLMDT_BASE_HOME`**

Directorio inicial del código de SCLM Developer Toolkit. El valor predeterminado es `$RSE_HOME`. No lo modifique.

#### **`SCLMDT_WORK_HOME`**

Directorio de trabajo base de SCLM Developer Toolkit. El valor predeterminado es `$_CMDSERV_WORK_HOME`. No lo modifique.

#### **`CGI_DTWORK`**

Soporte de SCLM Developer Toolkit para clientes más antiguos. El valor predeterminado es `$_SCLMDT_WORK_HOME`. No lo modifique.

## **Definir el rango de puertos (PORTRANGE) disponibles para el servidor 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 el puerto 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 del host 4035, el daemon RSE.
2. El puerto del daemon RSE crea una hebra de servidor RSE.
3. El servidor 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 daemon RSE devuelve el número de puerto al cliente.
5. El cliente se conecta al puerto del host.

#### **Nota:**

- El proceso es similar para el método de conexión alternativo (opcional) mediante REXEC/SSH.
- Hallará más información en: Capítulo 11, “Qué es Developer for System z”, en la página 191.

Para especificar el rango de puertos para que el cliente se comunique con z/OS, descomente y personalice 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**.

El formato de `PORTRANGE` es: `_RSE_PORTRANGE=min-max` (max 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 se especifica un número de puerto distinto de cero 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.

**Nota:** El host puede denegar este tipo de solicitud de conexión especificando la directiva `deny.nonzero.port=true` en `rsed.envvars`. Consulte “Definir parámetros de inicio Java adicionales con `_RSE_JAVAOPTS`” para obtener más información sobre esta directiva.

2. Si el número de puerto de las propiedades del subsistema es 0, y si se especifica `_RSE_PORTRANGE` en el archivo `rsed.envvars`, se utilizará el rango de puertos especificado por `_RSE_PORTRANGE`. Si no hay ningún puerto disponible en el rango, la conexión fallará.
3. Si el número de puerto de las propiedades del subsistema es 0, y si no se ha especificado `_RSE_PORTRANGE` 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.

## Definir parámetros de inicio Java adicionales con `_RSE_JAVAOPTS`

Con las distintas directivas `_RSE_*OPTS`, el archivo `rsed.envvars` ofrece la posibilidad de proporcionar parámetros adicionales a Java cuando inicia los procesos RSE. Las opciones de ejemplo incluidas en el archivo `rsed.envvars` se pueden activar a base de quitarles el carácter de comentario.

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

`_RSE_JAVAOPTS=""`

Inicialización de variables. No lo modifique.

`_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Xms1m -Xmx256m"`

Establecer el tamaño inicial (Xms) y máximo (Xmx) de la memoria dinámica. Los valores predeterminados son 1M y 256M respectivamente. Cámbielo para aplicar los valores de tamaño de almacenamiento dinámico deseados. Si esta directiva tiene caracteres de comentario, se utilizarán los valores predeterminados de Java, que son 4M y 512M respectivamente (1M y 64M para Java 5.0).

**Nota:** Consulte la sección “Definiciones de recursos clave” en la página 233 para determinar los valores óptimos para esta directiva.

`_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Ddaemon.log=/var/rdz/logs"`

Directorio que contiene los datos de inicio de sesión para el servidor y el daemon RSE, y los datos de auditoría de RSE. El valor predeterminado es `/var/rdz/logs`. Cámbielo para aplicar la ubicación deseada. Si esta directiva tiene caracteres de comentario, se utilizará el directorio inicial del ID de usuario asignado al daemon RSE. El directorio inicial se define en el segmento de seguridad OMVS del ID de usuario.

**Nota:** Si esta directiva (o su contrapartida, el directorio inicial) no especifican una vía de acceso absoluta (la vía de acceso no empieza

con una barra inclinada hacia delante (/)), la ubicación real de las anotaciones es relativa al directorio de configuración (de forma predeterminada /etc/rdz).

**\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Duser.log=/var/rdz/logs"**

Directorio que conduce a las anotaciones específicas del usuario. El valor predeterminado es /var/rdz/logs. Cámbielo para aplicar la ubicación deseada. Si esta directiva tiene caracteres de comentario, se utilizará el directorio inicial del ID de usuario del cliente. El directorio inicial se define en el segmento de seguridad OMVS del ID de usuario.

**Nota:**

- Si esta directiva (o su contrapartida, el directorio inicial) no especifican una vía de acceso absoluta (la vía de acceso no empieza con una barra inclinada hacia delante (/)), la ubicación real de las anotaciones es relativa al directorio de configuración (de forma predeterminada /etc/rdz).
- La vía de acceso completa a las anotaciones del usuario es userlog/dstorelog/\$LOGNAME/, donde userlog es el valor de la directiva user.log, dstorelog es el valor de la directiva DSTORE\_LOG\_DIRECTORY y \$LOGNAME es el ID de usuario del cliente en mayúsculas.
- Asegúrese de que los bits de permiso para userlog/dstorelog están establecidos de manera que cada cliente pueda crear \$LOGNAME.

**\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -DDSTORE\_LOG\_DIRECTORY=**

Este directorio se añade a la vía de acceso especificada en la directiva user.log. Juntos crean la vía de acceso que conduce a las anotaciones específicas del usuario. El valor predeterminado es una serie vacía. Cámbielo para aplicar el uso del directorio especificado. Si esta directiva tiene caracteres de comentario, se utilizará .eclipse/RSE/.

**Nota:**

- La vía de acceso completa a las anotaciones del usuario es userlog/dstorelog/\$LOGNAME/, donde userlog es el valor de la directiva user.log, dstorelog es el valor de la directiva DSTORE\_LOG\_DIRECTORY y \$LOGNAME es el ID de usuario del cliente en mayúsculas.
- El directorio especificado aquí está relacionado con el directorio especificado en user.log y, por ello, no puede empezar con una barra inclinada (/).
- Asegúrese de que los bits de permiso para userlog/dstorelog están establecidos de manera que cada cliente pueda crear \$LOGNAME.

Las directivas siguientes están comentadas por omisión.

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Dmaximum.clients=60"**

Número máximo de clientes a los que proporciona servicios una agrupación de hebras. El valor predeterminado es 60. Descomente y personalice este valor para limitar el número de clientes por agrupación de hebras. Tenga en cuenta que puede que otros límites impidan que RSE llegue a este límite.

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Dmaximum.threads=1000"**

Cantidad máxima de hebras activas en una agrupación de hebras para permitir clientes nuevos. El valor predeterminado es 1000. Descomente y personalice este valor para limitar el número de clientes por agrupación de hebras según el número de hebras que se estén utilizando. Tenga en cuenta que cada conexión de cliente utiliza varias hebras (16 o más) y que otros límites pueden impedir que RSE llegue a este límite.

**Nota:** Este valor debe ser inferior al valor de MAXTHREADS y MAXTHREADTASKS en SYS1.PARMLIB(BPXPRMxx).

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Dminimum.threadpool.process=1"**

Número mínimo de agrupaciones de hebras activas. El valor predeterminado es 1. Descomente y personalice este valor para iniciar como mínimo el número de procesos de agrupaciones de hebras indicado. Los procesos de agrupaciones de hebras se utilizan para el equilibrio de carga de las hebras del servidor RSE. Se inician más procesos nuevos cuando estos son necesarios. Iniciar procesos nuevos ayuda a evitar los retrasos de conexión pero utiliza más recursos durante momentos desocupados.

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Dmaximum.threadpool.process=100"**

Número máximo de agrupaciones de hebras activas. El valor predeterminado es 100. Descomente y personalice este valor para limitar el número de procesos de agrupaciones de hebras. Los procesos de agrupaciones de hebras se utilizan para el equilibrio de carga de las hebras del servidor RSE, por lo que, al limitarlos, se limitará la cantidad de conexiones de cliente activas.

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Dipv6=true"**

Versión de TCP/IP. El valor predeterminado es false, que significa que se utilizará una interfaz IPv4. Descoméntelo y especifique true para utilizar una interfaz IPv6.

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Dkeep.last.log=true"**

Conserve una copia de los archivos de anotaciones del host que pertenecen a la sesión anterior. El valor predeterminado es false. Descomente y especifique true para renombrar los archivos de anotaciones anteriores como \*.last durante el inicio del servidor y la conexión del cliente.

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Denable.standard.log=true"**

Escriba las secuencias stdout y stderr de las agrupaciones de hebras en un archivo de anotaciones. El valor predeterminado es false. Descomente y especifique true para guardar las secuencias stdout y stderr. Los archivos de anotaciones resultantes están ubicados en el directorio referido por la directiva daemon.log.

**Nota:**

- El mandato de operador **MODIFY RSESTANDARDLOG** se puede utilizar para detener o iniciar dinámicamente la actualización de los archivos de anotaciones de la secuencia.
- No hay archivos de anotaciones stdout.log y stderr.log específicos del usuario cuando la directiva enable.standard.log está activa. Ahora se escriben datos específicos del usuario en la secuencia de agrupación de hebras RSE coincidente.

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Denable.port.of.entry=true"**

Opción de comprobación de puerto de entrada (POE). El valor

predeterminado es false. Descomente y especifique true para aplicar la comprobación de POE a las conexiones de cliente. Durante la comprobación de POE, el software de seguridad correlaciona la dirección IP del cliente con una zona de seguridad de acceso a red. El ID de usuario del cliente debe tener autorización para utilizar el perfil que define la zona de seguridad.

**Nota:**

- La comprobación de POE también debe habilitarse en el producto de seguridad.
- Al habilitar la comprobación de POE, ésta se habilitará también para otros servicios z/OS UNIX, como por ejemplo INETD.

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Denable.certificate.mapping=false"**

Utilizar su software de seguridad para autenticar un inicio de sesión con un certificado X.509. El valor predeterminado es true. Descoméntelo y especifique false para que el daemon RSE realice la autenticación sin basarse en el soporte X.509 de su software de seguridad.

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Denable.automount=true"**

Soportar los directorios iniciales creados por automount de z/OS UNIX. El valor predeterminado es false. Quite el carácter de comentario y especifique true para asegurarse de que automount de z/OS UNIX utiliza el ID de usuario del cliente como propietario del directorio.

**Nota:** automount de z/OS UNIX utiliza el ID de usuario del proceso que invoca el servicio al crear un sistema de archivos. Si esta opción está inhabilitada, este proceso es el servidor de agrupaciones de hebras de RSE (ID de usuario STCRSE). Si esta opción está habilitada, se crea un proceso temporal nuevo mediante el ID de usuario de cliente antes de invocar el servicio.

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Denable.audit.log=true"**

Opción de auditoría. El valor predeterminado es false. Descoméntelo y especifique true para aplicar las anotaciones de auditoría a las acciones realizadas por los clientes. Las anotaciones de auditoría se graban en la ubicación de anotaciones del daemon RSE. Consulte la opción daemon.log de la variable \_RSE\_JAVAOPTS para saber dónde se encuentra.

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Daudit.cycle=30"**

Número de días almacenados en un archivo de anotaciones de auditoría. El valor predeterminado es 30. Descomente y personalice este valor para controlar la cantidad de datos de auditoría que se graban en un archivo de anotaciones de auditoría. El valor máximo es 365.

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Daudit.retention.period=0"**

Número de días que se conservan las anotaciones de auditoría. El valor predeterminado es 0 (sin límite). Descomente y personalice este valor para suprimir las anotaciones de auditoría después de un número de días determinado. El valor máximo es 365.

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Ddeny.nonzero.port=true"**

No permitir que el cliente elija el número del puerto de comunicaciones. El valor predeterminado es false. Quite el carácter de comentario y especifique true para rechazar las conexiones en las que el cliente especifica qué número de puerto de host debe utilizar el servidor RSE para la conexión. Hallará más información en: "Definir el rango de puertos (PORTRANGE) disponibles para el servidor RSE" en la página 41.

| **#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Dsingle.logon=false"**

| No permitir que un ID de usuario inicie varias sesiones. El valor  
| predeterminado es true. Quite el comentario y especifique false para  
| permitir que un ID de usuario inicie varias sesiones en un sólo daemon  
| RSE.

| **Nota:** Un segundo inicio de sesión hará que el host cancele el primero si  
| esta directiva no está activa o establecida en false. Esta cancelación  
| irá acompañada del mensaje de la consola FEK210I.

| **RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Dprocess.cleanup.interval=0"**

| Eliminar automáticamente agrupaciones de hebras RSE que estén en un  
| estado de error irrecuperable. De forma predeterminada, las agrupaciones  
| de hebras de RSE erróneas no se eliminan automáticamente. Quite el  
| carácter de comentario y personalice para eliminar automáticamente los  
| servidores de agrupaciones de hebras de RSE erróneas en cada intervalo (la  
| unidad del intervalo es segundos). Si se especifica 0, se inhabilita la  
| función.

| **#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -DAPPLID=FEKAPPL"**

| ID de aplicación del servidor RSE. El valor predeterminado es FEKAPPL.  
| Descoméntelo y personalice esta opción para aplicar el uso del ID de  
| aplicación deseado.

**Nota:**

- El ID de aplicación debe definirse para el software de seguridad. En caso de que esta acción no se realice correctamente, el cliente no podrá iniciar sesión.
- Consulte la "Utilizar PassTickets" en la página 164 para obtener información sobre las implicaciones de seguridad al cambiar este valor.
- El ID de aplicación debe coincidir con el ID de aplicación utilizado por el Supervisor de trabajos de JES. Consulte la sección "FEJJCNFG, archivo de configuración del supervisor de trabajos JES" en la página 27 para aprender a definir el ID de aplicación para el Supervisor de trabajos JES.

| **#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -DDENY\_PASSWORD\_SAVE=true"**

| Opción de guardado de contraseña. El valor predeterminado es false.  
| Descoméntelo y especifique true para 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 y posteriores.

| **#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Dhide\_zos\_unix=true"**

| Ocultar opción z/OS UNIX. El valor predeterminado es false.  
| Descoméntelo y especifique true para impedir que los usuario puedan ver  
| los elementos de z/OS UNIX (estructura del directorio y línea de  
| mandatos) en el cliente. Esta opción solo funciona con clientes de la  
| versión 7.6 y posteriores.

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

| **-DDSTORE\_IDLE\_SHUTDOWN\_TIMEOUT=3600000"**

| Desconectar los clientes desocupados. Por omisión, los clientes  
| desocupados no se desconectan. Descomente y personalice este valor para  
| desconectar los clientes que estén desocupados durante el número de  
| milisegundos indicado (3600000, igual a 1 hora).

```
#_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. Tenga en cuenta que el archivo de anotaciones .dstoreTrace resultante se crea en Unicode (ASCII), no en EBCDIC.

```
#_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. Tenga en cuenta que el archivo de anotaciones .dstoreMemLogging resultante se crea en Unicode (ASCII), no en EBCDIC.

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

Utilizar una transacción APPC para el servicio de mandatos TSO. Por omisión, se utiliza la Pasarela de cliente TSO/ISPF de ISPF. Descoméntelo para utilizar una transacción APPC en su lugar. No cambie el valor asignado.

## Definir parámetros de inicio Java adicionales con \_RSE\_CMDSERV\_OPTS

Con las distintas directivas \_RSE\_\*OPTS, el archivo rsed.envvars ofrece la posibilidad de proporcionar parámetros adicionales a Java cuando inicia los procesos RSE. Las opciones de ejemplo incluidas en el archivo rsed.envvars se pueden activar a base de quitarles el carácter de comentario.

Las directivas \_RSE\_CMDSERV\_OPTS son opciones Java específicas de RSE y solo entran en vigor cuando el Developer for System z utiliza la Pasarela de cliente TSO/ISPF de ISPF. (Este es el valor predeterminado.)

```
_RSE_CMDSERV_OPTS=""
```

Inicialización de variables. No lo modifique.

```
_RSE_CMDSERV_OPTS="$_RSE_CMDSERV_OPTS &ISPROF=  
&SYSUID..ISPROF="
```

Utilice un perfil ISPF existente para la inicialización de ISPF. Quite el carácter de comentario y cambie el nombre del conjunto de datos para que utilice el perfil ISPF especificado.

Pueden utilizarse las variables siguientes en el nombre del conjunto de datos:

- &SYSUID, para sustituir el ID de usuario del desarrollador.
- &SYSPREF, para sustituir el prefijo de TSO del desarrollador.

---

## ISPF.conf, archivo de configuración de la Pasarela de cliente TSO/ISPF de ISPF

La Pasarela de cliente TSO/ISPF de ISPF utiliza las definiciones de ISPF.conf para crear un entorno válido para ejecutar mandatos TSO e ISPF por lotes. Developer for System z utiliza este entorno para ejecutar algunos de los servicios basados en MVS. Estos servicios incluyen el servicio de mandatos TSO, SCLM Developer Toolkit y un método de inicio alternativo de CARMA.

ISPF.conf se encuentra en /etc/rdz/, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: "Configuración de la personalización" en la página 15. Puede editar el archivo con el mandato TSO OEDIT.

Las líneas de comentarios empiezan por un asterisco (\*) cuando se utiliza una página de códigos de EE.UU. Las líneas de datos solamente pueden tener una directiva y su valor asignado. Los comentarios no pueden estar en la misma línea. Las continuaciones de línea no están soportadas. Al concatenar nombres de conjuntos de datos, añádalos en la misma línea y separe los nombres con una coma (,).

Además de suministrar los nombres correctos para los conjuntos de datos de ISPF, también debe añadir el nombre del conjunto de datos del servicio de mandatos TSO, FEK.SFEKPROC, a la sentencia SYSPROC o SYSEXEC, como se muestra en el ejemplo siguiente.

```
* OBLIGATORIO:
sysproc=ISP.SISPCLIB,FEK.SFEKPROC
ispmlib=ISP.SISPMENU
isptlib=ISP.SISPTENU
ispplib=ISP.SISPPENU
ispplib=ISP.SISPLIB
ispllib=ISP.SISLOAD

* OPCIONAL:
*allocjob = FEK.#CUST.CNTL(CRAISPRX)
*ISPF_timeout = 900
```

*Figura 9. ISPF.conf - Archivo de configuración de ISPF*

**Nota:**

- Puede añadir sus propias sentencias de tipo DD y concatenaciones de conjuntos de datos para personalizar el entorno TSO, imitando un procedimiento de inicio de sesión de TSO. Encontrará más detalles en: Capítulo 16, “Personalizar el entorno TSO”, en la página 265.
- Puede que la Pasarela de cliente TSO/ISPF no funcione correctamente si utiliza un producto (de terceros) que intercepta mandatos ISPF, como por ejemplo **ISPSTART**. Consulte la documentación de dicho producto para saber cómo inhabilitarlo para Developer para System z. Si el producto requiere la asignación de una sentencia DD específica a DUMMY, puede simularlo en ISPF.conf asignando esa sentencia DD a nullfile.

Por ejemplo:

```
ISPTRACE=nullfile
```

- Al utilizar la directiva allocjob, tenga cuidado de no deshacer las definiciones DD creadas antes en ISPF.conf.
- En caso de que el parámetro JWT del miembro parmlib SMFPRMxx sea menor que el valor ISPF\_timeout de ISPF.conf se esperará la terminación anormal del sistema 522 del módulo ISPZTS0. Ello no afecta a las operaciones de Developer for System z, ya que la pasarela de cliente TSO/ISPF se reinicia automáticamente cuando es necesario.
- Los cambios están activos para todas las invocaciones nuevas. No es necesario reiniciar el servidor.

---

## Componentes opcionales

Los pasos de personalización indicados anteriormente corresponden a una configuración básica de Developer for System z. Consulte los capítulos dedicados a los componentes opcionales para conocer sus requisitos de personalización.

- Capítulo 3, “(Opcional) CARMA (Common Access Repository Manager)”, en la página 51

- Capítulo 4, “(Opcional) Gestor de despliegue de aplicaciones (ADM)”, en la página 73
- Capítulo 5, “(Opcional) SCLM Developer Toolkit”, en la página 81
- “(Opcional) Procedimiento almacenado DB2” en la página 89
- “(Opcional) Soporte de idiomas bidireccionales CICS” en la página 92
- “(Opcional) Cifrado SSL de RSE” en la página 93
- “(Opcional) Rastreo de RSE” en la página 96
- “(Opcional) Grupos de propiedades basadas en host” en la página 98
- “(Opcional) Proyectos basados en host” en la página 99
- “(Opcional) Integración del gestor de archivos” en la página 100
- “(Opcional) Caracteres no editables” en la página 101
- “(Opcional) Utilizar REXEC (o SSH)” en la página 102
- “(Opcional) Transacción APPC para el servicio de mandatos TSO” en la página 104
- “(Opcional) Limpieza de WORKAREA” en la página 108

---

## Verificación de la instalación

La descripción de los programas de verificación de la instalación (IVP) está ubicada en Capítulo 7, “Verificación de la instalación”, en la página 109, porque algunos de los IVP son para componentes opcionales.



---

## Capítulo 3. (Opcional) CARMA (Common Access Repository Manager)

CARMA (Common Access Repository Manager) es una ayuda de productividad para los desarrolladores que crean RAM (Repository Access Managers). Un RAM es una interfaz de programación de aplicaciones (API) para SCM (Software Configuration Managers) basados en z/OS.

A su vez, las aplicaciones escritas por usuario pueden iniciar un servidor CARMA que cargue los RAM y suministre una interfaz estándar para acceder al SCM.

Developer for System z da soporte a varios métodos para iniciar un servidor CARMA, cada uno de los cuales con sus propias ventajas e inconvenientes.

- El método “sometimiento por lotes” inicia un servidor CARMA sometiendo un trabajo. Este es el método predeterminado utilizado en los archivos de configuración de ejemplo suministrados. La ventaja de este método es que puede accederse fácilmente a las anotaciones de CARMA en la salida del trabajo. También permite utilizar JCL de servidor personalizado para cada desarrollador, cuyo mantenimiento realiza el propio desarrollador. Sin embargo, este método utiliza un iniciador de JES por cada desarrollador que inicia un servidor CARMA.
- El método “CRASTART” iniciar el servidor CARMA como subtask de RSE. Ofrece una configuración muy flexible mediante la utilización de un archivo de configuración independiente que define las asignaciones de conjunto de datos e invocaciones de programa necesarias para iniciar un servidor CARMA. Este método ofrece el mejor rendimiento y utiliza la menor cantidad de recursos, pero requiere que el módulo CRASTART se encuentre en LPA.
- El método de “Pasarela de cliente TSO/ISPF” utiliza la Pasarela de cliente TSO/ISPF de ISPF para crear un entorno TSO o ISPF, en el que se inicia el servidor CARMA. Permite asignaciones de conjunto de datos flexibles utilizando las posibilidades de ISPF.conf. Sin embargo, este método no es adecuado para acceder a los SCM que interfieren en las operaciones habituales de TSO o ISPF.

---

### Requisitos y lista de comprobación

Necesitará ayuda de un administrador de seguridad y de un administrador de TCP/IP para realizar esta tarea de personalización, que requiere los siguientes recursos o tareas de personalización especiales:

- Rango de puertos TCP/IP para la comunicación interna
- Regla de seguridad para permitir que los desarrolladores realicen actualizaciones en los archivos VSAM de CARMA
- (Opcional) Regla de seguridad para permitir a los usuarios someter trabajos CRA\*
- (Opcional) Actualización de LPA

A fin de empezar a utilizar CARMA en su sitio, debe realizar las tareas siguientes. A menos que se indique de otro modo, todas las tareas son obligatorias.

1. Crear los componentes de CARMA necesarios. Para obtener más detalles, consulte “Componentes de CARMA” en la página 52.

2. Personalización inicial de los archivos de configuración de RSE para intercambiar información con CARMA. La personalización completa depende del método escogido para iniciar CARMA. Para obtener más detalles, consulte “Interfaz RSE con CARMA” en la página 53.
3. Elija un método para iniciar CARMA y realice la personalización necesaria de los archivos de configuración relacionados. Para obtener más detalles, consulte:
  - “Inicio del servidor CARMA mediante el sometimiento por lotes” en la página 55
  - “(Opcional) Inicio alternativo del servidor CARMA mediante CRASTART” en la página 57
  - “(Opcional) Inicio alternativo del servidor CARMA mediante la Pasarela de cliente TSO/ISPF” en la página 60
4. Activar de forma opcional los RAM (Repository Access Managers) de ejemplo. Para obtener más detalles, consulte “(Opcional) Activar los RAM (Repository Access Managers) de ejemplo” en la página 62.
5. Opcionalmente, activar el RAM del CA Endevor®. Para obtener más detalles, consulte “(Opcional) Activar CA Endevor® SCM RAM” en la página 63.
6. Crear de forma opcional CRAXJCL como sustitución para IRXJCL. Para obtener más detalles, consulte “(Opcional) Comparación entre IRXJCL y CRAXJCL” en la página 71.

**Nota:** Los miembros de ejemplo a los que se hace referencia en este capítulo se encuentran en FEK.#CUST.\* y /etc/rdz, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15.

---

## Componentes de CARMA

Los siguientes componentes de CARMA deben personalizarse, independientemente del método de inicio elegido. Los siguientes miembros de ejemplo se encuentran en FEK.#CUST.JCL, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15.

1. Personalice y someta el JCL FEK.#CUST.JCL(CRA\$VDEF). Las instrucciones de personalización están en la documentación de CRA\$VDEF. CRA\$VDEF crea y prepara el conjunto de datos VSAM de configuración de CARMA, CRADEF.
2. Personalice y someta el JCL FEK.#CUST.JCL(CRA\$VMSG). Las instrucciones de personalización están en la documentación de CRA\$VMSG. CRA\$VMSG crea y prepara el conjunto de datos VSAM de mensajes de CARMA, CRAMSG.
3. Personalice y someta el JCL FEK.#CUST.JCL(CRA\$VSTR). Las instrucciones de personalización están en la documentación de CRA\$VSTR. CRA\$VSTR crea y prepara el conjunto de datos VSAM de información de personalización de CARMA, CRASTRS.

**Nota:**

- Los VSAM de CARMA creados con estos trabajos definen las RAM de ejemplo. Consulte “(Opcional) Activar CA Endevor® SCM RAM” en la página 63 para definir el RAM del CA Endevor®.
- Consulte el trabajo de ejemplo FEK.#CUST.JCL(CRA#UADD) si necesita fusionar las definiciones de un RAM (personalizada) en una configuración de VSAM existente. Es necesario personalizar y someter este trabajo por cada VSAM de CARMA modificado. Consulte la publicación *Rational*

*Developer for System z Common Access Repository Manager Developer's Guide* (SC23-7660) para obtener más información acerca de la estructura de registro utilizada por los distintos VSAM de CARMA.

- Utilice el trabajo de ejemplo FEK.#CUST.JCL(CRA#UQRY) para extraer las definiciones activas de un VSAM a un conjunto de datos secuencial.

## Notas de migración de VSAM de CARMA

Developer for System z Versión 7.6.1 soporta un diseño de estructura de datos nuevo para el conjunto de datos VSAM de información personalizada de CARMA, CRASTRS, para eliminar limitaciones de longitud de mensaje.

Antes de Developer for System z Versión 7.6.1, las series definidas en el conjunto de datos VSAM de información personalizada de CARMA estaban limitadas a las longitudes predefinidas. Esta limitación hace que los desarrolladores RAM abrevien las series descriptivas o que utilicen plug-ins del lado del cliente para visualizar series en toda su longitud.

Developer for System z Versión 7.6.1 soporta un diseño de estructura de datos de longitud variable nuevo para el conjunto de datos VASAM de información personalizada de CARMA, CRASTRS, en el que las series están separadas por un carácter delimitador en lugar de tener una longitud fija.

Personalice y someta el JCL FEK.SFEKSAMP(CRA#VS2) JCL para convertir el conjunto de datos VSAM de información personalizada de CARMA de longitud fija existente, CRASTRS, en el formato de longitud variable nueva.

### Nota:

- Empezando por la versión 7.6.1, el conjunto de datos VSAM de información personalizada de CARMA se envía en formato de longitud variable.
- A partir de la versión 7.6.1, el módulo de carga de CARMA, CRASERV, soporta el formato de longitud fija y el formato de longitud variable para el conjunto de datos VSAM de información personalizada de CARMA.
- Las versiones más antiguas del módulo de carga de CARMA no soportan el formato de longitud variable y generarán series confusas cuando se utilicen con un conjunto de datos VSAM de información personalizada de CARMA.

---

## Interfaz RSE con CARMA

El servidor CARMA suministra una API estándar para que otros productos basados en host-based puedan acceder a uno o varios SCM (Software Configuration Manager). Sin embargo, no suministra métodos para la comunicación directa con un PC cliente. Para ello se basa en otros productos, como por ejemplo el servidor RSE. El servidor RSE utiliza los valores de CRASRV.properties para iniciar un servidor CARMA y conectarse a él.

CRASRV.properties se encuentra en /etc/rdz/, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: "Configuración de la personalización" en la página 15. Puede editar el archivo con el mandato TSO OEDIT.

**Nota:** Es necesario reiniciar la tarea iniciada RSED para recoger los cambios que haya realizado.

```
# CRASRV.properties - Opciones de configuración de CARMA
#
port.start=5227
port.range=100
startup.script.name=/usr/lpp/rdz/bin/carma.startup.rex
clist.dsname='FEK.#CUST.CNTL(CRASUBMT)'
crastart.stub=/usr/lpp/rdz/bin/CRASTART
crastart.configuration.file=/etc/rdz/crastart.conf
crastart.syslog=Partial
crastart.timeout=420
#crastart.steplib=FEK.SFEKLPA
#crastart.tasklib=TASKLIB
```

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

#### **port.start**

Primer puerto utilizado para la comunicación entre CARMA y el servidor RSE. 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 151.

#### **port.range**

Rango de puertos, empezando por `port.start`, que se utilizará para la comunicación de 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 inicio de CARMA. El valor predeterminado es `/usr/lpp/rdz/bin/carma.startup.rex`. Este exec REXX desencadenará el inicio de un servidor CARMA.

#### **clist.dsname**

Define el método de inicio del servidor CARMA.

- **\*CRASTART** indica que el servidor CARMA debe iniciarse como subtask de RSE mediante CRASTART. Encontrará más detalles en: “(Opcional) Inicio alternativo del servidor CARMA mediante CRASTART” en la página 57. Si especifica **\*CRASTART**, también debe especificar las directivas `crastart.*`.
- **\*ISPF** indica que el servidor CARMA debe iniciarse mediante la Pasarela de cliente TSO/ISPF de ISPF. Encontrará más detalles en: “(Opcional) Inicio alternativo del servidor CARMA mediante la Pasarela de cliente TSO/ISPF” en la página 60.
- Cualquier otro valor define la ubicación de la CLIST CRASUBMT, mediante convenios de denominación de tipo TSO. Si se especifica entre comillas simples (‘), el nombre del conjunto de datos es una referencia absoluta, sin comillas simples (‘) el nombre del conjunto de datos se prefija con el ID de usuario de los clientes, no con el prefijo de TSO. Este último procedimiento requiere que todos los usuarios de CARMA mantengan su propia CLIST CRASUBMT.

El valor predeterminado es 'FEK.#CUST.CNTL(CRASUBMT)'. Esta CLIST iniciará un servidor CARMA cuando se abra una conexión mediante el método de sometimiento por lotes.

#### **crastart.stub**

Apéndice de z/OS UNIX para llamar a CRASTART. El valor predeterminado es /usr/lpp/rdz/bin/CRASTART. Este apéndice pone el módulo de carga CRASTART basado en MVS a disposición de los procesos de z/OS UNIX. Esta directiva sólo se utiliza si la directiva `clist.dsname` tiene el valor \*CRASTART.

#### **crastart.configuration.file**

Especifica el nombre del archivo de configuración de CRASTART. El valor predeterminado es /etc/rdz/crastart.conf. Este archivo especifica las asignaciones de conjunto de datos e invocaciones de programa necesarias para iniciar un servidor CARMA. Esta directiva sólo se utiliza si la directiva `clist.dsname` tiene el valor \*CRASTART.

#### **crastart.syslog**

Especifica cuánta información se graba en las anotaciones del sistema mientras CRASTART inicia un servidor CARMA. El valor predeterminado es Parcial. Los valores válidos son:

A (Todo)	Toda la información de rastreo se graba en SYSLOG
P (Parcial)	Sólo la información de conexión, desconexión y errores se graba en SYSLOG
resto de información	Sólo las condiciones de error se graban en SYSLOG

Esta directiva sólo se utiliza si la directiva `clist.dsname` tiene el valor \*CRASTART.

#### **crastart.timeout**

Tiempo, en segundos, que debe transcurrir antes de que un servidor CARMA finalice debido a la falta de actividad. El valor predeterminado es 420 (7 minutos). Esta directiva sólo se utiliza si la directiva `clist.dsname` tiene el valor \*CRASTART.

#### **crastart.steplib**

La ubicación del módulo CRASTART cuando se accede a través de la directiva STEPLIB de `rsed.envvars`. El valor predeterminado es FEK.SFEKLPA. Descomente y personalice esta directiva si el módulo CRASTART no puede formar parte de LPA o LINKLIST. Tenga en cuenta que pueden surgir problemas de control de programa y de APF si el módulo CRASTART no se encuentra en LPA. Esta directiva sólo se utiliza si la directiva `clist.dsname` tiene el valor \*CRASTART.

#### **crastart.tasklib**

Nombre alternativo del nombre de DD TASKLIB de `crastart.conf`. El valor predeterminado es TASKLIB. Descomente y personalice esta directiva si el nombre de DD TASKLIB tiene un significado especial para el SCM o RAM y no puede utilizarse como sustitución de STEPLIB. Esta directiva sólo se utiliza si la directiva `clist.dsname` tiene el valor \*CRASTART.

---

## **Inicio del servidor CARMA mediante el sometimiento por lotes**

La información de esta sección describe cómo configurar el método predeterminado para que Developer for System z inicie un servidor CARMA. Este paso de personalización puede pasarse por alto si se utiliza otro método de inicio.

Developer for System z utiliza, por omisión, el método de inicio del servidor CARMA de sometimiento por lotes que no requiere que el módulo CRASTART se encuentre en LPA y que no depende de la Pasarela de cliente TSO/ISPF. El método somete el servidor CARMA como un trabajo por lotes de larga ejecución en su JES.

## Ajustar CRASRV.properties

El servidor RSE utiliza los valores de `/etc/rdz/CRASRV.properties` para iniciar un servidor CARMA y conectarse a él, como se indica en la sección “Interfaz RSE con CARMA” en la página 53. Puede editar el archivo con el mandato TSO **OEDIT**. Tenga en cuenta que RSE debe reiniciarse para que los cambios entren en vigor.

Cambie el valor de la directiva `clist.dsname` por el conjunto de datos y el nombre de miembro de la CLIST de inicio del servidor CARMA `CRASUBMT`, como se muestra en el ejemplo siguiente. Consulte la sección “Interfaz RSE con CARMA” en la página 53 para obtener más información sobre las diferentes directivas.

```
port.start=5227
port.range=100
startup.script.name=/usr/lpp/rdz/bin/carma.startup.rex
clist.dsname='FEK.#CUST.CNTL(CRASUBMT)'
```

*Figura 11. CRASRV.properties - Inicio de CARMA mediante sometimiento por lotes*

## Ajustar CRASUBMT

Personalice la CLIST de `CRASUBMT`, como se muestra en el ejemplo de código siguiente. Las instrucciones de personalización están en la documentación de `CRASUBMT`. La CLIST de `CRASUBMT` somete un servidor CARMA.

`CRASUBMT` se encuentra en `FEK.#CUST.CNTL`, a menos que haya especificado otra ubicación al personalizar y someter el trabajo `FEK.SFEKSAMP(FEKSETUP)`. Encontrará más detalles en: “Configuración de la personalización” en la página 15.

```

PROC 1 PORT TIMEOUT(420)
SUBMIT * END($$)
//CRA&PORT JOB CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1)
//RUN      EXEC PGM=IKJEFT01,DYNAMNBR=25,REGION=1024K,TIME=NOLIMIT
//STEPLIB DD DISP=SHR,DSN=FEK.SFEKLOAD
//*        DD DISP=SHR,DSN=FEK.#CUST.LOAD
//CRADEF   DD DISP=SHR,DSN=FEK.#CUST.CRADEF
//CRAMSG   DD DISP=SHR,DSN=FEK.#CUST.CRAMSG
//CRASTRS  DD DISP=SHR,DSN=FEK.#CUST.CRASTRS
//*CRARAM1 DD DISP=SHR,DSN=FEK.#CUST.CRARAM1
//*
//ISPPROF DD DISP=(NEW,DELETE,DELETE),
//          SPACE=(TRK,(1,1,5)),LRECL=80,RECFM=FB,UNIT=SYSALLDA
//ISPMLIB DD DISP=SHR,DSN=ISP.SISPMENU
//ISPPLIB DD DISP=SHR,DSN=ISP.SISPPENU
//ISPSLIB DD DISP=SHR,DSN=ISP.SISPSENU
//ISPTLIB DD DISP=SHR,DSN=ISP.SISPTENU
//ISPEXEC DD DISP=SHR,DSN=ISP.SISPEXEC
//SYSPROC DD DISP=SHR,DSN=ISP.SISPCLIB
//*
//CARMALOG DD SYSOUT=*
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
ISPSTART PGM(CRASERV) PARM(&PORT &TIMEOUT)
//*
$$
EXIT CODE(0)

```

*Figura 12. CRASUBMT - Inicio de CARMA mediante sometimiento por lotes*

**Nota:**

- Puede añadir sus propias sentencias de tipo DD y concatenaciones de conjunto de datos para personalizar el entorno TSO, imitando un procedimiento de inicio de sesión de TSO.
- 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 FEK.#CUST.CNTL(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).
- En el archivo rsecomm.log se ofrecen detalles del proceso de inicio de CARMA. Consulte la sección “(Opcional) Rastreo de RSE” en la página 96 para obtener más información acerca de cómo establecer el nivel de detalle de rsecomm.log.
- Los cambios están en vigor para todos los servidores de CARMA iniciados tras la actualización.

## **(Opcional) Inicio alternativo del servidor CARMA mediante CRASTART**

La información de esta sección describe cómo configurar un método alternativo para que Developer for System z inicie un servidor CARMA. Este paso de personalización puede pasarse por alto si se utiliza otro método de inicio.

Developer for System z da soporte a un método de inicio alternativo del servidor CARMA que no depende de la Pasarela de cliente TSO/ISPF y que no somete un trabajo servidor mediante un iniciador de JES. El método utiliza CRASTART para iniciar el servidor CARMA como subtask de RSE y es similar al servicio de Pasarela de cliente TSO/ISPF.

**Nota:** En el archivo `rsecomm.log` se ofrecen detalles del proceso de inicio de CARMA. Consulte la sección “(Opcional) Rastreo de RSE” en la página 96 para obtener más información acerca de cómo establecer el nivel de detalle de `rsecomm.log`.

## Ajustar CRASRV.properties

El servidor RSE utiliza los valores de `/etc/rdz/CRASRV.properties` para iniciar un servidor CARMA y conectarse a él, como se indica en la sección “Interfaz RSE con CARMA” en la página 53. Puede editar el archivo con el mandato TSO **OEDIT**. Tenga en cuenta que RSE debe reiniciarse para que los cambios entren en vigor.

Cambie el valor de la directiva `clist.dsname` por `*CRASTART` y suministre los valores correctos para las directivas `crastart.*`, como se muestra en el ejemplo siguiente. Consulte la sección “Interfaz RSE con CARMA” en la página 53 para obtener más información sobre las diferentes directivas.

```
port.start=5227
port.range=100
startup.script.name=/usr/lpp/rdz/bin/carma.startup.rex
clist.dsname=*CRASTART
crastart.stub=/usr/lpp/rdz/bin/CRASTART
crastart.configuration.file=/etc/rdz/crastart.conf
crastart.syslog=Partial
crastart.timeout=420
#crastart.steplib=FEK.SFEKLPA
#crastart.tasklib=TASKLIB
```

*Figura 13. CRASRV.properties - Inicio alternativo de CARMA con \*CRASTART*

**Nota:** En caso que el parámetro `JWT` del miembro `parmlib SMFPRMxx` sea menor que el valor de tiempo de espera de `CRASRV.properties` se producirá la terminación anormal del sistema 522 del módulo `ISPZTS0`. Ello no afecta a las operaciones de CARMA, ya que el servidor se reinicia automáticamente si es necesario.

## Ajustar crastart.conf

Tenga a mano una copia impresa del `CRASUBMT` personalizado (consulte la sección “Inicio del servidor CARMA mediante el sometimiento por lotes” en la página 55) para poder consultarlo con facilidad durante este paso de configuración. La copia impresa será de utilidad aunque no haya personalizado el miembro.

`CRASTART` utiliza las definiciones de `crastart.conf` para crear un entorno válido para ejecutar mandatos TSO e ISPF por lotes. Developer for System z utiliza este entorno para ejecutar el servidor CARMA `CRASERV`.

`crastart.conf` se encuentra en `/etc/rdz/`, a menos que haya especificado otra ubicación al personalizar y someter el trabajo `FEK.SFEKSAMP(FEKSETUP)`. Encontrará más detalles en: “Configuración de la personalización” en la página 15. Puede editar el archivo con el mandato TSO **OEDIT**.

**Nota:** Los cambios están en vigor para todos los servidores de CARMA iniciados tras la actualización.

Son necesarios los siguientes pasos de personalización para ajustar el archivo de configuración que se muestra en el ejemplo de código que figura más adelante.

- Añada los conjuntos de datos asignados a la concatenación `STEPLIB` del procedimiento `CRASUBMT` a la sentencia `TASKLIB` de `crastart.conf`.

- Cree entradas para las DD VSAM de CARMA obligatorias CRADEF, CRAMSG y CRSTRS. Utilice los nombres de conjunto de datos suministrados en el procedimiento CRASUBMT (personalizado).
- Añada las sentencias de DD personalizadas y las concatenaciones de conjuntos de datos relacionadas, disponibles en el procedimiento CRASUBMT (personalizado). Por ejemplo, añada la sentencia DD CRARAM1 y el nombre de conjunto de datos si utiliza el RAM PDS de ejemplo. Tenga en cuenta que puede utilizar nombres de conjunto de datos (asignados con DISP=SHR) y construcciones SYSOUT y DUMMY.
- Opcionalmente, añada un mandato BPXWDYN mediante la sentencia -COMMAND. Puede haber varias sentencias -COMMAND. BPXWDYN permite realizar asignaciones más complejas, que incluyen la creación de conjuntos de datos temporales, disposiciones que no sean SHR, asignaciones a otros subsistemas, etc. Consulte la publicación *Using REXX and z/OS UNIX System Services* (SA22-7806) para obtener más información acerca de BPXWDYN.
- Seleccione el método de invocación de programa que desee mediante la sentencia PROGRAM. El método recomendado es "PROGRAM=IKJEFT01 CRASERV &CRAPRM1. &CRAPRM2.", ya que proporciona un entorno TSO que puede manejar una mezcla de conjuntos de datos APF y no APF. Consulte el ejemplo crastart.conf para conocer otros métodos.

**Nota:** Las definiciones de crastart.conf no pueden dividirse en varias líneas.

\* crastart.conf - Opciones de asignación de CARMA

```
TASKLIB = FEK.SFEKLOAD
CRADEF = FEK.#CUST.CRADEF
CRAMSG = FEK.#CUST.CRAMSG
CRASTRS = FEK.#CUST.CRASTRS
*CRARAM1 = FEK.#CUST.CRARAM1
*
CARMALOG = SYSOUT(H)
SYSTSPRT = SYSOUT(H)
SYSTSIN = DUMMY
-COMMAND=ALLOC FI(SCRATCH) NEW DELETE DSORG(PS) RECFM(F,B) LRECL(80) UNIT(VIO)
*
PROGRAM=IKJEFT01 CRASERV &CRAPRM1. &CRAPRM2.
```

Figura 14. crastart.conf - Inicio alternativo de CARMA con \*CRASTART

Pueden utilizarse las variables siguientes en el archivo de configuración:

Tabla 10. variables de crastart.conf

&CRAUSER.	ID de usuario de inicio de sesión del cliente.
&CRADATE.	Fecha actual en el formato Daaaaddd (Juliano de 7 caracteres).
&CRATIME.	Hora actual en el formato Thhmmss (horas, minutos y segundos).

Tabla 10. variables de crastart.conf (continuación)

&CRAPRM3. a &CRAPRM9.	<p>Variables adicionales con valores asignados por el usuario. La utilización de estas variables requiere la personalización del REXX de inicio de CARMA al que se hace referencia en startup.script.name de CRASRV.properties.</p> <p>Cuando utiliza estas variables, debe personalizar una copia del REXX de inicio predeterminado, /usr/lpp/rdz/bin/carma.startup.rex y hace que startup.script.name apunte a esta copia. Esto es para evitar que pierda su trabajo cuando el mantenimiento actualice el REXX predeterminado.</p>
símbolo del sistema	Cualquier símbolo del sistema definido en SYS1.PARMLIB(IEASYMxx)
-<DD>	Un guión (-) seguido de un nombre DD definido anteriormente actúa como retrorreferencia *.ddname de JCL. La DD original debe asignarse mediante la sentencia -COMMAND.

**Nota:** No existe ninguna variable para el prefijo TSO, ya que TSO no está activo cuando se interpreta el archivo de configuración.

## (Opcional) Inicio alternativo del servidor CARMA mediante la Pasarela de cliente TSO/ISPF

La información de esta sección describe cómo configurar un método alternativo para que Developer for System z inicie un servidor CARMA. Este paso de personalización puede pasarse por alto si se utiliza otro método de inicio.

Developer for System z da soporte a un método de inicio alternativo del servidor CARMA que no requiere que el módulo CRASTART se encuentre en LPA y que no somete un trabajo servidor mediante un iniciador de JES. El método utiliza la Pasarela de cliente TSO/ISPF de ISPF y es similar al procedimiento predeterminado para acceder al servicio de mandatos TSO.

**Nota:** En el archivo rsecomm.log se ofrecen detalles del proceso de inicio de CARMA. Consulte la sección “(Opcional) Rastreo de RSE” en la página 96 para obtener más información acerca de cómo establecer el nivel de detalle de rsecomm.log.

### Ajustar CRASRV.properties

El servidor RSE utiliza los valores de /etc/rdz/CRASRV.properties para iniciar un servidor CARMA y conectarse a él, como se indica en la sección “Interfaz RSE con CARMA” en la página 53. Puede editar el archivo con el mandato TSO OEDIT. Tenga en cuenta que RSE debe reiniciarse para que los cambios entren en vigor.

Cambie el valor de la directiva clist.dsname por \*ISPF, como se muestra en el ejemplo siguiente. Consulte la sección “Interfaz RSE con CARMA” en la página 53 para obtener más información sobre las diferentes directivas.

```
port.start=5227
port.range=100
startup.script.name=/usr/lpp/rdz/bin/carma.startup.rex
clist.dsname=*ISPF
```

Figura 15. CRASRV.properties - Inicio alternativo de CARMA con \*ISPF

## Ajustar ISPF.conf

Tenga a mano una copia impresa del CRASUBMT personalizado (consulte la sección “Inicio del servidor CARMA mediante el sometimiento por lotes” en la página 55) para poder consultarlo con facilidad durante este paso de configuración. La copia impresa será de utilidad aunque no haya personalizado el miembro.

La Pasarela de cliente TSO/ISPF de ISPF utiliza las definiciones de ISPF.conf para crear un entorno válido para ejecutar mandatos TSO e ISPF por lotes. Developer for System z utiliza este entorno para ejecutar el servidor CARMA.

ISPF.conf se encuentra en /etc/rdz/, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15. Puede editar el archivo con el mandato TSO **OEDIT**.

**Nota:** Los cambios están en vigor para todos los servidores de CARMA iniciados tras la actualización.

Son necesarios los siguientes pasos de personalización para ajustar el archivo de configuración que se muestra en el ejemplo de código que figura más adelante.

- Especifique los nombres correctos para los conjuntos de datos ISPF obligatorios (pero no asigne ISPPROF, ya que éste se asigna dinámicamente).
- Añada la proclib de Developer for System z, FEK.SFEKPROC, a la sentencia SYSPROC o SYSEXEC para que el sistema pueda encontrar el exec de CRASRVI. Este exec inicia el servidor CARMA (y por tanto sustituye la DD SYSTSIN de CRASUBMT).
- Añada la concatenación de DD STEPLIB del procedimiento CRASUBMT a la sentencia ispllib.
- Cree entradas para las DD VSAM de CARMA obligatorias CRADEF, CRAMSG y CRSTRS. Utilice los nombres de conjunto de datos suministrados en el procedimiento CRASUBMT (personalizado).
- Añada las sentencias de DD personalizadas y la concatenación de conjunto de datos relacionado, disponible en el procedimiento CRASUBMT (personalizado). Por ejemplo, añada la sentencia DD CRARAM1 y el nombre de conjunto de datos si utiliza el RAM PDS de ejemplo. Tenga en cuenta que sólo puede utilizar nombres de conjunto de datos (asignados con DISP=SHR).
- Opcionalmente, puede descomentar y personalizar la directiva allocexec para realizar asignaciones adicionales mediante un exec.

**Nota:** No incluya las DD SYSTSIN, SYSTSOUT ni CARMALOG ni ninguna otra sentencia DD que utilice construcciones JES como datos de corriente de entrada y SYSOUT=. Estas entradas deben convertirse para utilizar conjuntos de datos.

```

sysproc=ISP.SISPLIB,FEK.SFEKPROC
isp1lib=FEK.SFEKLOAD
ispmlib=ISP.SISPMENU
isptlib=ISP.SISPTENU
ispplib=ISP.SISPPENU
ispplib=ISP.SISPSLIB
CRADEF =FEK.#CUST.CRADEF
CRAMSG =FEK.#CUST.CRAMSG
CRASTRS=FEK.#CUST.CRASTRS
*CRARAM1=FEK.#CUST.CRARAM1
allocjob=FEK.#CUST.CNTL(CRAISPRX)

```

Figura 16. ISPF.conf - Inicio alternativo de CARMA con \*ISPF

DD CARMALOG hace referencia por omisión a SYSOUT=\*, que no puede correlacionarse en ISPF.conf. No puede correlacionar las DD directamente con un conjunto de datos cualquiera, ya que todos los usuarios de Developer for System z van a utilizar el mismo archivo ISPF.conf y, por tanto, los mismos conjuntos de datos.

Sin embargo, tal como se describe en Capítulo 16, “Personalizar el entorno TSO”, en la página 265, sección “Avanzado – Utilizar un exec de asignación” en la página 267, puede utilizar un exec de asignación para crear y asignar un conjunto de datos en función del ID de usuario activo. Consulte el miembro de ejemplo CRAISPRX del conjunto de datos FEK.#CUST.CNTL para obtener un ejemplo de cómo asignar la DD CARMALOG al nombre de conjunto de datos TSOPREFIX'. 'USERID'.CRA.'TIMESTAMP'.CARMALOG'.

**Nota:**

- Al utilizar la directiva allocjob, tenga cuidado de no deshacer las definiciones DD creadas antes en ISPF.conf.
- En caso de que el parámetro JWT del miembro parmlib SMFPRMxx sea menor que el valor ISPF\_timeout de ISPF.conf se esperará la terminación anormal del sistema 522 del módulo CRASERV. Ello no afecta a las operaciones de CARMA, ya que el servidor se reinicia automáticamente si es necesario.

---

## (Opcional) Activar los RAM (Repository Access Managers) de ejemplo

Los RAM (Repository Access Managers) son API escritas por el usuario para intercambiar información con los SCM (Software Configuration Managers) 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.

Consulte la publicación *Rational Developer for System z Common Access Repository Manager Developer's Guide* (SC23-7660) para obtener más información acerca de los RAM de ejemplo y el código fuente de ejemplo que se suministran.

Los siguientes miembros de ejemplo se encuentran en FEK.#CUST.JCL, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15.

## Activar los RAM de PDS

El RAM de PDS proporciona una lista de conjuntos de datos similar a **MVS Archivos -> Mis conjuntos de datos** en la vista Sistemas remotos. El RAM de PDS utiliza el ID de RAM 0 de forma predeterminada.

**Nota:** El RAM de PDS espera que CARMA se inicie dentro de ISPF (utilizando ISPSTART).

1. Personalice y someta el JCL FEK.#CUST.JCL(CRA#VPDS). Las instrucciones de personalización se encuentran en la documentación de CRA#VPDS. CRA#VPDS crea y prepara el VSAM de mensajes de RAM de PDS.
2. Añada la sentencia DD CRARAM1 al método de inicio de CARMA seleccionado y especifique el nombre de conjunto de datos del VSAM de mensajes del RAM de PDS.

## Activar los RAM de SCLM

El RAM de SCLM le proporciona una entrada básica a SCLM, el Software Configuration Manager de ISPF. El RAM de SCLM utiliza el ID de RAM 1 de forma predeterminada.

**Nota:** El RAM de SCLM espera que CARMA se inicie dentro de ISPF (utilizando ISPSTART).

1. Personalice y someta el JCL FEK.#CUST.JCL(CRA#VSLM). Las instrucciones de personalización están en la documentación de CRA#VSLM. CRA#VSLM crea y prepara el conjunto de datos VSAM de mensajes RAM de SCLM.
2. Añada la sentencia DD CRARAM2 al método de inicio de CARMA seleccionado y especifique el nombre de conjunto de datos del VSAM de mensajes del RAM de SCLM.
3. Personalice el JCL FEK.#CUST.JCL(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 FEK.#CUST.JCL(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 los RAM de esqueleto

El RAM de esqueleto proporciona una infraestructura de esqueleto que se puede utilizar para desarrollar sus propias RAM. El RAM de esqueleto utiliza el ID de RAM 3 de forma predeterminada.

1. Personalice y someta el JCL FEK.#CUST.JCL(CRA#CRAM). Las instrucciones de personalización están en la documentación de CRA#CRAM. CRA#CRAM compila el RAM de esqueleto.
2. Añada la biblioteca de carga que contiene el módulo RAM de esqueleto compilado, CRARAMSA, a la DD STEPLIB del método de inicio de CARMA seleccionado (DD TASKLIB para el método CRASTART).

---

## (Opcional) Activar CA Endevor® SCM RAM

La interfaz de IBM® Rational® Developer for System z con el Gestor de configuraciones de software de CA Endevor® proporciona a los clientes de Developer for System z acceso directo a CA Endevor® SCM. En adelante, nos referiremos a la interfaz de IBM® Rational® Developer for System z con CA Endevor® SCM como CA Endevor® SCM RAM (Gestor de acceso a repositorios).

Al contrario que los RAM de ejemplo documentados en esta publicación, CA Endeavor® SCM RAM es un RAM de tipo de producción. No debe activar ambos tipos de RAM en la misma configuración.

Atención: los trabajos de configuración proporcionados para CA Endeavor® SCM RAM sustituyen la configuración de CARMA activa con una que contiene únicamente CA Endeavor® SCM RAM.

**Nota:** El método de inicio de la Pasarela de cliente TSO/ISPF no se puede utilizar junto a CA Endeavor® SCM RAM.

## Requisitos y lista de comprobación

Necesita ayuda de un administrador de seguridad y de un administrador de TCP/IP para realizar esta tarea de personalización, que requiere los siguientes recursos o tareas de personalización especiales:

- Rango de puertos TCP/IP para la comunicación interna
- (Opcional) Regla de seguridad para permitir a los usuarios someter trabajos CRA\*
- (Opcional) Actualización de LPA

A fin de empezar a utilizar CA Endeavor® SCM RAM en su sitio, debe realizar las tareas siguientes. A menos que se indique de otro modo, todas las tareas son obligatorias.

1. Asigne y de prioridad a conjuntos de datos de VSAM que definan CA Endeavor® SCM RAM para CARMA. Para obtener más detalles, consulte “Definir CA Endeavor® SCM RAM”.
2. Elija el método de inicio preferido, el sometimiento por lotes o CRASTART y haga la personalización necesaria de los archivos de configuración relacionados. Para obtener más detalles, consulte:
  - “Inicio de CA Endeavor® SCM RAM mediante el sometimiento por lotes” en la página 65
  - “Inicio de CA Endeavor® SCM RAM mediante CRASTART” en la página 67
3. También puede personalizar el exec de asignación utilizado para la asignación dinámica de conjuntos de datos específicos de usuario. Para obtener más detalles, consulte “(Opcional) Personalizar CRANDVRA” en la página 68.
4. También puede personalizar los archivos de configuración específica de CA Endeavor® SCM RAM. Para obtener más detalles, consulte “(Opcional) Personalizar CA Endeavor® SCM RAM” en la página 69.

## Definir CA Endeavor® SCM RAM

Los siguientes componentes de CARMA deben personalizarse, independientemente del método de inicio elegido. Los siguientes miembros de ejemplo se encuentran en FEK.#CUST.JCL, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15.

1. Personalice y someta el JCL FEK.#CUST.JCL(CRA#VCAD). Las instrucciones de personalización están en la documentación de CRA\$VDEF. CRA#VCAD crea y prepara el conjunto de datos VSAM de configuración de CARMA, CRADEF.
2. Personalice y someta el JCL FEK.#CUST.JCL(CRA\$VMSG). Las instrucciones de personalización están en la documentación de CRA\$VMSG. CRA\$VMSG crea y prepara el conjunto de datos VSAM de mensajes de CARMA, CRAMSG.

**Nota:** Este es el mismo trabajo que el de los RAM de ejemplo.

3. Personalice y someta el JCL FEK.#CUST.JCL(CRA#VCAS). Las instrucciones de personalización están en la documentación de CRA\$VSTR. CRA#VCAS crea y prepara el conjunto de datos VSAM de información de personalización de CARMA, CRASTRS.

**Nota:**

- CA Endevor® SCM RAM utiliza el ID 0 de RAM de forma predeterminada.
- Consulte el trabajo de ejemplo FEK.#CUST.JCL(CRA#UADD) si necesita fusionar las definiciones de un RAM (personalizada) en una configuración de VSAM existente. Es necesario personalizar y someter este trabajo por cada VSAM de CARMA modificado. Consulte la publicación *Rational Developer for System z Common Access Repository Manager Developer's Guide* (SC23-7660) para obtener más información acerca de la estructura de registro utilizada por los distintos VSAM de CARMA.
- Utilice el trabajo de ejemplo FEK.#CUST.JCL(CRA#UQRY) para extraer las definiciones activas de un VSAM a un conjunto de datos secuencial.

## Inicio de CA Endevor® SCM RAM mediante el sometimiento por lotes

No ejecute este paso si utiliza el método CRASTART para iniciar el servidor CARMA con CA Endevor® SCM RAM.

Developer for System z puede utilizar el método de inicio de servidor CARMA de sometimiento por lotes para iniciar CA Endevor® SCM RAM. El método somete el servidor CARMA como un trabajo por lotes de larga ejecución en su JES.

Consulte la sección “Inicio del servidor CARMA mediante el sometimiento por lotes” en la página 55 para obtener más información sobre el método de inicio de sometimiento por lotes.

### Ajustar CRASRV.properties

El servidor RSE utiliza los valores de /etc/rdz/CRASRV.properties para iniciar un servidor CARMA y conectarse a él, como se indica en la sección “Interfaz RSE con CARMA” en la página 53. Puede editar el archivo con el mandato TSO **OEDIT**. Tenga en cuenta que RSE debe reiniciarse para que los cambios entren en vigor.

Cambie el valor de la directiva `clist.dsname` por el conjunto de datos y el nombre de miembro de la CLIST de inicio del servidor CARMA CRASUBCA, como se muestra en el ejemplo siguiente. Consulte la sección “Interfaz RSE con CARMA” en la página 53 para obtener más información sobre las diferentes directivas.

```
port.start=5227
port.range=100
startup.script.name=/usr/lpp/rdz/bin/carma.startup.rex
clist.dsname='FEK.#CUST.CNTL(CRASUBCA)'
```

Figura 17. Figura x1. CRASRV.properties - Inicio de CA Endevor® SCM RAM mediante el sometimiento por lotes

### Ajustar CRASUBCA

Personalice la CLIST de CRASUBCA, como se muestra en el ejemplo de código siguiente. Las instrucciones de personalización están en la documentación de CRASUBCA. La CLIST CRASUBCA somete un servidor CARMA para CA Endevor® SCM.

CRASUBCA se encuentra en FEK.#CUST.CNTL, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15.

```

PROC 1 PORT TIMEOUT(420)
SUBMIT * END($$)
//CRA&PORT JOB CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1)
//RUN      EXEC PGM=IKJEFT01,DYNAMNBR=125,REGION=0M,TIME=NOLIMIT,
//          PARM='%CRANDVRA NDVRC1 PGM(CRASERV) PARM(&PORT &TIMEOUT)'
//STEPLIB DD DISP=SHR,DSN=FEK.SFEKLOAD
//          DD DISP=SHR,DSN=CA.NDVR.AUTHLIB
//          DD DISP=SHR,DSN=CA.NDVRU.AUTHLIB
//CRADEF DD DISP=SHR,DSN=FEK.#CUST.CRADEF
//CRMSG DD DISP=SHR,DSN=FEK.#CUST.CRMSG
//CRASTRS DD DISP=SHR,DSN=FEK.#CUST.CRASTRS
//*
//SYSPROC DD DISP=SHR,DSN=ISP.SISPCLIB
//          DD DISP=SHR,DSN=FEK.SFEKPROC
//ISPEXEC DD DISP=SHR,DSN=ISP.SISPEXEC
//ISPLIB DD DISP=SHR,DSN=ISP.SISPMENU
//ISPPLIB DD DISP=SHR,DSN=ISP.SISPPENU
//ISPSLIB DD DISP=SHR,DSN=ISP.SISPSENU
//ISPTLIB DD DISP=SHR,DSN=ISP.SISPTENU
//ISPTL0 DD DISP=(NEW,DELETE,DELETE),UNIT=SYSALLDA,
//          SPACE=(TRK,(1,1)),LRECL=80,RECFM=FB
//ISPTL1 DD DISP=(NEW,DELETE,DELETE),UNIT=SYSALLDA,
//          SPACE=(TRK,(1,1)),LRECL=80,RECFM=FB
//ISPPROF DD DISP=(NEW,DELETE,DELETE),UNIT=SYSALLDA,
//          SPACE=(TRK,(1,1,5)),LRECL=80,RECFM=FB
//*
//CARMALOG DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD DUMMY
//*
//CONLIB DD DISP=SHR,DSN=CA.NDVR.CONLIB
//JCLOUT DD SYSOUT=(A,INTRDR),DCB=(LRECL=80,RECFM=F,BLKSIZE=80)
//EXT1ELM DD DISP=(NEW,DELETE),UNIT=SYSALLDA,
//          RECFM=VB,LRECL=4096,BLKSIZE=27998,SPACE=(TRK,(5,5))
//EXT1DEP DD DISP=(NEW,DELETE),UNIT=SYSALLDA,
//          RECFM=VB,LRECL=4096,BLKSIZE=27998,SPACE=(TRK,(5,5))
//MSG3FILE DD DISP=(NEW,DELETE),UNIT=SYSALLDA,
//          RECFM=FB,LRECL=133,BLKSIZE=27930,SPACE=(TRK,(5,5))
//C1MSGSG1 DD DISP=(NEW,DELETE),UNIT=SYSALLDA,
//          RECFM=FB,LRECL=133,BLKSIZE=27930,SPACE=(TRK,(5,5))
//C1EXMSGSG DD DISP=(NEW,DELETE),UNIT=SYSALLDA,
//          RECFM=FB,LRECL=133,BLKSIZE=27930,SPACE=(TRK,(5,5))
//TYPEMAP DD DISP=SHR,DSN=FEK.#CUST.PARMLIB(CRATMAP)
//SHOWVIEW DD DISP=SHR,DSN=FEK.#CUST.PARMLIB(CRASHOW)
$$
EXIT CODE(0)

```

Figura 18. Figura x2. CRASUBCA - Inicio de CA Endeavor® SCM RAM mediante el sometimiento por lotes

**Nota:**

- Puede añadir sus propias sentencias de tipo DD y concatenaciones de conjunto de datos para personalizar el entorno TSO, imitando un procedimiento de inicio de sesión de TSO.
- También puede cambiar el valor de tiempo de espera de CARMA modificando la línea PROC 1 PORT TIMEOUT(420) en la CLIST. 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).

- En el archivo `rsecomm.log` se ofrecen detalles del proceso de inicio de CARMA. Consulte la sección “(Opcional) Rastreo de RSE” en la página 96 para obtener más información acerca de cómo establecer el nivel de detalle de `rsecomm.log`.
- Los cambios están en vigor para todos los servidores de CARMA iniciados tras la actualización.

## Inicio de CA Endeavor® SCM RAM mediante CRAFTSTART

No ejecute este paso si utiliza el método de sometimiento por lotes para iniciar el servidor CARMA con CA Endeavor® SCM RAM.

Developer for System z puede utilizar el método de inicio de servidor CARMA CRAFTSTART para iniciar CA Endeavor® SCM RAM. El método utiliza CRAFTSTART para iniciar el servidor CARMA como subtask dentro de RSE..

Consulte “(Opcional) Inicio alternativo del servidor CARMA mediante CRAFTSTART” en la página 57 para obtener más información sobre el método de inicio de CRAFTSTART.

**Nota:** En el archivo `rsecomm.log` se ofrecen detalles del proceso de inicio de CARMA. Consulte la sección “(Opcional) Rastreo de RSE” en la página 96 para obtener más información acerca de cómo establecer el nivel de detalle de `rsecomm.log`.

### Ajustar CRASRV.properties

El servidor RSE utiliza los valores de `/etc/rdz/CRASRV.properties` para iniciar un servidor CARMA y conectarse a él, como se indica en la sección “Interfaz RSE con CARMA” en la página 53. Puede editar el archivo con el mandato TSO **OEDIT**. Tenga en cuenta que RSE debe reiniciarse para que los cambios entren en vigor.

Cambie el valor de la directiva `clist.dsname` por `*CRAFTSTART` y suministre los valores correctos para las directivas `crastart.*`, como se muestra en el ejemplo siguiente. Consulte la sección “Interfaz RSE con CARMA” en la página 53 para obtener más información sobre las diferentes directivas.

```
port.start=5227
port.range=100
startup.script.name=/usr/lpp/rdz/bin/carma.startup.rex
clist.dsname=*CRAFTSTART
crastart.stub=/usr/lpp/rdz/bin/CRAFTSTART
crastart.configuration.file=/etc/rdz/crastart.endevor.conf
crastart.syslog=Partial
crastart.timeout=420
#crastart.steplib=FEK.SFEKLPA
#crastart.tasklib=TASKLIB
```

*Figura 19. Figura x3. CRASRV.properties - Inicio de CA Endeavor® SCM RAM mediante CRAFTSTART*

**Nota:** En caso que el parámetro `JWT` del miembro `parmlib SMFPRMxx` sea menor que el valor de tiempo de espera de `CRASRV.properties` se producirá la terminación anormal del sistema 522 del módulo `ISPZTS0`. Ello no afecta a las operaciones de CARMA, ya que el servidor se reinicia automáticamente si es necesario.

### Ajustar crastart.endevor.conf

CRAFTSTART utiliza las definiciones de `crastart.endevor.conf` para crear un entorno (TSO/ISPF) válido para invocar CA Endeavor® SCM. Developer for System z utiliza este entorno para ejecutar CA Endeavor® SCM RAM.

crastart.endevor.conf se encuentra en /etc/rdz/, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15. Puede editar el archivo con el mandato TSO OEDIT.

**Nota:** Los cambios están en vigor para todos los servidores de CARMA iniciados tras la actualización.

```
TASKLIB = FEK.SFEKLOAD,CA.NDVR.AUTHLIB,CA.NDVRU.AUTHLIB
CRADEF  = FEK.#CUST.CRADEF
CRAMSG  = FEK.#CUST.CRAMSG
CRASTRS = FEK.#CUST.CRASTRS

SYSPROC = ISP.SISPCLIB,FEK.SFEKPROC
SYSEXEC = ISP.SISPEXEC
ISPMLIB = ISP.SISPMENU
ISPPLIB = ISP.SISPPENU
ISPSLIB = ISP.SISPSENU
-COMMAND=ALLOC FI(ISPCTL0) NEW DELETE DSORG(PS) RECFM(F,B) LRECL(80)
BLKSIZE(6160) SPACE(2,2) TRACKS UNIT(SYSALLDA)
-COMMAND=ALLOC FI(ISPCTL1) NEW DELETE DSORG(PS) RECFM(F,B) LRECL(80)
BLKSIZE(6160) SPACE(2,2) TRACKS UNIT(SYSALLDA)
-COMMAND=ALLOC FI(ISPPROF) NEW DELETE DSORG(PO) DIR(5) RECFM(F,B) LRECL(80)
BLKSIZE(6160) SPACE(2,2) TRACKS UNIT(SYSALLDA)
ISPTLIB = -ISPPROF,ISP.SISPTENU
ISPTABL = -ISPPROF

CARMALOG = SYSOUT(H)
SYSPRINT= SYSOUT(H)
SYSTSPRT = SYSOUT(H)
SYSTSIN  = DUMMY

TYPEMAP = FEK.#CUST.PARMLIB(CRATMAP)
SHOWVIEW= FEK.#CUST.PARMLIB(CRASHOW)
CONLIB   = CA.NDVR.CONLIB
-COMMAND=ALLOC FI(JCLOUT) SYSOUT(A) WRITER(INTRDR) RECFM(F) LRECL(80)
BLKSIZE(80)
-COMMAND=ALLOC FI(EXT1ELM) NEW DELETE DSORG(PS) RECFM(V,B) LRECL(4096)
BLKSIZE(27998) SPACE(5,5) TRACKS UNIT(SYSALLDA)
-COMMAND=ALLOC FI(EXT1DEP) NEW DELETE DSORG(PS) RECFM(V,B) LRECL(4096)
BLKSIZE(27998) SPACE(5,5) TRACKS UNIT(SYSALLDA)
-COMMAND=ALLOC FI(MSG3FILE) NEW DELETE DSORG(PS) RECFM(F,B) LRECL(133)
BLKSIZE(27930) SPACE(5,5) TRACKS UNIT(SYSALLDA)
-COMMAND=ALLOC FI(C1EXMSG) NEW DELETE DSORG(PS) RECFM(F,B) LRECL(133)
BLKSIZE(27930) SPACE(5,5) TRACKS UNIT(SYSALLDA)
-COMMAND=ALLOC FI(C1MSG1) NEW DELETE DSORG(PS) RECFM(F,B) LRECL(133)
BLKSIZE(27930) SPACE(5,5) TRACKS UNIT(SYSALLDA)

PROGRAM=IKJEFT01 %CRANDVRA NDVRC1 PGM(CRASERV) PARM(&CRAPRM1.
&CRAPRM2.)
```

Figura 20. crastart.conf - Inicio de CA Endevor® SCM RAM mediante CRASTART

## (Opcional) Personalizar CRANDVRA

Ambos métodos de inicio, el de sometimiento por lotes y CRASTART invocan REXX exec CRANDVRA para asignar conjuntos de datos específicos de usuario utilizados por CA Endevor® SCM RAM.

DD	Nombre de conjunto de datos	Tipo
DEPEND	&SYSPREF..&SYSUID.. &SYSNAME..CRA\$NDVR.DEPEND	Permanente

DD	Nombre de conjunto de datos	Tipo
BROWSE	&SYSPREF..&SYSUID.. &SYSNAME..CRA\$NDVR.BROWSE	Temporal
C1PRINT	&SYSPREF..&SYSUID.. &SYSNAME..CRA\$NDVR.LISTING	Temporal

Puede personalizar una copia de este REXX exec si ciertos valores predeterminados, como por ejemplo el nombre del conjunto de datos, no cumplen los estándares del sitio. CRANDVRA está ubicado en FEK.SFEKPROC, a menos que haya utilizado un calificador de alto nivel diferente durante la instalación SMP/E de Developer for System z.

Las instrucciones de personalización están en la documentación de CRANDVRA.

**Nota:** Debe copiar el REXX de asignación de ejemplo a un conjunto de datos nuevo y personalizar esta copia para evitar que se sobrescriba al aplicar el mantenimiento. Cuando haga esto, debe actualizar la referencia a SFEKPROC en el SYSEXEC DD del método de inicio de CARMA elegido para que coincida con el nombre de conjunto de datos nuevo.

## (Opcional) Personalizar CA Endevor® SCM RAM

Los siguientes componentes de CARMA se pueden personalizar, independientemente del método de inicio elegido. Los siguientes miembros de ejemplo se encuentran en FEK.#CUST.PARMLIB, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15.

1. (Opcional) Personalizar FEK.#CUST.PARMLIB(CRASHOW). Consulte la documentación en CRASHOW para obtener instrucciones de personalización. CRASHOW define filtros predeterminados para entornos CA Endevor® SCM, sistemas, etc.
2. (Opcional) Personalizar FEK.#CUST.PARMLIB(CRATMAP). Las instrucciones de personalización están en la documentación de CRATMAP. CRATMAP altera temporalmente el tipo de CA Endevor® SCM a correlaciones de extensión de archivo.

## (Opcional) Soportar varios RAM

CARMA permite definir varios RAM y puede ejecutarlos simultáneamente. Sin embargo, puesto que sólo hay un servidor CARMA activo por usuario, incluso cuando haya varios RAM, es posible que sean necesarios algunos cambios de configuración para que esta configuración funcione.

Los RAM los define un desarrollador de RAM en el conjunto de datos VSAM de configuración de CARMA. CRADEF. Durante el inicio, el servidor CARMA, CRASERV, identificará todos los RAM definidos y presentará la información al cliente CARMA. El usuario puede seleccionar entonces uno o varios RAM que se cargarán en el servidor CARMA.

Puesto que los RAM están activos como plug-ins del servidor CARMA, debe asegurarse de que todos los prerequisites (como por ejemplo asignaciones de conjuntos de datos) para cada uno de los RAM están disponibles en el espacio de dirección del servidor CARMA. Para ello será necesario realizar cambios en los

ejemplos de configuración de CARMA, como por ejemplo CRASUBMT o crastart.conf, que se envían con Developer for System z.

## Ejemplo

En el ejemplo siguiente, empieza desde una configuración de CA Endevor® SCM RAM existente, mediante el método de inicio CRASTART y añade el PDS RAM de ejemplo.

Definiciones para CA Endevor® SCM RAM:

- FEK.SFEKVSM2(CRA0VCAD) - Definiciones de CRADEF
- FEK.SFEKVSM2(CRA0VCAS) - Definiciones de CRASTRS
- /etc/rdz/crastart.endevor.conf - Archivo de configuración de CRASTART

Definiciones para el PDS RAM:

- FEK.SFEKVSM2(CRA0VDEF) - Definiciones de CRADEF
- FEK.SFEKVSM2(CRA0VSTR) - Definiciones de CRASTRS
- FEK.#CUST.CRARAM1 - Definiciones de CRARAM1

El proceso empieza con un desarrollador de RAM recopilando los datos y la información necesarios para que el programador del sistema realice la configuración.

1. Extraiga los datos específicos de PDS RAM de los miembros SFEKVSM2 (estos miembros albergan definiciones para todos los RAM, no sólo PDS RAM).
2. Fusione estos datos con los miembros SFEKVSM2 de CA Endevor® SCM RAM.
3. Cree una lista de prerequisites específicos de PDS RAM:
  - DD CRARAM1, enlazado a FEK.#CUST.CRARAM1
  - Entorno TSO

El programador del sistema utiliza entonces estos datos para crear los conjuntos de datos CARMA VSAM actualizados y utiliza la información de prerequisite para crear un archivo de configuración CRASTART capaz de soportar ambos RAM.

1. Utilice los datos combinados como entrada de los trabajos CRA\$VDEF y CRA\$VSTR para crear los conjuntos de datos VSAM de información de configuración y personalización de CARMA, CRADEF y CRASTRS.
2. Añada una definición de CRARAM1 a crastart.endevor.conf:

```
CRARAM1 = FEK.#CUST.CRARAM1
```

3. Verifique la sentencia PROGRAM de crastart.endevor.conf para asegurarse de que sea capaz de proporcionar el entorno necesario para ambos RAM:

```
PROGRAM=IKJEFT01 %CRANDVRA NDVRC1 PGM(CRASERV)  
PARM(&CRAPRM1. &CRAPRM2.)
```

- IKJEFT01: TSO, utilizado para permitir determinadas llamadas autorizadas en un entorno no autorizado y utilizado como entorno para ejecutar el exec de preasignación de CA Endevor® SCM RAM.
- %CRANDVRA: CA Endevor® exec de preasignación de SCM RAM (ubicado en FEK.SFEKPROC), que asigna conjuntos de datos de trabajo específicos del usuario temporales (y permanentes).
- NDVRC1: componente de fondo de CA Endevor® que tiene un mecanismo incorporado para ejecutar mandatos TSO y ISPF.

- PGM(CRASERV): mandato para iniciar un servidor CARMA en formato de mandato ISPF
- PARM(&CRAPRM1. &CRAPRM2.): parámetros para CRASERV, en formato de mandato ISPF. &CRAPRM1 es el puerto a utilizar y &CRAPRM2 es el valor de tiempo de espera.

CA Endeavor® SCM RAM está activo en un entorno ISPF, lo que implica que el entorno TSO necesario para PDS RAM también está disponible.

---

## (Opcional) Comparación entre IRXJCL y CRAXJCL

Si el servidor CARMA se ha iniciado mediante TSO (IKJEFTxx), puede experimentar problemas si los RAM llaman a servicios que, a su vez, llaman a IRXJCL, la interfaz de proceso por lotes de REXX. El problema puede producirse cuando los procesadores llamados por los RAM se ejecutaron anteriormente sin TSO o sólo en TSO en línea, y se asignaban dinámicamente DD SYSTSIN o SYSTSPRT. Se suministra un programa de ejemplo, CRAXJCL, para solucionar este problema.

El procesador puede fallar si intenta asignar SYSTSIN o SYSTSPRT (necesarias para IRXJCL) debido a que el TSO por lotes (necesario para CARMA) ya tiene asignados y abiertos esos nombres de DD. El módulo de sustitución CRAXJCL intenta asignar SYSTSIN y SYSTSPRT a DUMMY, pero ignora los errores producidos si fallan las asignaciones.

Esto significa que, cuando los procesadores se ejecutan en un entorno CARMA iniciado por TSO, las asignaciones a SYSTSIN y SYSTSPRT son las mismas que las utilizadas por CARMA. Cuando los procesadores se ejecutan fuera de TSO/CARMA, CRAXJCL creará las asignaciones de SYSTSIN y SYSTSPRT. Por tanto, los procesadores no deben confiar en el contenido del conjunto de datos asignado a SYSTSIN.

Se presupone que las llamadas a IRXJCL utilizan el campo PARM para pasar el nombre y los parámetros de inicio de REXX, como se describe en la publicación *TSO/E REXX Reference* (SA22-7790). Esto significa que CARMA puede utilizar SYSTSIN de forma segura. Cualquier salida enviada a SYSTSPRT por IRXJCL finalizará en las anotaciones de CARMA.

Los procesadores que llaman al módulo de sustitución CRAXJCL no deben intentar asignar las DD SYSTSIN o SYSTSPRT antes de llamar a CRAXJCL.

## Crear CRAXJCL

El módulo de sustitución CRAXJCL se suministra en formato fuente, ya que deberá personalizarlo para especificar las asignaciones específicas que desea utilizar para SYSTSPRT. Generalmente, SYSTSIN debe asignarse a un conjunto de datos ficticio.

Se suministran código fuente de ensamblado de ejemplo y un trabajo de compilación/enlace de ejemplo como FEK.#CUST.ASM(CRAXJCL) y FEK.#CUST.JCL(CRA#CIRX), respectivamente, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15.

Personalice el código fuente de ensamblado de CRAXJCL según sus necesidades utilizando la documentación del miembro. Después, personalice y someta el JCL de

CRA#CIRX para crear el módulo de carga CRAXJCL. Las instrucciones de personalización se encuentran en la documentación de CRA#CIRX.

---

## Capítulo 4. (Opcional) Gestor de despliegue de aplicaciones (ADM)

Developer for System z utiliza determinadas funciones del Gestor de despliegue de aplicaciones como procedimiento de despliegue común para varios componentes. Los pasos de personalización indicados en este capítulo son necesarios si los desarrolladores utilizan alguna de las funciones siguientes:

- Enterprise Service Tools (EST)
- Diseñador de pantallas BMS
- Diseñador de pantallas MFS
- Generación de código CICSTS

**Nota:** Enterprise Service Tools (EST) abarca varias herramientas, como SFM (Modelador de flujo de servicios) y XSE (Servicios XML para la empresa).

Personalizar el Gestor de despliegue de aplicaciones Manager añade el servidor CDR (definiciones de recursos CICS), que se ejecuta como una aplicación CICS de z/OS para soportar estas funciones:

- Consultas de recursos CICS
- Solicitudes de instalación y desinstalación de recursos CICS en entornos CICSplex SM y no - CICSplex SM
- Solicitudes de introducción progresiva de programa y conjunto de mapas
- Solicitudes de exploración de conducto
- Solicitudes de exportación, importación y actualización de manifiestos

Los administradores de CICS pueden encontrar más información acerca del servidor CRD en Capítulo 15, “consideraciones de CICSTS”, en la página 253.

---

### Requisitos y lista de comprobación

Necesitará ayuda de un administrador de CICS, de un administrador de TCP/IP y de un administrador de seguridad para realizar esta tarea de personalización, que requiere los siguientes recursos o tareas de personalización especiales:

- Puerto TCP/IP para comunicación externa
- Actualizar JCL de región CICS
- Actualizar CSD de región CICS
- Definir grupo en la región CICS
- Regla de seguridad para permitir que los administradores realicen actualizaciones en un VSAM del Gestor de despliegue de aplicaciones
- Configuración de seguridad de CICSTS
- (Opcional) Definir nombres de transacción CICS
- (Opcional) Regla de seguridad para permitir que los usuarios realicen actualizaciones en un VSAM del Gestor de despliegue de aplicaciones

A fin de empezar a utilizar el Gestor de despliegue de aplicaciones en su sitio, debe realizar las tareas siguientes. A menos que se indique de otro modo, todas las tareas son obligatorias.

1. Crear el repositorio CRD. Para obtener más detalles, consulte “Repositorio CRD”.
2. Elija la interfaz CICS (RESTful o Servicio Web) para utilizarla. (Las interfaces puede coexistir). Para obtener más detalles, consulte “RESTful versus Servicio Web” en la página 75.
3. Si lo desea, realice las personalizaciones específicas de RESTful. Para obtener más detalles, consulte “Servidor CRD utilizando la interfaz RESTful” en la página 75.
  - Definir el servidor CRD en la región de conexión primaria CICS
  - De forma opcional, definir el servidor CRD en las regiones de conexión no primarias CICS.
  - De forma opcional, personalizar los ID de transacción del servidor CRD.
4. Si lo desea, realice las personalizaciones específicas del servicio Web. Para obtener más detalles, consulte “Servidor CRD utilizando la interfaz de servicios Web” en la página 77.
  - Añada el manejo de mensajes de conducto (posiblemente personalizado) a la concatenación RPL de CICS.
  - Definir el servidor CRD en la región de conexión primaria CICS.
  - De forma opcional, definir el servidor CRD en las regiones de conexión no primarias CICS.
5. De forma opcional, crear el repositorio de manifiestos. Para obtener más detalles, consulte “(Opcional) Repositorio de manifiesto” en la página 79.

---

## Repositorio CRD

Personalice y someta el trabajo ADNVCRD para asignar e inicializar el conjunto de datos VSAM del repositorio CRD. Consulte la documentación del miembro para obtener instrucciones de personalización.

ADNVCRD se encuentra en FEK.#CUST.JCL, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP (FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15.

Debe crear 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.

### Nota:

- Un repositorio de servidor CRD debe alargarse para habilitar el soporte URIMAP añadido al programa de utilidad administrativo en Developer for System z Versión 7.6.1. Encontrará más detalles en: “Notas de migración del programa de utilidad administrativo” en la página 261.
- 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 for System z.

Los usuarios necesitan acceso de lectura (READ) al repositorio CRD, y los administradores de CICS necesitan acceso de actualización (UPDATE).

## Programa de utilidad administrativo de CICS

Developer for System z suministra el programa de utilidad administrativo que permite a los administradores de CICS proporcionar los valores predeterminados

para las definiciones de recursos CICS. Estos valores predeterminados pueden ser de sólo lectura o pueden ser editables para los desarrolladores de aplicaciones.

El programa de utilidad administrativo se invoca mediante el trabajo de ejemplo ADNJSPAU. La utilización de este programa de utilidad requiere acceso de actualización (UPDATE) al repositorio de CRD.

ADNJSPAU se encuentra en FEK.#CUST.JCL, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP (FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15.

Encontrará más información en la sección Capítulo 15, “consideraciones de CICSTS”, en la página 253.

---

## RESTful versus Servicio Web

CICS Transaction Server proporciona en la versión 4.1 y superior soporte para una interfaz HTTP diseñada utilizando los principios de la transferencia de estado representativo (RESTful). Esta interfaz RESTful es ahora la interfaz CICSTS estratégica para las aplicaciones de cliente. La interfaz de servicio web anterior se ha estabilizado, y las mejoras serán únicamente para la interfaz RESTful.

El Gestor de despliegue de aplicaciones sigue esta sentencia de dirección y necesita el servidor CRD de RESTful para todos los servicios que son nuevos Developer for System versión 7.6 o superior.

Las interfaces RESTful y de servicio Web puede estar activas simultáneamente en una única región CICS, si lo desea. En este caso, habrá dos servidores CRD activos en la región. Ambos servidores compartirán el mismo repositorio CRD. Tenga en cuenta que CICS emitirá algunos avisos sobre definiciones duplicadas cuando se defina la segunda interfaz en la región.

---

## Servidor CRD utilizando la interfaz RESTful

La información de esta sección describe cómo definir el servidor CRD que utiliza la interfaz RESTful para comunicarse con el cliente de Developer for System z.

Las interfaces RESTful y de servicio Web puede estar activas simultáneamente en una única región CICS, si lo desea. En este caso, habrá dos servidores CRD activos en la región. Ambos servidores compartirán el mismo repositorio CRD. Tenga en cuenta que CICS emitirá algunos avisos sobre definiciones duplicadas cuando se defina la segunda interfaz en la región.

## Región de conexión primaria CICS

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

- Coloque los módulos de carga FEK.SFEKLOAD(ADNCRD\*, ADNANAL y ADNREST) 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 ADNCSDRS para que actualice la definición de sistema CICS de la región de conexión primaria CICS. Consulte la documentación del miembro para obtener instrucciones de personalización.

ADNCSDRS se encuentra en FEK.#CUST.JCL, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15.

- Utilice el mandato CEDA adecuado para instalar el grupo de Gestor de despliegue de aplicaciones para esta región; por ejemplo:

CEDA INSTALL GROUP(ADNPCRGP)

## Regiones de conexión no primarias CICS

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

**Nota:** No hace falta realizar estos pasos si se utiliza BAS (Business Application Services) de CICSplex SM para gestionar las definiciones de recursos CICS.

- Coloque el módulo de carga del Gestor de despliegue de aplicaciones module FEK.SFEKLOAD(ADNCRD\*) 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 ADNCSDAR para que actualice la CSD estas regiones de conexión no primarias. Consulte la documentación del miembro para obtener instrucciones de personalización.

ADNCSDAR se encuentra en FEK.#CUST.JCL, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15.

- Utilice el mandato CEDA pertinente para instalar el grupo de Gestor de despliegue de aplicaciones para estas regiones; por ejemplo:

CEDA INSTALL GROUP(ADNARRGP)

## (Opcional) Personalizar ID de transacción del servidor CRD

Developer for System z suministra varias transacciones que el servidor CRD utiliza al definir y consultar recursos CICS.

Tabla 11. ID de transacción predeterminados del servidor CRD

Transacción	Descripción
ADMS	Para las solicitudes de la herramienta de Proceso de manifiestos para cambiar 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.

Puede cambiar los ID de transacción para que coincidan con los estándares del local siguiendo estos pasos:

1. Personalice y someta ADNTXNC para crear el módulo de carga ADNRCUST. Consulte la documentación del miembro para obtener instrucciones de personalización.
2. Coloque el módulo de carga ADNRCUST resultante en la concatenación RPL de CICS (sentencia DD DFHRPL) de las regiones CICS donde está definido el servidor CRD.

3. Personalice y someta ADNCSDTX para definir ADNRCUST como programa en las regiones CICS donde está definido el servidor CRD. Consulte la documentación del miembro para obtener instrucciones de personalización.

**Nota:** El servidor CRD de RESTful siempre intentará cargar el módulo de carga ADNRCUST. De esta manera, puede obtener una pequeña ventaja de rendimiento creando y definiendo el módulo de carga ADNRCUST, incluso aunque no cambie los ID de transacción.

---

## Servidor CRD utilizando la interfaz de servicios Web

La información de esta sección describe cómo definir el servidor CRD que utiliza la interfaz de servicio Web para comunicarse con el cliente de Developer for System z.

Las interfaces RESTful y de servicio Web puede estar activas simultáneamente en una única región CICS, si lo desea. En este caso, habrá dos servidores CRD activos en la región. Ambos servidores compartirán el mismo repositorio CRD. Tenga en cuenta que CICS emitirá algunos avisos sobre definiciones duplicadas cuando se defina la segunda interfaz en la región.

## Manejador de mensajes de conducto

El manejador de mensajes de conducto (ADNTMSGH) se utiliza para la seguridad, procesando el ID de usuario y la contraseña en la cabecera SOAP. El archivo de configuración de conducto de ejemplo hace referencia a ADNTMSGH, que, por lo tanto, se debe colocar en la concatenación RPL de CICS. Consulte la sección Capítulo 15, “consideraciones de CICSTS”, en la página 253 para obtener más información acerca del manejador de mensajes de conducto y la configuración de seguridad necesaria.

Developer for System z suministra varias transacciones que el servidor CRD utiliza al definir y consultar recursos CICS. Estos ID de transacción se establecen mediante ADNTMSGH, dependiendo de la operación solicitada. Se proporciona código fuente COBOL para permitir personalizaciones específicas del sitio para ADNTMSGH:

*Tabla 12. ID de transacción predeterminados del servidor CRD*

Transacción	Descripción
ADMS	Para las solicitudes de la herramienta de Proceso de manifiestos para cambiar recursos CICS. Normalmente, está destinado a los administradores de CICS.
ADMI	Para las peticiones que definen, instalan o desinstalan recursos CICS.
ADMR	Para todas las demás peticiones que recuperan información de recursos o de entorno de CICS.

Utilizando el valor predeterminado:

- Coloque el módulo de carga de FEK.SFEKLOAD(ADNTMSGH) 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 ADNTMSGH:

Los miembros de ejemplo ADNMSGH\* se encuentran en FEK.#CUST.JCL y FEK.#CUST.COBOL a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15.

- Personalice el código fuente del Manejador de mensajes de conducto (COBOL) de ejemplo, FEK.#CUST.COBOL(ADNMSGHS), para que coincida con los estándares de su ubicación.
- Personalice y someta el trabajo FEK.#CUST.JCL(ADNMSGHC) para compilar el fuente de ADNMSGHS personalizado. Consulte la documentación en ADNMSGHC para obtener instrucciones de personalización. Tenga en cuenta que el módulo de carga resultante debe denominarse ADNTMSGH.
- Coloque el módulo de carga ADNTMSGH 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 ADNTMSGH personalizado se coloque antes de hacer ninguna referencia a FEK.SFEKLOAD, pues de lo contrario se utilizaría el valor predeterminado.

## 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 servicio procedentes de Developer for System z.

- Coloque los módulos de carga FEK.SFEKLOAD(ADNCRD\*, ADNANAL y ADNREST) 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. Tenga en cuenta que el módulo de carga del manejador de mensajes de conducto, ADNTMSGH, también debe colocarse en la concatenación RPL, como se describe en la sección “Manejador de mensajes de conducto” en la página 77.
- Personalice y someta el trabajo ADNCSOWS para actualizar la definición del sistema CICS (CSD) de la región de conexión primaria CICS. Consulte la documentación del miembro para obtener instrucciones de personalización. Tenga en cuenta que los ID de transacción utilizados en este trabajo deben coincidir con los utilizados por el manejador de mensajes de conducto (que puede haber sido personalizado).

ADNCSOWS se encuentra en FEK.#CUST.JCL, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15.

- Utilice el mandato CEDA adecuado para instalar el grupo de Gestor de despliegue de aplicaciones para esta región; por ejemplo:  
CEDA INSTALL GROUP(ADNPCRGP)

## Regiones de conexión no primarias CICS

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

**Nota:** No hace falta realizar estos pasos si se utiliza BAS (Business Application Services) de CICSplex SM para gestionar las definiciones de recursos CICS.

- Coloque los módulo de carga del Gestor de despliegue de aplicaciones FEK.SFEKLOAD(ADNCRD\*) en la concatenación RPL de CICS (sentencia DD DFHRPL) de estas regiones de conexión no primarias. Debe hacerlo 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 ADNCSDAR para que actualice la CSD estas regiones de conexión no primarias. Consulte la documentación del miembro para obtener instrucciones de personalización.

ADNCSDAR se encuentra en FEK.#CUST.JCL, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP).

Encontrará más detalles en: “Configuración de la personalización” en la página 15.

- Utilice el mandato CEDA pertinente para instalar el grupo de Gestor de despliegue de aplicaciones para estas regiones; por ejemplo:

CEDA INSTALL GROUP(ADNARRGP)

---

## (Opcional) Repositorio de manifiesto

Developer for System z permite a los clientes examinar y opcionalmente cambiar manifiestos que describen recursos CICS seleccionados. En función de los permisos establecidos por el administrador de CICS, los cambios pueden realizarse directamente o exportarse al repositorio de manifiestos para que el administrador de CICS los procese con mayor detalle.

### Nota:

- Este paso sólo es necesario para los clientes que exportan manifiestos desde Developer for System z para procesarlos con la herramienta de Proceso de manifiestos.
- La herramienta de Proceso de manifiestos es un plug-in de IBM CICS Explorer.

Personalice y someta el trabajo ADNVMFST para asignar e inicializar el conjunto de datos VSAM del repositorio de manifiestos y para definirlo en la región de conexión primaria de CICS. Consulte la documentación del miembro para obtener instrucciones de personalización. Debe crearse un repositorio de manifiesto aparte para cada región de conexión primaria CICS. Todos los usuarios necesitan acceso de actualización (UPDATE) al repositorio de manifiestos.

ADNVMFST se encuentra en FEK.#CUST.JCL, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15.



---

## Capítulo 5. (Opcional) SCLM Developer Toolkit

SCLM Developer Toolkit suministra las herramientas necesarias para ampliar las prestaciones de SCLM para el cliente. SCLM en sí es un gestor de código fuente basado en host-based que se suministra como parte de ISPF.

SCLM Developer Toolkit contiene un plug-in de Eclipse-based 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 el SCLM del sistema central, incluidas las tareas de construir, ensamblar y desplegar el código J2EE desde el sistema central.

---

### Requisitos y lista de comprobación

Necesitará ayuda de un administrador de SCLM y, opcionalmente, de un administrador de seguridad para realizar esta tarea de personalización, que requiere los siguientes recursos y/o tareas de personalización especiales:

- Actualizaciones de APF y LINKLIST
- Definir conversores de lenguaje SCLM para soporte JAVA/J2EE
- Definir tipos SCLM para soporte JAVA/J2EE
- (Opcional) Regla de seguridad para permitir que los usuarios realicen actualizaciones en un VSAM de SCLM
- (Opcional) Instalación de Ant

A fin de empezar a utilizar SCLM Developer Toolkit en su sitio, debe realizar las tareas siguientes. A menos que se indique de otro modo, todas las tareas son obligatorias.

1. Verifique y ajuste los requisitos previos y las actualizaciones de PARMLIB. Para obtener más detalles, consulte “Prerrequisitos” en la página 82.
2. Archivos de configuración para la personalización de Developer for System z. Para obtener más detalles, consulte:
  - “Actualizaciones de ISPF.conf para SCLMDT” en la página 82
  - “Actualizaciones de rsed.envvars para SCLMDT” en la página 83
3. De forma opcional, defina un soporte de conversión de nombres largos/abreviados. Para obtener más detalles, consulte “(Opcional) Conversión de nombres largos/abreviados” en la página 84.
4. De forma opcional, instale y personalice Ant para utilizar el soporte de construcción JAVA/J2EE. Para obtener más detalles, consulte “(Opcional) Instalar y personalizar Ant” en la página 86.
5. Actualice SCLM para definir las partes específicas de SCLMDT. Para obtener más detalles, consulte “Actualizaciones de SCLM para SCLMDT” en la página 87.
6. De forma opcional, configure la automatización para limpiar periódicamente el área de trabajo SCLMDT. Para obtener más detalles, consulte “Eliminar archivos antiguos de WORKAREA” en la página 88.

---

## Prerrequisitos

Consulte Apéndice E, “Requisitos”, en la página 333 para obtener una lista del mantenimiento de SCLM necesario.

Este apéndice también documenta las especificaciones Ant necesarias para construcciones JAVA/J2EE en SCLM Developer Toolkit.

**Atención:** SCLM Developer Toolkit requiere la utilización de la Pasarela de cliente TSO/ISPF de ISPF, lo que implica que sea necesario z/OS 1.8 o posterior.

Como se describe en “Cambios de PARMLIB” en la página 16, SCLM Developer Toolkit requiere una personalización adicional de los valores del sistema. Estos cambios incluyen:

- (BPXPRMxx) Aumente el número máximo de procesos por ID de usuario de z/OS UNIX.
- (PROGxx) Autorización APF SYS1.LINKLIB y el entorno de ejecución REXX, REXX.V1R4M0.SEAGLPA o REXX.V1R4M0.SEAGALT.
- (PROGxx/LPALSTxx) Coloque ISP.SISPLPA, ISP.SISPLOAD, SYS1.LINKLIB y el entorno de ejecución de REXX en LINKLIST/LPALIB.

Asimismo, SCLM Developer Toolkit utiliza SDSF o el mandato **OUTPUT** de TSO para recuperar el estado de finalización de trabajo y la salida del trabajo. Ambos métodos requieren atención adicional:

- SDSF debe solicitarse, instalarse y configurarse por separado. También requiere la utilización de JES2.
- Los valores predeterminados del mandato **OUTPUT** de TSO permiten al usuario recuperar sólo la salida del trabajo que empieza con su ID de usuario. Si desea utilizar plenamente el recurso **OUTPUT**, puede que sea necesario modificar la salida TSO/E de ejemplo IKJEFF53 para que el usuario pueda recuperar la salida de un trabajo del que es propietario, pero que no empieza por su ID de usuario. Para obtener más información acerca de esta salida, consulte la publicación *TSO/E Customization* (SA22-7783).

Los usuarios requieren las autorizaciones de lectura, grabación y ejecución (READ, WRITE y EXECUTE) sobre los directorios de z/OS UNIX /tmp/ y /var/rdz/WORKAREA/. El directorio WORKAREA/ se encuentra en /var/rdz/, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15.

---

## Actualizaciones de ISPF.conf para SCLMDT

SCLM Developer Toolkit utiliza los esqueletos estándar de ISPF/SCLM, por lo que debe asegurarse de que la biblioteca de esqueletos ISP.SISPSLIB se asigne a la concatenación de ISPSLIB en ISPF.conf. La utilización del conjunto de datos ISP.SISPSENU es opcional.

ISPF.conf se encuentra en /etc/rdz/, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15. Puede editar el archivo con el mandato TSO **OEDIT**.

**Nota:** Los cambios están en vigor para todos los clientes que se conecten al host tras la actualización.

El ejemplo de código siguiente muestra el archivo `ISPF.conf`, que debe personalizarse para que coincida con el entorno del sistema. Las líneas de comentarios empiezan por un asterisco (\*). Añada los conjuntos de datos a la concatenación en la misma línea y separe los nombres con una coma (.). Consulte “`ISPF.conf`, archivo de configuración de la Pasarela de cliente TSO/ISPF de ISPF” en la página 47 para obtener más detalles acerca de la personalización de `ISPF.conf`.

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

* OPCIONAL:
*allocjob = FEK.#CUST.CNTL(CRAISPRX)
*ISPF_timeout = 900
```

*Figura 21. Actualizaciones de `ISPF.conf` para `SCLMDT`*

**Nota:**

- Puede añadir sus propias sentencias de tipo DD y concatenaciones de conjuntos de datos para personalizar el entorno TSO, imitando un procedimiento de inicio de sesión de TSO. Encontrará más detalles en: Capítulo 16, “Personalizar el entorno TSO”, en la página 265.
- Si realizar construcciones por lotes, asegúrese de que la versión personalizada del esqueleto `FLMLIBS` está concatenada antes que la biblioteca de esqueletos `ISPF/SCLM`.

```
ispslib=hlq.USERSEKEL,ISP.SISPSLIB
```

---

## Actualizaciones de `rsed.envvars` para `SCLMDT`

`SCLM Developer Toolkit` utiliza algunas directivas establecidas en `rsed.envvars` para localizar conjuntos de datos y directorios.

`rsed.envvars` se encuentra en `/etc/rdz/`, a menos que haya especificado otra ubicación al personalizar y someter el trabajo `FEK.SFEKSAMP (FEKSETUP)`. Encontrará más detalles en: “Configuración de la personalización” en la página 15. Puede editar el archivo con el mandato TSO `OEDIT`.

**Nota:** Es necesario reiniciar la tarea iniciada `RSED` para recoger los cambios que haya realizado.

El ejemplo de código siguiente muestra las directivas de `SCLMDT` del archivo `rsed.envvars`, que debe personalizarse para que coincida con el entorno del sistema. Consulte la sección “`rsed.envvars`, archivo de configuración de `RSE`” en la página 32 para obtener más detalles acerca de la personalización de `rsed.envvars`.

```
_SCLMDT_CONF_HOME=/var/rdz/sclmdt
#STEPLIB=$STEPLIB:FEK.SFEKAUTH:FEK.SFEKLOAD
#_SCLMDT_TRANTABLE=FEK.#CUST.LSTRANS.FILE
#ANT_HOME=/usr/lpp/apache/Ant/apache-ant-1.7.1
_SCLMDT_BASE_HOME=$RSE_HOME
_SCLMDT_WORK_HOME=$CMDSERV_WORK_HOME
CGI_DTWORk=$_SCLMDT_WORK_HOME
```

Figura 22. Actualizaciones de *rsed.envvars* para SCLMDT

---

## (Opcional) Conversión de nombres largos/abreviados

SCLM Developer Toolkit ofrece la posibilidad de almacenar archivos de nombre largo (archivos cuyos nombres superan los 8 caracteres o con mayúsculas/minúsculas mezcladas) en SCLM. Esto se logra mediante la utilización de un archivo VSAM que contiene la correlación del nombre de archivo largo con el nombre de miembro de 8 caracteres utilizado en SCLM.

### Nota:

- En las versiones anteriores a z/OS 1.8, este recurso se suministra por medio de un PTF base de ISPF/SCLM destinado al APAR OA11426.
- Otros productos relacionados con SCLM también utilizan la conversión de nombres largos/abreviados, como por ejemplo IBM SCLM Administrator Toolkit.

## Crear LSTRANS.FILE, el VSAM de conversión de nombres largos/abreviados.

Personalice y someta el miembro de ejemplo FLM02LST de la biblioteca de ejemplo ISPF ISP.SISPSAMP para crear el archivo VSAM de conversión de nombres largos/abreviados. En los pasos de configuración de esta publicación se espera que el VSAM se denomine FEK.#CUST.LSTRANS.FILE, como se muestra en el JCL de configuración del ejemplo siguiente.

```

//FLM02LST JOB <parámetros 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. Sustituya ***** por el volumen que contendrá el VSAM.
/* 3. Cambie todas las referencias de FEK.#CUST.LSTRANS.FILE para que
/* coincidan con el convenio de denominación del VSAM de conversión SCLM.
/*
//CREATE EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
DELETE FEK.#CUST.LSTRANS.FILE
SET MAXCC=0
DEFINE CLUSTER(NAME(FEK.#CUST.LSTRANS.FILE) -
               VOLUMES(*****)) -
               RECORDSIZE(58 2048) -
               SHAREOPTIONS(3 3) -
               CYLINDERS(1 1) -
               KEYS(8 0) -
               INDEXED) -
DATA (NAME(FEK.#CUST.LSTRANS.FILE.DATA)) -
INDEX (NAME(FEK.#CUST.LSTRANS.FILE.INDEX))

/* DEFINIR ÍNDICE ALTERNATIVO CON CLAVES NO EXCLUSIVAS -> ESDS */

DEFINE ALTERNATEINDEX(-
                     NAME(FEK.#CUST.LSTRANS.FILE.AIX) -
                     RELATE(FEK.#CUST.LSTRANS.FILE) -
                     RECORDSIZE(58 2048) -
                     VOLUMES(*****)) -
                     CYLINDERS(1 1) -
                     KEYS(50 8) -
                     UPGRADE -
                     NONUNIQUEKEY) -
DATA (NAME(FEK.#CUST.LSTRANS.FILE.AIX.DATA)) -
INDEX (NAME(FEK.#CUST.LSTRANS.FILE.AIX.INDEX))

/*
/*
//PRIME EXEC PGM=IDCAMS,COND=(0,LT)
//SYSPRINT DD SYSOUT=*
//INITREC DD *
INITREC1
/*
//SYSIN DD *
REPRO INFILE(INITREC) -
      OUTDATASET(FEK.#CUST.LSTRANS.FILE)
IF LASTCC = 4 THEN SET MAXCC=0

BLDINDEX IDS(FEK.#CUST.LSTRANS.FILE) -
          ODS(FEK.#CUST.LSTRANS.FILE.AIX)

IF LASTCC = 0 THEN -
  DEFINE PATH (NAME(FEK.#CUST.LSTRANS.FILE.PATH) -
              PATHENTRY (FEK.#CUST.LSTRANS.FILE.AIX))
/*

```

Figura 23. FLM02LST - JCL de configuración de la conversión de nombres largos/abreviados

**Nota:** Los usuarios necesitan la autorización de actualización (UPDATE) sobre este conjunto de datos VSAM, como se describe en Capítulo 10, “Consideraciones relativas a la seguridad”, en la página 159.

## Actualizaciones de rsed.envvars para la conversión de nombres largos/abreviados

Antes de utilizar la conversión de nombres largos/abreviados, descomente y establezca la variable de entorno de rsed.envvars `_SCLMDT_TRANTABLE` para que coincida con el nombre del VSAM de conversión de nombres largos/abreviados.

rsed.envvars se encuentra en `/etc/rdz/`, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15. Puede editar el archivo con el mandato TSO **OEDIT**.

**Nota:** Es necesario reiniciar la tarea iniciada RSED para recoger los cambios que haya realizado.

---

### (Opcional) Instalar y personalizar Ant

Este paso sólo es necesario si tiene previsto utilizar el soporte de construcción JAVA/J2EE en SCLM.

Apache Ant es una herramienta de construcción Java Open Source que puede descargarse desde <http://ant.apache.org/>. Ant consta de archivos de texto distribuidos en formato ASCII y, por tanto, requieren una conversión ASCII/EBCDIC para poder ejecutarse en z/OS UNIX.

Para implementar Ant en z/OS y definirlo en Developer for System z, siga estos pasos:

- Descargue en formato binario el archivo comprimido Ant más reciente en el sistema de archivos de z/OS UNIX. Es aconsejable descargar la versión .zip de ANT debido a los problemas que pueden surgir en z/OS al extraer versiones con el formato de sufijo tar.gz o tar.bz2.
- Abra una sesión de línea de mandatos z/OS UNIX para continuar con la instalación, por ejemplo con el mandato **TSO OMVS**.
- Cree un directorio inicial para la instalación de Ant con el mandato **mkdir -p /home-dir** y conviértalo en su directorio actual con el mandato **cd /home-dir**.
- Utilice el mandato de extracción de JAR **jar -xf apache-ant-1.7.1.zip** para extraer el archivo en el directorio actual. Para poder utilizar el mandato **jar**, debe haber un directorio bin Java en la vía de acceso de z/OS UNIX. De lo contrario, deberá calificar totalmente el mandato con la ubicación de bin Java (por ejemplo, **/usr/lpp/java/J5.0/bin/jar -xf apache-ant-1.7.1.zip**).
- Convierta todos los archivos de texto Ant a EBCDIC ejecutando (y opcionalmente personalizando) el script de ejemplo `/usr/lpp/rdz/samples/BWBTRANT`.

**Nota:** Ejecute este script solamente una vez. Ejecutarlo varias veces dañará su instalación de Ant.

- Para comprobar que la conversión ha sido satisfactoria, localice y examine un archivo de texto del directorio ANT, como por ejemplo `apache-ant-1.7.1/README`. Si el archivo es legible, la conversión ha sido satisfactoria.
- Utilice el mandato **chmod -R 755 \*** para permitir que todos los usuarios lean y ejecuten (autorizaciones READ y EXECUTE) archivos del directorio ANT.
- Antes de utilizar Ant, establezca las variables de entorno de rsed.envvars `JAVA_HOME` y `ANT_HOME`.

- JAVA\_HOME es necesaria para señalar al directorio inicial de Java, por ejemplo:  
JAVA\_HOME=/usr/lpp/java/IBM/J5.0
- ANT\_HOME es necesaria para señalar al directorio inicial de Ant, por ejemplo:  
ANT\_HOME=/usr/lpp/Apache/Ant/apache-ant-1.7.1

Por ejemplo:

- TSO OMVS
- mkdir -p /usr/lpp/Apache/Ant
- cd /usr/lpp/Apache/Ant
- jar -xf /u/userid/apache-ant-1.7.1
- /usr/lpp/rdz/samples/BWBTRANT
- cat ./apache-ant-1.7.1/README
- chmod -R 755 \*
- oedit /etc/rsed.envvars

Para comprobar que la inicialización de Ant ha sido satisfactoria:

- Añada los directorios bin de Ant y Java a la variable de entorno PATH.

Ejemplo:

```
export PATH=/usr/lpp/Apache/Ant/apache-ant-1.7.1/bin:$PATH
export PATH=/usr/lpp/java/IBM/J5.0/bin:$PATH
```

- Ejecute **ant -version** para visualizar la versión, si se ha instalado correctamente.

Ejemplo:

```
ant -version
```

**Nota:** Establecer de este modo la sentencia PATH sólo es necesario a efectos de prueba, no a efectos operativos.

## Actualizaciones de SCLM para SCLMDT

El propio SCLM también requiere personalización para funcionar con SCLM Developer Toolkit. Consulte la *Guía del administrador de IBM Rational Developer for System z SCLM Developer Toolkit*, SC11-3815-00 (SC23-9801), para obtener más información acerca de las tareas de personalización necesarias:

- Definir conversores de lenguaje para soporte JAVA/J2EE
- Definir tipos SCLM para soporte JAVA/J2EE

Para realizar las tareas de personalización y definición de proyectos, el administrador de SCLM necesita conocer varios valores personalizables de Developer for System z, como se describe en la Tabla 13.

Tabla 13. Lista de comprobación del administrador de SCLM

Descripción	<ul style="list-style-type: none"> <li>• Valor predeterminado</li> <li>• Dónde encontrar la respuesta</li> </ul>	Valor
Biblioteca de ejemplos de Developer for System z	<ul style="list-style-type: none"> <li>• FEK.SFEKSAMV</li> <li>• Instalación de SMP/E</li> </ul>	

Tabla 13. Lista de comprobación del administrador de SCLM (continuación)

Descripción	<ul style="list-style-type: none"> <li>• Valor predeterminado</li> <li>• Dónde encontrar la respuesta</li> </ul>	Valor
Directorio de ejemplos de Developer for System z	<ul style="list-style-type: none"> <li>• /usr/lpp/rdz/samples</li> <li>• Instalación de SMP/E</li> </ul>	
Directorio bin Java	<ul style="list-style-type: none"> <li>• /usr/lpp/java/J5.0/bin</li> <li>• rsed.envvars - \$JAVA_HOME/bin</li> </ul>	
Directorio bin Ant	<ul style="list-style-type: none"> <li>• /usr/lpp/Apache/Ant/apache-ant-1.7.1/bin</li> <li>• rsed.envvars - \$ANT_HOME/bin</li> </ul>	
Directorio inicial WORKAREA	<ul style="list-style-type: none"> <li>• /var/rdz</li> <li>• rsed.envvars - \$_CMDSESV_CONF_HOME</li> </ul>	
Directorio inicial de configuración de proyectos SCLMDT	<ul style="list-style-type: none"> <li>• /var/rdz/sclmdt</li> <li>• rsed.envvars - \$_SCLMDT_CONF_HOME</li> </ul>	
VSAM de conversión de nombres largos/abreviados	<ul style="list-style-type: none"> <li>• FEK.#CUST.LSTRANS.FILE</li> <li>• rsed.envvars - \$_SCLMDT_TRANTABLE</li> </ul>	

## Eliminar archivos antiguos de WORKAREA

SCLM Developer Toolkit y la Pasarela de cliente TSO/ISPF de ISPF comparten la misma WORKAREA, que puede necesitar operaciones de limpieza periódicas. Consulte la sección “(Opcional) Limpieza de WORKAREA” en la página 108 para obtener más información.

---

## Capítulo 6. (Opcional) Otras tareas de personalización

Esta sección combina diversas tareas de personalización opcionales. Siga las instrucciones de la sección adecuada para configurar el servicio que desee.

- “(Opcional) Procedimiento almacenado DB2”
- “(Opcional) Soporte de Enterprise Service Tools (EST)” en la página 91
- “(Opcional) Soporte de idiomas bidireccionales CICS” en la página 92
- “(Opcional) Diagnóstico de mensajes de error de IRZ” en la página 93
- “(Opcional) Cifrado SSL de RSE” en la página 93
- “(Opcional) Rastreo de RSE” en la página 96
- “(Opcional) Grupos de propiedades basadas en host” en la página 98
- “(Opcional) Proyectos basados en host” en la página 99
- “(Opcional) Integración del gestor de archivos” en la página 100
- “(Opcional) Caracteres no editables” en la página 101
- “(Opcional) Utilizar REXEC (o SSH)” en la página 102
- “(Opcional) Transacción APPC para el servicio de mandatos TSO” en la página 104
- “(Opcional) Limpieza de WORKAREA” en la página 108

---

### (Opcional) Procedimiento almacenado DB2

Necesitará ayuda de un administrador de WLM y de un administrador de DB2 para realizar esta tarea de personalización, que requiere los siguientes recursos o tareas de personalización especiales:

- Actualización de WLM
- Nuevo miembro PROCLIB
- Actualización de DB2

---

Developer para System z suministra un procedimiento almacenado DB2 de ejemplo (Constructor de procedimientos almacenados PL/I y COBOL) para construir procedimientos almacenados COBOL y PL/I desde el cliente Developer para System z.

**Nota:** Los miembros de ejemplo ELAXM\* se encuentran en FEK.#CUST.JCL y FEK.#CUST.PROCLIB a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15.

### Cambios del gestor de carga de trabajo (WLM)

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. Consulte la publicación *MVS Planning Workload Management* (SA22-7602) para obtener información acerca de esta operación.

**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.

## Cambios de PROCLIB

Personalice la de procedimiento almacenado de ejemplo FEK.#CUST.PROCLIB(ELAXMSAM), tal como se describe dentro del miembro, y cópielo en SYS1.PROCLIB. Como se muestra en el ejemplo de código que figura a continuación, debe suministrar lo siguiente:

- El nombre del entorno de aplicación definido en WLM para este procedimiento almacenado
- El nombre del subsistema DB2
- El calificador de alto nivel de diversos conjuntos de datos

```
//ELAXMSAM PROC RGN=0M,  
//          NUMTCB=1,  
//          APPLENV=#w1mwd4z,  
//          DB2SSN=#ssn,  
//          DB2PRFX='DSN810',  
//          COBPRFX='IGY.V3R4M0',  
//          PLIPRFX='IBMZ.V3R6M0',  
//          LIBPRFX='CEE',  
//          LODPRFX='FEK'  
//*  
//DSNX9WLM EXEC PGM=DSNX9WLM,REGION=&RGN,TIME=NOLIMIT,DYNAMNBR=10,  
//          PARM='&DB2SSN,&NUMTCB,&APPLENV'  
//STEPLIB DD DISP=SHR,DSN=&DB2PRFX..SDSNEXIT  
//          DD DISP=SHR,DSN=&DB2PRFX..SDSNLOAD  
//          DD DISP=SHR,DSN=&LIBPRFX..SCEERUN  
//          DD DISP=SHR,DSN=&COBPRFX..SIGYCOMP  
//          DD DISP=SHR,DSN=&PLIPRFX..SIBMZCMP  
//SYSEXEC DD DISP=SHR,DSN=&LODPRFX..SFEKPROC  
//SYSTSPRT DD SYSOUT=*  
//CEEDUMP DD SYSOUT=*  
//SYSABEND DD DUMMY  
//SYSUT1 DD UNIT=SYSALLDA,SPACE=(CYL,(1,1))  
//SYSUT2 DD UNIT=SYSALLDA,SPACE=(CYL,(1,1))  
//SYSUT3 DD UNIT=SYSALLDA,SPACE=(CYL,(1,1))  
//SYSUT4 DD UNIT=SYSALLDA,SPACE=(CYL,(1,1))  
//SYSUT5 DD UNIT=SYSALLDA,SPACE=(CYL,(1,1))  
//SYSUT6 DD UNIT=SYSALLDA,SPACE=(CYL,(1,1))  
//SYSUT7 DD UNIT=SYSALLDA,SPACE=(CYL,(1,1))  
//*
```

Figura 24. ELAXMSAM - Tarea de procedimiento almacenado DB2

### Nota:

- El procedimiento almacenado DB2 utiliza el exec de REXX ELAXMREX, ubicado en FEK.SFEKPROC. No cambie esta ubicación si desea que el posible mantenimiento SMP/E se active automáticamente.
- Si desea cambiar el nombre de los miembros ELAXMSAM o ELAXMREX, vea: Capítulo 17, “Ejecutar varias instancias”, en la página 273.

## Cambios de DB2

Personalice y someta el miembro de ejemplo ELAXMJCL del conjunto de datos FEK.#CUST.JCL para definir el procedimiento almacenado en DB2. Consulte la documentación del miembro para obtener instrucciones de personalización.

```

//ELAXMJCL JOB <job parameters>
//JOBPROC JCLLIB ORDER=(#hlq.SDSNPROC)
//JOBLIB DD DISP=SHR,DSN=#hlq.SDSNEXIT
// DD DISP=SHR,DSN=#hlq.SDSNLOAD
//*
//RUNTIAD EXEC PGM=IKJEFT01,DYNAMNBR=20
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
DSN S(#ssn) R(1) T(1)
RUN PROGRAM(DSNTIAD) PLAN(DSNTIAD) -
LIB('#hlq.RUNLIB.LOAD')
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
CREATE PROCEDURE SYSPROC.ELAXMREX
( IN FUNCTION_REQUEST VARCHAR(20) CCSID EBCDIC
, IN SQL_ROUTINE_NAME VARCHAR(27) CCSID EBCDIC
, IN SQL_ROUTINE_SOURCE VARCHAR(32672) CCSID EBCDIC
, IN BIND_OPTIONS VARCHAR(1024) CCSID EBCDIC
, IN COMPILE_OPTIONS VARCHAR(255) CCSID EBCDIC
, IN PRECOMPILE_OPTIONS VARCHAR(255) CCSID EBCDIC
, IN PRELINK_OPTIONS VARCHAR(32672) CCSID EBCDIC
, IN LINK_OPTIONS VARCHAR(255) CCSID EBCDIC
, IN ALTER_STATEMENT VARCHAR(32672) CCSID EBCDIC
, IN SOURCE_DATASETNAME VARCHAR(80) CCSID EBCDIC
, IN BUILDOWNER VARCHAR(8) CCSID EBCDIC
, IN BUILDUTILITY VARCHAR(18) CCSID EBCDIC
, OUT RETURN_VALUE VARCHAR(255) CCSID EBCDIC )
PARAMETER STYLE GENERAL RESULT SETS 1
LANGUAGE REXX EXTERNAL NAME ELAXMREX
COLLID DSNREXCS WLM ENVIRONMENT ELAXMSAM
PROGRAM TYPE MAIN MODIFIES SQL DATA
STAY RESIDENT NO COMMIT ON RETURN NO
ASUTIME NO LIMIT SECURITY USER;

COMMENT ON PROCEDURE SYSPROC.ELAXMREX IS
'PLI & COBOL PROCEDURE PROCESSOR (ELAXMREX), INTERFACE LEVEL 0.01';

GRANT EXECUTE ON PROCEDURE SYSPROC.ELAXMREX TO PUBLIC;
//*
```

*Figura 25. ELAXMJCL – Definición de procedimiento almacenado DB2*

**Nota:** 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) Soporte de Enterprise Service Tools (EST)

Esta tarea de personalización no requiere asistencia, recursos especiales ni tareas de personalización especiales.

---

El cliente de Developer for System z tiene un componente de generación de código denominado Enterprise Service Tools (EST). Dependiendo del tipo de código que se está generando, este código se basa en las funciones proporcionadas por la instalación de host de Developer for System z. Cómo hacer que estas funciones de host estén disponibles se describe en las secciones siguientes:

- Capítulo 4, “(Opcional) Gestor de despliegue de aplicaciones (ADM)”, en la página 73
- “(Opcional) Soporte de idiomas bidireccionales CICS” en la página 92
- “(Opcional) Diagnóstico de mensajes de error de IRZ” en la página 93

**Nota:** Enterprise Service Tools (EST) abarca varias herramientas, como SFM (Modelador de flujo de servicios) y XSE (Servicios XML para la empresa).

---

## (Opcional) Soporte de idiomas bidireccionales CICS

Necesitará ayuda de un administrador de CICS para realizar esta tarea de personalización, que requiere los siguientes recursos o tareas de personalización especiales:

- Actualizar JCL de región CICS
  - Definir un programa en CICS
- 

El componente Herramientas de servicio de empresa (EST) de Developer for 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.

Además, el código generado por EST puede soportar la transformación bidi en entornos distintos de SFR de CICS (Tiempo de ejecución de flujo de servicios). Un ejemplo son las 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, FEK.SFEKLOAD.

Realice las tareas siguientes para activar el soporte de idiomas bidireccionales CICS:

1. Coloque los módulos de carga de FEK.SFEKLOAD FEJBDCMP y FEJBDTRX en la concatenación RPL de CICS (sentencia DD DFHRPL). Debe hacerlo añadiendo el conjunto de datos de instalación a la concatenación para que el mantenimiento aplicado esté automáticamente disponible para CICS.

**Nota:** Si no concatena el conjunto de datos de instalación, sino que copia los módulos en un conjunto de datos nuevo o existente, no olvide que estos módulos son DLL y DEBEN residir en una biblioteca PDSE.

2. Defina FEJBDCMP y FEJBDTRX como programas para CICS mediante el mandato CEDA adecuado, por ejemplo:

```
CEDA DEF PROG(FEJBDCMP) LANG(LE) G(xxx)
CEDA DEF PROG(FEJBDTRX) LANG(LE) G(xxx)
```

---

## (Opcional) Diagnóstico de mensajes de error de IRZ

Esta tarea de personalización no requiere asistencia, pero sí requiere los siguientes recursos o tareas de personalización especiales:

- Actualización de LINKLIST
  - Actualizar JCL de región CICS
- 

El cliente de Developer for System z tiene un componente de generación de código denominado Enterprise Service Tools (EST). Para que el código generado por EST emita mensajes de error de diagnóstico, todos los módulos IRZ\* y IIRZ\* de la biblioteca de carga FEK.SFEKLOAD deben estar disponibles para el código generado. EST puede generar un código para los siguientes entornos:

- CICS
- IMS
- Proceso por lotes MVS

Cuando el código generado se ejecuta en una transacción CICS, añada todos los módulos IRZ\* y IIRZ\* de FEK.SFEKLOAD al DD DFHRPL de la región CICS. Debe hacerlo añadiendo el conjunto de datos de instalación a la concatenación para que el mantenimiento aplicado esté automáticamente disponible para CICS.

En el resto de casos, haga que todos los módulos IRZ\* y IIRZ\* de FEK.SFEKLOAD estén disponibles, ya sea través de STEPLIB o de LINKLIST. Debe hacerlo añadiendo el conjunto de datos de instalación a la concatenación para que el mantenimiento aplicado esté automáticamente disponible para CICS.

Si decide utilizar STEPLIB, debe definir los módulos no disponibles por medio de LINKLIST en la directiva STEPLIB de la tarea que ejecuta el código.

Si los módulos de carga no están disponibles y el código generado encuentra un error, se emitirá este mensaje:

IRZ9999S No se ha podido recuperar el texto del mensaje de tiempo de ejecución de Language Environment. Compruebe que el módulo del mensaje de tiempo de ejecución de Language Environment para el recurso

---

## (Opcional) Cifrado SSL de RSE

Necesitará ayuda de un administrador de seguridad para realizar esta tarea de personalización, que requiere los siguientes recursos o tareas de personalización especiales:

- Actualización de LINKLIST
  - Regla de seguridad para añadir conjuntos de datos controlados por programa
  - (Opcional) Regla de seguridad para añadir el certificado para SSL
- 

La comunicación externa (cliente-host) puede cifrarse mediante SSL (Capa de sockets seguros). Esta característica está inhabilita por omisión y está controlada por los valores de `ssl.properties`.

`ssl.properties` se encuentra en `/etc/rdz/`, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15. Puede editar el archivo con el mandato TSO **OEDIT**. Tenga en cuenta que RSE debe reiniciarse para que los cambios entren en vigor.

El cliente se comunica con el daemon RSE durante la configuración de la conexión y con el servidor RSE durante la propia sesión. Las dos secuencias de datos están cifradas cuando la SSL está habilitada.

El daemon RSE y el servidor RSE soportan distintos mecanismos para almacenar certificados debido a las diferencias de su arquitectura. Esto hace que las definiciones SSL sean necesarias tanto para el daemon RSE como para el servidor RSE. Se puede utilizar un certificado compartido si el daemon RSE y el servidor RSE utilizan el mismo método de gestión de certificados.

*Tabla 14. Mecanismos de almacenamiento de certificados de SSL*

Almacenamiento de certificados	Creado y gestionado por	daemon RSE	servidor RSE
anillo de claves	producto de seguridad compatible con SAF	soportado	soportado
base de datos de claves	gskkyman de z/OS UNIX	soportado	/
almacén de claves	Keytool de Java	/	soportado

**Nota:**

- Los anillos de claves compatibles con SAF son el método preferido para gestionar certificados.
- Los anillos de claves compatibles con SAF puede almacenar la clave privada del certificado en la base de datos de seguridad o mediante el ICSF, la interfaz al hardware criptográfico de System z. El acceso a ICSF está protegido con perfiles en la clase de seguridad CSFSERV.

El daemon RSE utiliza funciones de SSL del sistema para gestionar la SSL. Ello implica que SYS1.SIEALNKE debe estar controlado por programa por su software de seguridad y disponible para RSE a través de la directiva LINKLIST o STEPLIB en rsed.envvars.

El ejemplo de código siguiente muestra el archivo ssl.properties de ejemplo, que debe personalizarse para que coincida con el entorno del sistema. Las líneas de comentarios empiezan por el signo de almohadilla (#) cuando se utiliza una página de códigos de EE.UU. Las líneas de datos solamente pueden tener una directiva y su valor asignado; los comentarios no pueden estar en la misma línea. Las continuaciones de línea no están soportadas.

```
# ssl.properties – Archivo de
configuración SSL
enable_ssl=false

# Propiedades del daemon

#daemon_keydb_file=
#daemon_keydb_password=
#daemon_key_label=

# Propiedades del servidor

#server_keystore_file=
#server_keystore_password=
#server_keystore_label=
#server_keystore_type=JCERACFKS
```

*Figura 26. 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 A, “Configurar SSL y autenticación de X.509”, en la página 295.

#### **enable\_ssl**

Habilitar o inhabilitar la comunicación SSL. El valor predeterminado es false. Las únicas opciones válidas son true y false.

#### **daemon\_keydb\_file**

Nombre de anillo de claves de RACF (o de un producto de seguridad similar). Especifique el nombre de la base de datos de claves si ha utilizado **gskkyman** para crear una base de datos de claves en lugar de utilizar un anillo de claves. Descomente y personalice esta directiva si SSL está habilitado.

#### **daemon\_keydb\_password**

Déjelo sin comentarios o en blanco si utiliza un anillo de claves; de lo contrario, especifique la contraseña de la base de datos de claves. Descomente y personalice esta directiva si SSL está habilitado y está utilizando una base de datos de claves **gskkyman**.

#### **daemon\_key\_label**

Etiqueta de certificado utilizada en el anillo de claves o en la base de datos de claves, si no se ha definido como valor predeterminado. Si se utiliza el valor predeterminado, hay que quitar el carácter de comentario. Descomente y personalice esta directiva si SSL está habilitado y no utiliza el certificado de seguridad predeterminado.

#### **server\_keystore\_file**

Nombre del almacén de claves creado por el mandato **keytool** de Java o el nombre del anillo de claves de RACF (o producto de seguridad similar) si **server\_keystore\_type=JCERACFKS**. Descomente y personalice esta directiva si SSL está habilitado.

#### **server\_keystore\_password**

Déjelo sin comentarios o en blanco si utiliza un anillo de claves; de lo contrario, especifique la contraseña del almacén de claves. Descomente y personalice esta directiva si SSL está habilitado y está utilizando un almacén de claves **keytool**.

#### **server\_keystore\_label**

Etiqueta de certificado utilizada en el anillo de claves o en el almacén de claves. El valor predeterminado es el primer certificado válido que se

encuentre. Descomente y personalice esta directiva si SSL está habilitado y no utiliza el certificado de seguridad predeterminado.

**server\_keystore\_type**

Tipo de almacén de claves. El valor predeterminado es JKS. Los valores válidos son:

*Tabla 15. Tipos de almacenes de claves válidos*

Palabra clave	Tipo de almacén de claves
JKS	Almacén de claves Java
JCERACFKS	Anillo de claves compatible con SAF, donde la clave privada del certificado se almacena en la base de datos de seguridad.
JCECCARACFKS	Anillo de claves compatible con SAF, donde la clave privada del certificado se almacena mediante el ICSF, la interfaz al hardware criptográfico de System z.

**Nota:** En el momento de la publicación, IBM z/OS Java requiere una actualización del archivo `/usr/lpp/java/J5.0/lib/security/java.security` para soportar JCECCARACFKS. Es necesario añadir esta línea:

```
security.provider.1=com.ibm.crypto.hwcca.provider.IBMJCECCA
```

El archivo resultante será como este:

```
security.provider.1=com.ibm.crypto.hwcca.provider.IBMJCECCA
security.provider.2=com.ibm.jsse2.IBMJSSEProvider2
security.provider.3=com.ibm.crypto.provider.IBMJCE
security.provider.4=com.ibm.security.jgss.IBMJGSSProvider
security.provider.5=com.ibm.security.cert.IBMCertPath
security.provider.6=com.ibm.security.sasl.IBMSASL
```

---

## (Opcional) Rastreo de RSE

Esta tarea de personalización no requiere asistencia, recursos especiales ni tareas de personalización especiales.

---

Developer for System z da soporte a diversos niveles de rastreo del flujo de programa interno a efectos de resolución de problemas. RSE y algunos de los servicios a los que llama RSE utilizan los valores de `rsecomm.properties` para conocer el nivel de detalle deseado en las anotaciones de salida.

**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.

`rsecomm.properties` se encuentra en `/etc/rdz/`, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15. Puede editar el archivo con el mandato TSO **OEDIT**.

El ejemplo de código que sigue muestra el archivo `rsecomm.properties`, que puede personalizar para que coincida con sus necesidades de rastreo. Las líneas de comentarios empiezan por el signo de almohadilla (#) cuando se utiliza una página de códigos de EE.UU. Las líneas de datos solamente pueden tener una directiva y su valor asignado; los comentarios no pueden estar en la misma línea. Las

continuaciones de línea no están soportadas.

```
# 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  
debug_level=1  
  
# Ubicación de las anotaciones  
# Log_To_StdOut  
# Log_To_File  
log_location=Log_To_File
```

Figura 27. *rsecomm.properties* – archivo de configuración de anotaciones

#### **server.version**

Versión del servidor de anotaciones. El valor predeterminado es 5.0.0. No lo modifique.

#### **debug\_level**

Nivel de detalle de las anotaciones de salida. El valor predeterminado es 1 (anotar mensajes de error y aviso). Tenga en cuenta que `debug_level` controla el nivel de detalle de varios servicios (y por tanto varios archivos de salida). El aumento del nivel de detalle afectará negativamente al rendimiento y solo debe realizarse bajo indicación del centro de soporte de IBM. Consulte la sección “Rastreo de RSE” en la página 146 para obtener más información acerca de las anotaciones controladas por esta directiva.

Los valores válidos son los siguientes:

0	Anotar sólo mensajes de error.
1	Anotar mensajes de error y aviso.
2	Anotar mensajes de error, aviso e informativos.

**Nota:** `debug_level` se puede cambiar dinámicamente con el mandato de operador **modify rsecommlog**, tal como se describe en Capítulo 8, “Mandatos de operador”, en la página 125.

#### **log\_location**

Soporte de salida para las anotaciones relacionadas con RSE. El valor predeterminado es `Log_To_File`. No lo cambie si está utilizando el método de conexión de daemon RSE (valor predeterminado), a menos que así se lo indique el centro de soporte de IBM.

Los valores válidos son los siguientes:

Log_To_File	Enviar los mensajes de anotaciones a un archivo separado del directorio de salida de anotaciones. <ul style="list-style-type: none"><li>• Daemon RSE: <code>rsedaemon.log</code> en <code>daemonlog</code></li><li>• Agrupaciones de hebras RSE: <code>rseserver.log</code> en <code>daemonlog</code></li><li>• Usuario: <code>rsecomm.log</code> in <code>userlog/.eclipse/RSE/\$LOGNAME</code></li></ul>
-------------	--

Log_To_StdOut	<p>Enviar los mensajes de anotaciones a stdout.</p> <ul style="list-style-type: none"> <li>• Daemon RSE: redireccionado a STDOUT de DD en la tarea iniciada RSED</li> <li>• Agrupaciones de hebras RSE: no definido</li> <li>• Usuario: redireccionado a stdout.log en userlog/.eclipse/RSE/\$LOGNAME</li> </ul>
---------------	--

daemonlog es el valor de la directiva daemon.log en rsed.envvars. Si la directiva daemon.log no tiene caracteres de comentario o no está presente, se utiliza la vía de acceso inicial del ID de usuario asignado a la tarea iniciada RSED. La vía de acceso inicial se define en el segmento de seguridad OMVS del ID de usuario.

Las anotaciones específicas del usuario van a userlog/.eclipse/RSE/\$LOGNAME, donde userlog es el valor de la directiva user.log de rsed.envvars, y \$LOGNAME es el ID de usuario de inicio de sesión (en mayúsculas). Si la directiva user.log no tiene caracteres de comentario o no está presente, se utiliza la vía de acceso inicial del usuario. La vía de acceso inicial se define en el segmento de seguridad OMVS del ID de usuario.

---

## (Opcional) Grupos de propiedades basadas en host

Esta tarea de personalización no requiere asistencia, recursos especiales ni tareas de personalización especiales.

---

Los clientes de Developer for System z pueden definir grupos de propiedades que contienen valores predeterminados para diversas propiedades (por ejemplo, las opciones de compilador COBOL que deben utilizarse al compilar código fuente COBOL). Developer for System z tiene algunos valores predeterminados incorporados, pero también permite definir valores predeterminados personalizados específicos del sistema.

La ubicación de los archivos de configuración de valores predeterminados y grupos de propiedades personalizadas se define en propertiescfg.properties, que se encuentra en /etc/rdz/, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP) . Encontrará más detalles en: “Configuración de la personalización” en la página 15. Puede editar el archivo con el mandato TSO **OEDIT**. Tenga en cuenta que RSE debe reiniciarse para que los cambios entren en vigor.

El ejemplo de código siguiente muestra el archivo propertiescfg.properties, que debe personalizarse para que coincida con el entorno del sistema. Las líneas de comentarios empiezan por el signo de almohadilla (#) cuando se utiliza una página de códigos de EE.UU. Las líneas de datos solamente pueden tener una directiva y su valor asignado. Los comentarios no pueden estar en la misma línea. Las continuaciones de línea no están soportadas.

```
#
# grupos de propiedades basadas en host - archivo de configuración raíz
#
ENABLED=FALSE
RDZ-VERSION=7.5.0.0
PROPERTY-GROUP=/var/rdz/properties
DEFAULT-VALUES=/var/rdz/properties
```

*Figura 28. propertiescfg.properties - archivo de configuración de grupos de propiedades basadas en host*

#### **ENABLED**

Indica si Developer for System z utilizará los archivos de configuración de valores predeterminados y grupos de propiedades. El valor predeterminado es FALSE. Las únicas opciones válidas son TRUE y FALSE.

#### **RDZ-VERSION**

Nivel mínimo de cliente Developer for System z para utilizar grupos de propiedades basadas en host. El valor predeterminado es 7.5.0.0. No lo modifique.

#### **PROPERTY-GROUP**

La ubicación del archivo de configuración de grupos de propiedades. El valor predeterminado es /var/rdz/properties.

#### **DEFAULT-VALUES**

La ubicación del archivo de configuración de valores predeterminados. El valor predeterminado es /var/rdz/properties.

Consulte Developer for System z Information Center (<http://publib.boulder.ibm.com/infocenter/ratdevz/v7r6/index.jsp>) para obtener más información acerca de la creación del archivo de configuración de grupos de propiedades (propertygroups.xml) y del archivo de configuración de valores predeterminados (defaultvalues.xml).

---

## **(Opcional) Proyectos basados en host**

Esta tarea de personalización no requiere asistencia, recursos especiales ni tareas de personalización especiales.

---

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 projectcfg.properties, que se encuentra en /etc/rdz/, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP) . Encontrará más detalles en: "Configuración de la personalización" en la página 15. Puede editar el archivo con el mandato TSO **OEDIT**. Tenga en cuenta que RSE debe reiniciarse para que los cambios entren en vigor.

El ejemplo de código siguiente muestra el archivo projectcfg.properties, que debe personalizarse para que coincida con el entorno del sistema. Las líneas de comentarios empiezan por el signo de almohadilla (#) cuando se utiliza una página de códigos de EE.UU. Las líneas de datos solamente pueden tener una directiva y su valor asignado. Los comentarios no pueden estar en la misma línea. Las

continuaciones de línea no están soportadas.

```
#
# proyectos basados en host – archivo de configuración raíz
#
# WSED-VERSION – no lo modifique!
WSED-VERSION=7.0.0.0
# Especifique la ubicación de los archivos de definición de los proyectos basados en host
PROJECT-HOME=/var/rdz/projects
```

Figura 29. *projectcfg.properties* – archivo de configuración de proyectos basados en host

#### WSED-VERSION

Nivel mínimo de cliente Developer for System z para utilizar proyectos basados en host. El valor predeterminado es 7.0.0.0. No lo modifique.

#### PROJECT-HOME

Directorio base de las definiciones de proyecto. El valor predeterminado es /var/rdz/projects.

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

Consulte Developer for System z Information Center (<http://publib.boulder.ibm.com/infocenter/ratdevz/v7r6/index.jsp>) para obtener más información sobre los proyectos basados en host.

---

## (Opcional) Integración del gestor de archivos

Necesitará ayuda de un administrador de seguridad para realizar esta tarea de personalización, que requiere los siguientes recursos o tareas de personalización especiales:

- Regla de seguridad para añadir conjuntos de datos controlados por programa

Developer for 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 los programas de utilidad habituales para examinar, editar, copiar e imprimir existentes en ISPF, mejoradas para responder a las necesidades de los desarrolladores de aplicaciones. En la versión actual de Developer for System z, sólo están permitidos los programas de utilidad para examinar y editar conjuntos de datos MVS (incluidos todos los tipo de VSAM), crear y editar plantillas de conjuntos de datos MVS (incluidas las plantillas dinámicas), y copiar con opciones avanzadas.

Tenga en cuenta que el producto IBM File Manager para z/OS se debe pedir, instalar y configurar por separado. Consulte *Rational Developer for System z Prerrequisitos* (SC23-7659) para saber qué nivel del gestor de archivos se necesita para su versión de Developer for System z. La instalación y la personalización de este producto no se describe en este manual.

Tenga en cuenta que ni Developer for System z ni el Gestor de archivos soportan la interfaz por lotes para acceder a los servicios del gestor de archivos. Ahora es necesario utilizar la escucha del Gestor de archivos.

**Nota:** Además de las tareas de configuración del escucha habituales descritas en la documentación del gestor de archivos, Developer for System z requiere que los conjuntos de datos STEPLIB del servidor estén controlados por programa.

Las definiciones de la integración del gestor de archivos que necesita Developer for System z se almacenan en `FMIEXT.properties`, que se encuentra en `/etc/rdz/`, a menos que haya especificado otra ubicación al personalizar y someter el trabajo `FEK.SFEKSAMP(FEKSETUP)`. Encontrará más detalles en: “Configuración de la personalización” en la página 15. Puede editar el archivo con el mandato `TSO OEDIT`. Tenga en cuenta que RSE debe reiniciarse para que los cambios entren en vigor.

El ejemplo de código siguiente muestra el archivo `FMIEXT.properties`, que debe personalizarse para que coincida con el entorno del sistema. Las líneas de comentarios empiezan por el signo de almohadilla (#) cuando se utiliza una página de códigos de EE.UU. Las líneas de datos solamente pueden tener una directiva y su valor asignado. Los comentarios no pueden estar en la misma línea. Las continuaciones de línea no están soportadas.

```
# Propiedades de la extensión Integración del gestor de archivos (FMI)
#
enabled=false
fmlistenport=1960
```

*Figura 30. FMIEXT.properties – archivo de configuración del gestor de archivos*

#### **habilitado**

Indica si la escucha del Gestor de archivos está o no disponible en el mismo sistema host. El valor predeterminado es `false`. Los únicos valores permitidos son `true` y `false`.

#### **fmlistenport**

Puerto utilizado por el escucha del gestor de archivos. El valor predeterminado es 1960. La comunicación en este puerto está confinada a la máquina de hospedaje.

---

## **(Opcional) Caracteres no editables**

Esta tarea de personalización no requiere asistencia, recursos especiales ni tareas de personalización especiales.

---

Algunos caracteres no se convierten correctamente entre las páginas de códigos del host (basadas en EBCDIC) y las páginas de códigos del cliente (basadas en ASCII). El editor del cliente de Developer for System z utiliza las definiciones del archivo `uchars.settings` para identificar estos caracteres no editables. Cuando abre un conjunto de datos con un carácter identificado en `uchars.settings`, el editor impondrá la modalidad de sólo lectura para evitar que se dañe el conjunto de datos cuando se guarda.

El archivo `uchars.settings` se encuentra en `/etc/rdz/`, a menos que haya especificado otra ubicación al personalizar y someter el trabajo `FEK.SFEKSAMP(FEKSETUP)`. Encontrará más detalles en: “Configuración de la personalización” en la página 15. Puede editar el archivo con el mandato `TSO OEDIT`. Tenga en cuenta que RSE debe reiniciarse para que los cambios entren en vigor. Tenga en cuenta también que no es aconsejable cambiar este archivo, a

menos que lo indique el centro de soporte de IBM.

```
# uchars.settings - puntos de código no editables
#
*          *          0D 15 25;

# DBCS (japonés, coreano y chino)
IBM-930    *          0D 15 1E 1F 25;
IBM-933    *          0D 15 1E 1F 25;
IBM-935    *          0D 15 1E 1F 25;
IBM-937    *          0D 15 1E 1F 25;
IBM-939    *          0D 15 1E 1F 25;
IBM-1390   *          0D 15 1E 1F 25;
IBM-1399   *          0D 15 1E 1F 25;
IBM-1364   *          0D 15 1E 1F 25;
IBM-1371   *          0D 15 1E 1F 25;
IBM-1388   *          0D 15 1E 1F 25;

# UNICODE
UTF-8      *          0D 0A;
UTF-16BE   *          0D 0A;
UTF-16LE   *          0D 0A;
UTF-16     *          0D 0A;
```

*Figura 31. uchars.settings - Configuración de caracteres no editables*

El archivo consta de varias entradas, con el formato siguiente:

PÁGINACÓDIGOS-HOST PÁGINACÓDIGOS-LOCAL PUNTOSCÓDIGO-HEX ;

Donde PUNTOSCÓDIGO-HEX es una lista delimitada por espacios en blanco de puntos de código hexadecimales de 2 dígitos que identifican los caracteres no editables. La lista debe terminar con un signo de punto y coma (;).

Se aplican las siguientes normas de sintaxis:

- Las líneas de comentarios empiezan por el signo de almohadilla (#) cuando se utiliza una página de códigos de EE.UU.
- Las líneas de datos solamente pueden contener datos; los comentarios no pueden estar en la misma línea.
- Puede utilizarse un asterisco (\*) para PÁGINACÓDIGOS-HOST y/o PÁGINACÓDIGOS-LOCAL. Actúa como comodín y representa todas las páginas de códigos.
- Las entradas específicas tienen preferencia sobre las entradas genéricas (comodín).
- "pc-host \*" tiene preferencia si se especifican tanto "pc-host \*" como "\* pc-local" y "pc-host pc-local" no altera temporalmente estos valores.
- Si se especifica más de una vez el mismo par de páginas de códigos, se utilizará la última entrada.

---

## (Opcional) Utilizar REXEC (o SSH)

Esta tarea de personalización no requiere asistencia, recursos especiales ni tareas de personalización especiales.

---

REXEC (Ejecución remota) es un servicio TCP/IP que permite a los clientes ejecutar un mandato en el host. SSH (Shell segura) es un servicio similar, pero en él toda la comunicación se cifra mediante SSL (Capa de sockets seguros).

Developer for System z utiliza cualquiera de los servicios para realizar acciones remotas (basadas en host) en subproyectos z/OS UNIX.

Developer for System z también puede configurarse para utilizar REXEC (o SSH) para iniciar un servidor RSE en el host. Sin embargo, tenga en cuenta que cada conexión iniciada de ese modo generará un servidor RSE independiente, cada uno de los cuales utilizará una cantidad considerable de recursos del sistema. Por tanto, este método de conexión alternativo sólo es viable para un pequeño número de conexiones.

Asimismo, dado que el método de conexión alternativo REXEC (o SSH) pasa por alto el daemon RSE, no tiene acceso a todos los servicios de host descritos en esta publicación, como por ejemplo el proceso y auditoría de servidor único. Póngase en contacto con el servicio de soporte de IBM para saber si un servicio de host determinado está soportado por el método de conexión alternativo REXEC.

**Nota:** Developer for System z utiliza la versión z/OS UNIX de REXEC, no la versión TSO.

## Acciones remotas (basadas en host) para subproyectos z/OS UNIX

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 for 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**. Consulte la sección “Configuración de REXEC (o SSH)” para saber qué puerto se utiliza.

## Método de conexión RSE alternativo

Los clientes de Developer for System z deben conocer dos valores para poder iniciar una conexión RSE a través de REXEC (o SSH):

- El directorio en el que se encuentra el script de arranque `server.zseries`. `server.zseries` se encuentra en `/etc/rdz/`, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15.

**Nota:** Para disuadir de la utilización de REXEC (o SSH) como método de inicio de sesión alternativo, `server.zseries` ya no se copia automáticamente a `/etc/rdz`. Si desea utilizar esta función, debe copiarla manualmente de `/usr/lpp/rdz/bin`, tal como se muestra en el mandato de ejemplo siguiente:

```
cp /usr/lpp/rdz/bin/server.zseries /etc/rdz
```

- El puerto que se está utilizando.  
Consulte la sección “Configuración de REXEC (o SSH)” para saber qué puerto se utiliza.

## Configuración de REXEC (o SSH)

La publicación *Communications Server IP Configuration Guide* (SC31-8775) describe los pasos necesarios para configurar REXEC (o SSH). Consulte la sección Apéndice C, “Configurar INETD”, en la página 317 para obtener consideraciones

de configuración específicas de Developer for System z (no hay pasos de configuración específicos de Developer for System z).

Un puerto común utilizado por REXEC es el 512. Para verificarlo, puede comprobar `/etc/inetd.conf` y `/etc/services` para localizar el número de puerto que se emplea.

- 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`
- 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. Su puerto común es el 22, y el nombre de servicio es `sshd`.

**Nota:** `/etc/inetd.conf` y `/etc/services` pueden tener nombres diferentes. Consulte Apéndice C, “Configurar INETD”, en la página 317 para obtener más información.

---

## (Opcional) Transacción APPC para el servicio de mandatos TSO

Necesitará ayuda de un administrador de APPC y de un administrador de WLM para realizar esta tarea de personalización, que requiere los siguientes recursos o tareas de personalización especiales:

- Actualización de LINKLIST
- Transacción APPC
- Actualización de WLM

---

El servicio de mandatos TSO puede implementarse como un programa de transacción APPC, FEKFRSRV. Esta transacción actúa como servidor de hospedaje para ejecutar los mandatos TSO que se emiten desde la estación de trabajo. No se necesita APPC en la estación de trabajo, porque el cliente se comunica con FEKFRSRV por RSE. Cada cliente puede tener una conexión activa con múltiples hosts al mismo tiempo.

**Nota:**

- Por omisión, Developer for System z utiliza la Pasarela de cliente TSO/ISPF de ISPF para acceder al servicio de mandatos TSO.
- Si no está familiarizado con APPC, consulte el Apéndice D, “Configurar APPC”, en la página 325 antes de seguir adelante con esta sección.
- RSE utiliza la API de socket REXX de TCP/IP para comunicarse con FEKFRSRV. Ello implica que la biblioteca de carga de TCP/IP, TCPIP.SEZALOAD de forma predeterminada, debe estar disponible ya sea a través de LINKLIST o de la directiva STEPLIB de `rsed.envvars`.
- RSE debe reiniciarse para que los cambios descritos entren en vigor.

## Preparación

- Las siguientes tareas son un prerrequisito y se deben realizar antes de configurar el servidor de mandatos TSO. La descripción de estas tareas se encuentra en las publicaciones mencionadas.

1. Instale, configure e inicie VTAM en su sistema z/OS. Consulte la publicación *Communications Server IP SNA Network Implementation Guide* (SC31-8777) para obtener más información.
  2. Instale, configure e inicie TCP/IP en el sistema z/OS. Hallará más información en: Apéndice B, “Configurar TCP/IP”, en la página 309.
  3. Configure e inicie APPC y el subsistema del planificador de transacciones APPC (ASCH). Hallará más información en: Apéndice D, “Configurar APPC”, en la página 325.
- Para gestionar APPC mediante paneles ISPF, se puede usar el siguiente REXX de ejemplo.

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

Figura 32. REXX para paneles ISPF 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 16. Lista de comprobación para la transacción APPC

Conocimientos técnicos	Información necesaria:	Valor
	<ul style="list-style-type: none"> <li>• Valor predeterminado</li> <li>• Dónde encontrar la respuesta</li> </ul>	
APPC	Nombre de conjunto de datos de TPDATA <ul style="list-style-type: none"> <li>• Valor predeterminado: SYS1.APPCTP</li> <li>• El valor se encuentra en SYS1.PARMLIB(APPCPMxx)</li> </ul>	
APPC	Nombre de transacción que hay que usar (puede no existir) <ul style="list-style-type: none"> <li>• Valor predeterminado: FEKFRSRV</li> <li>• Para consultar las transacciones existentes, seleccione la opción de administración de perfiles TP en el menú ISPF de APPC</li> </ul>	
APPC	Clase de transacción APPC que hay que usar <ul style="list-style-type: none"> <li>• Valor predeterminado: A</li> <li>• Las clases APPC están definidas en SYS1.PARMLIB(ASCHPMxx)</li> </ul>	

Tabla 16. Lista de comprobación para la transacción APPC (continuación)

Conocimientos técnicos	Información necesaria:	Valor
	<ul style="list-style-type: none"> <li>• Valor predeterminado</li> <li>• Dónde encontrar la respuesta</li> </ul>	
WLM/SRM	Grupo de rendimiento TSO y dominio <ul style="list-style-type: none"> <li>• No hay valor predeterminado de IBM (depende del local)</li> </ul>	
RACF	Cada usuario de Developer for System z tiene acceso a un segmento OMVS (es necesario) <ul style="list-style-type: none"> <li>• No hay valor predeterminado de IBM (depende del local)</li> <li>• El mandato TSO de RACF <b>LU userid OMVS</b> se visualizará en un segmento OMVS personal existente</li> </ul>	
RACF	Cada usuario de Developer for System z debe tener acceso de lectura (READ) a hlq.SFEKPROC(FEKFRSRV) <ul style="list-style-type: none"> <li>• No hay valor predeterminado de IBM (depende del local)</li> <li>• El mandato TSO RACF <b>LD 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</li> </ul>	

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

## Implementación

**Nota:** Los miembros de ejemplo FEKAPPC\* se encuentran en FEK.#CUST.JCL, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15.

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 FEK.#CUST.JCL(FEKAPPC). Por lo tanto, la clase que hay en FEKAPPC 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 D, “Configurar APPC”, en la página 325.

- La clase de transacción APPC que se utilice debe tener suficientes iniciadores APPC para permitir un iniciador para cada usuario simultáneo de Developer for System z.
2. Defina la transacción APPC que funcionará a modo de servidor de mandatos. Puede utilizar el JCL de ejemplo FEK.#CUST.JCL(FEKAPPCC) para definir esta transacción. Las instrucciones para personalizar este JCL se encuentran dentro del JCL.

**Nota:**

- a. Si ha cambiado el nombre del programa de transacción (el valor predeterminado es FEKFRSRV) en este paso, hay que asignar el nuevo nombre a `_FEKFSCMD_TP_NAME_`, en `rsed.envvars`, como se describe en: “rsed.envvars, archivo de configuración de RSE” en la página 32.
  - b. La transacción APPC utiliza el REXX exec FEKFRSRV, situado en FEK.SFEKPROC. No cambie esta ubicación si desea que el posible mantenimiento SMP/E se active automáticamente.
  - c. También se proporciona un JCL de ejemplo para la visualización, FEK.#CUST.JCL(FEKAPPCL), o suprimir, FEK.#CUST.JCL(FEKAPPCX), la transacción.
3. Habilite RSE para utilizar APPC descomentando la directiva `RSE_JAVA_OPTS="$_RSE_JAVA_OPTS -DTSO_SERVER=APPC"` de `rsed.envvars`, como se describe en “rsed.envvars, archivo de configuración de RSE” en la página 32.
  4. Controle la prioridad de envío del programa de transacción 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.
  5. Defina un segmento OMVS predeterminado para el sistema o bien uno individual para cada usuario de Developer for System z.
  6. Otorgue a los usuarios de Developer for System z acceso de lectura (READ) sobre FEK.SFEKPROC, la biblioteca ejecutable TSO de Developer for System z.

## Consideraciones relativas a la utilización de APPC

- Al utilizar APPC para el servicio de mandatos TSO, Developer for System z depende de que TCP/IP tenga el nombre de host correcto cuando se inicializa. 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, consulte la sección Apéndice B, “Configurar TCP/IP”, en la página 309 y *TCPIP.DATA configuration statements* de la publicación *Communications Server IP Configuration Reference* (SC31-8776). Puede probar la configuración de TCP/IP iniciando el daemon RSE con el parámetro `IVP=IVP` o con el programa de verificación de la instalación (IVP) `fekfivpt`, como se ha descrito en Capítulo 7, “Verificación de la instalación”, en la página 109.
- Al utilizar APPC para el servicio de mandatos TSO, Developer for System z requiere un socket adicional (puerto TCP/IP) para la comunicación restringida al host por cada archivo MVS abierto. Se utilizará cualquier puerto disponible. Este mecanismo de selección de puerto no puede cambiarse.
- Al utilizar APPC para el servicio de mandatos TSO, 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. Vea: “La apertura de los conjuntos de datos de MVS falla” en la página 156.

- Al utilizar APPC para el servicio de mandatos TSO para evitar configurar la Pasarela de cliente TSO/ISPF de ISPF, tenga en cuenta que otros servicios, como por ejemplo SCLM Developer Toolkit, se basan en la Pasarela de cliente TSO/ISPF.
- Consulte la sección Apéndice D, “Configurar APPC”, en la página 325 para obtener consideraciones generales acerca de la utilización de APPC.

---

## (Opcional) Limpieza de WORKAREA

Esta tarea de personalización no requiere asistencia, recursos especiales ni tareas de personalización especiales.

---

Las funciones de Pasarela de cliente TSO/ISPF de ISPF y SCLM Developer Toolkit utilizan el directorio WORKAREA para almacenar archivos de trabajo temporales, que se eliminan antes de cerrar la sesión. Sin embargo, a veces queda salida temporal, por ejemplo, si se produce un error de comunicación durante el proceso. Por esta razón, es aconsejable limpiar el directorio WORKAREA de vez en cuando.

z/OS UNIX suministra un script de shell, `skulker`, que suprime los archivos en función del directorio en el que se encuentran y de su utilización. En combinación con el daemon `cron` de z/OS UNIX, que ejecuta mandatos en fechas y horas especificadas, puede configurar una herramienta automatizada que limpie periódicamente el directorio WORKAREA. Consulte la publicación *UNIX System Services Command Reference* (SA22-7802) para obtener más información acerca del script `skulker` y el daemon `cron`.

**Nota:** El directorio WORKAREA se encuentra en `/var/rdz/`, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15.

---

## Capítulo 7. Verificación de la instalación

Después de completar la personalización del producto, puede utilizar los Programas de verificación de la instalación (IVP) descritos en este capítulo para verificar la configuración satisfactoria de componentes de productos clave.

---

### Verificar tareas iniciadas

#### JMON, Supervisor de trabajos JES

Inicie la tarea iniciada JMON (o el trabajo de usuario). La información de inicio de STDOUT de DD debe finalizar con este mensaje:

```
JM200I Inicialización del servidor completada.
```

Si el trabajo finaliza con el código de retorno 66, sabrá que FEK.SFEKAUTH no está autorizada por APF.

**Nota:** Inicie el Supervisor de trabajos JES antes de seguir el resto de pruebas IVP.

#### LOCKD, Daemon de bloqueo

Inicie la tarea iniciada LOCKD (o el trabajo de usuario). El daemon de bloqueo emite el siguiente mensaje de consola cuando el inicio se realiza correctamente:

```
FEK501I Daemon de bloqueo iniciado, puerto=4036, intervalo de limpieza=1440,  
nivel de anotaciones=1
```

#### RSED, daemon RSE

Inicie la tarea iniciada RSED (o el trabajo de usuario) con el parámetro IVP=IVP. Con este parámetro, el servidor finalizará después de realizar algunas pruebas de verificación de la instalación. La salida de estas pruebas está disponible en STDOUT de DD. En caso de errores, también habrá datos disponibles en STDERR de DD. Los datos de STDOUT deben ser similares a los del ejemplo siguiente:

**Nota:** Inicie el daemon RSE, sin el parámetro IVP, antes de seguir con el resto de pruebas IVP. El daemon RSE emite el siguiente mensaje de consola cuando el inicio se realiza correctamente:

```
FEK002I Daemon RSE iniciado. (puerto=4035)
```

Prueba IVP del daemon RSE

```
Wed Jul 2 17:11:52 2008 UTC  
uid=8(STCRSE) gid=1(STCGROUP)
```

```
El puerto del daemon RSE es 4035  
Archivos de configuración de RSE ubicados en /etc/rdz
```

```
-----  
variables de entorno actuales  
-----
```

```
@="/usr/lpp/rdz/bin/rsed.sh" @[1]="4035" @[2]="/etc/rdz"  
CGI_DTCONF="/var/rdz/scldmt"  
CGI_DTWORK="/var/rdz"  
CGI_TRANTABLE="FEK.#CUST.LSTRANS.FILE"  
CLASSPATH=".:usr/lpp/rdz/lib:usr/lpp/rdz/lib/dstore_core.jar:usr/lpp/  
ERRNO="0"  
HOME="/tmp"
```

```

IFS=""
"
JAVA_HOME="/usr/lpp/java/J5.0"
JAVA_PROPAGATE="NO"
LANG="C"
LIBPATH=".:usr/lib:usr/lpp/java/J5.0/bin:usr/lpp/java/J5.0/bin/classi
LINENO="66"
LOGNAME="STCRSE"
MAILCHECK="600"
OLDPWD="/tmp"
OPTIND="1"
PATH=".:usr/lpp/java/J5.0/bin:usr/lpp/rdz/bin:usr/lpp/ispf/bin:bin:/
PPID="33554711"
PS1="\$ "
PS2="> "
PS3="#? "
PS4="+ "
PWD="/etc/rdz"
RANDOM="27298"
RSE_CFG="/etc/rdz"
RSE_HOME="/usr/lpp/rdz"
RSE_LIB="/usr/lpp/rdz/lib"
SECONDS="0"
SHELL="/bin/sh"
STEPLIB="NONE"
TZ="EST5EDT"
_BPX_SHAREAS="YES"
_BPX_SPAWN_SCRIPT="YES"
_CEE_DMPTARG="/tmp"
_CEE_RUNOPTS="ALL31(ON) HEAP(32M,32K,ANYWHERE,KEEP,,) TRAP(ON)"
_CMDSERV_BASE_HOME="/usr/lpp/ispf"
_CMDSERV_CONF_HOME="/etc/rdz"
_CMDSERV_WORK_HOME="/var/rdz"
_RSE_CMDSERV_OPTS="&SESSION=SPAWN"
_RSE_DAEMON_CLASS="com.ibm.etools.zos.server.RseDaemon"
_RSE_DAEMON_IVP_TEST="1"
_RSE_DAEMON_PORT="4035"
_RSE_JAVAOPTS=" -DISPF_OPTS='&SESSION=SPAWN' -DA_PLUGIN_PATH=/usr/lpp/rd
_RSE_POOL_SERVER_CLASS="com.ibm.etools.zos.server.ThreadPoolProcess"
_RSE_PWD="/tmp"
_RSE_SERVER_CLASS="org.eclipse.dstore.core.server.Server"
_RSE_SERVER_TIMEOUT="120000"
_SCLMDT_BASE_HOME="/usr/lpp/rdz"
_SCLMDT_CONF_HOME="/var/rdz/sclmdt"
_SCLMDT_TRANTABLE="FEK.#CUST.LSTRANS.FILE"
_SCLMDT_WORK_HOME="/var/rdz"
_SCLM_BASE="/var/rdz/WORKAREA"
_SCLM_BWBCALL="/usr/lpp/rdz/bin/BWBCALL"
_SCLM_DWGET="/var/rdz/WORKAREA"
_SCLM_DWTRANSFER="/var/rdz/WORKAREA"
_SCLM_J2EPUT="/var/rdz/WORKAREA"

```

```

-----
prueba de inicio java...
-----

```

```

java version "1.5.0"
Java(TM) 2 Runtime Environment, Standard Edition (build pmz31dev-2008031
IBM J9 VM (build 2.3, J2RE 1.5.0 IBM J9 2.3 z/OS s390-31 j9vmmz3123-2008
J9VM - 20080314_17962_bHdSMr
JIT - 20080130_0718ifx2_r8
GC - 200802_08)
JCL - 20080314

```

```

-----
pruebe IVP de TCP/IP...
-----

```

```
Wed Jul  2 13:11:54 EDT 2008
uid=8(STCRSE) gid=1(STCGROUP)
using /etc/rdz/rsed.envvars
```

```
-----
configuración del resolvente TCP/IP (orden de búsqueda de z/OS UNIX):
-----
```

```
Inicialización de rastreo de resolvente completada -> 2008/07/02 13:11:54.745964
```

```
Valores de resolvente res_init:
Conjunto de datos Tcp/Ip global      = Ninguno
Conjunto de datos Tcp/Ip predeterminado = Ninguno
Conjunto de datos Tcp/Ip local      = /etc/resolv.conf
Tabla de conversión                  = Predeterminada
IDusuario/NombreTrabajo              = STCRSE
API llamante                        = LE C Sockets
Modalidad llamante                   = EBCDIC
(L) DataSetPrefix = TCPIP
(L) HostName      = CDFMVS08
(L) TcpIpJobName  = TCPIP
(L) DomainOrigin = RALEIGH.IBM.COM
(L) NameServer   = 9.42.206.2
                      9.42.206.3
(L) NsPortAddr   = 53          (L) ResolverTimeout   = 10
(L) ResolveVia   = UDP         (L) ResolverUdpRetries = 1
(*) Options NDots = 1
(*) SockNoTestStor
(*) AlwaysWto    = NO          (L) MessageCase      = MIXED
(*) LookUp       = DNS LOCAL
```

```
res_init Satisfactoria
```

```
res_init Iniciada: 2008/07/02 13:11:54.755363
```

```
res_init Finalizada: 2008/07/02 13:11:54.755371
```

```
*****
```

```
MVS TCP/IP NETSTAT CS V1R9      Nombre TCPIP: TCPIP      13:11:54
```

```
Tcpip iniciado a las 01:28:36 el 06/23/2008 con IPv6 habilitado
```

```
-----
dirección IP del host:
-----
```

```
hostName=CDFMVS08
hostAddr=9.42.112.75
bindAddr=9.42.112.75
localAddr=9.42.112.75
```

```
Satisfactorio, coincidencia de direcciones
```

```
-----
Prueba IVP de PassTicket...
-----
```

```
Satisfactorio, el IVP de PassTicket ha finalizado con normalidad
```

```
-----
IVP de daemon RSE finalizado
```

---

## Verificar servicios

La instalación de Developer for System z proporciona varios programas de verificación de la instalación (IVP) para los servicios básicos y opcionales. Los scripts de los IVP se encuentran en el directorio de instalación, que es /usr/lpp/rdz/bin/ por omisión.

Tabla 17. IVPs para servicios

fekfivpa	“(Opcional) Conexión del servicio de mandatos TSO mediante APPC” en la página 118
fekfivpd	“Conexión del daemon RSE” en la página 115
fekfivpi	“Conexión de la Pasarela de cliente TSO/ISPF de ISPF” en la página 116
fekfivpj	“Conexión del Supervisor de trabajos JES” en la página 115
fekfivpl	“Conexión del daemon de bloqueo” en la página 116
fekfivpr	“(Opcional) Conexión REXEC” en la página 120
fekfivps	“(Opcional) Conexión SCLMDT” en la página 118
fekfivpt	“Configuración de TCP/IP” en la página 114
fekfivpz	“(Opcional) Script de shell REXEC/SSH” en la página 121

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.

El ID de usuario que ejecuta los IVP necesita un tamaño de región grande, ya que se ejecutarán funciones que necesitan mucha memoria, tales como Java. Debe establecer el tamaño de región en 131072 kilobytes (128 megabytes) o superior.

El siguiente error de ejemplo es una indicación clara de un tamaño de región insuficiente. (Pero también pueden producirse otros errores. Por ejemplo, que no se puede iniciar Java.)

```
CEE5213S Se ha recibido la señal SIGPIPE.
%mandato de z/OS UNIX%: el número de serie 13 ha desactivado el mandato
  %número-línea% *- * %mandato REXX%
    +++ RC(137) +++
```

**Nota:** Las tareas iniciadas de Developer for System z deben estar activas antes de iniciar la prueba IVP.

## Inicialización de IVP

Para todos los mandatos de ejemplo de esta sección se espera que estén establecidas determinadas variables de entorno. De ese modo, los scripts del IVP están disponibles mediante la sentencia PATH, y se conoce la ubicación de los archivos de configuración personalizados. Utilice los mandatos **pwd** y **cd** para verificar y cambiar el directorio actual por el directorio que contiene los archivos de configuración personalizados. Luego se puede utilizar el script de shell **ivpinit** para establecer las variables de entorno de RSE, como en el ejemplo que sigue (\$ es el indicador de z/OS UNIX):

```
$ pwd
/u/userid
$ cd /etc/rdz
$ ./ivpinit
Archivos de configuración de RSE ubicados en /etc/rdz - valor predeterminado
/usr/lpp/rdz/bin añadido a PATH
```

El primer punto "." de `./ivpinit` 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 `./ivpinit` NO se ejecuta antes de los scripts `fekfivp*`, la vía de acceso a estos scripts se tiene que especificar al llamarlos, como en este ejemplo:

```
/usr/lpp/rdz/bin/fekfivpr 512 USERID
```

Asimismo, la mayoría de los scripts `fekfivp*` solicitarán la ubicación del archivo `rzed.envvars` personalizado si `./ivpinit` no se ejecuta en primer lugar.

- 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):

```
$ EXPORT STEPLIB=$STEPLIB:TCPIP.SEZALOAD
```

Tenga en cuenta que al añadir una biblioteca sin autorización APF a un `STEPLIB` existente se elimina la autorización APF de los conjuntos de datos `STEPLIB` existentes.

Tenga también en cuenta que si `CEE.SCEELKED` está en `LINKLIST` o `STEPLIB`, `TCPIP.SEZALOAD` debe colocarse antes de `CEE.SCEELKED`. Si no lo hace, se producirá una terminación anormal del sistema 0C1 para las llamadas de socket REXX de TCP/IP.

Para obtener información sobre cómo diagnosticar los problemas de conexión del RSE, consulte la sección Capítulo 9, "Resolución de problemas de configuración", en la página 137 o las fichas técnicas de la página de soporte de Developer for System z, <http://www-306.ibm.com/software/awdtools/rdz/support/>.

## Disponibilidad de puertos

La disponibilidad del supervisor de trabajos JES, del daemon RSE y, opcionalmente, de REXEC o puerto SSH 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 VxRy TCPIP Nombre TCPIP: TCPIP 13:57:36
ID us. Conexión Socket Local Socket Foráneo Estado
-----
RSED 0000004B 0.0.0.0..4035 0.0.0.0..0 Escucha
LOCKD 0000004C 0.0.0.0..4036 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 VxRy TCPIP Nombre TCPIP: TCPIP 14:03:35
ID usuario Conexión Estado
-----
RSED 0000004B Escucha
Socket local: 0.0.0.0..4035
Socket foráneo: 0.0.0.0..0
```

```

LOCKD    0000004C Escucha
Socket local:  0.0.0.0..4036
Socket foráneo: 0.0.0.0..0
JMON     00000037 Escucha
Socket local:  0.0.0.0..6715
Socket foráneo: 0.0.0.0..0

```

## Configuración de TCP/IP

Al utilizar APPC para el servicio de mandatos TSO, Developer for System z depende de que TCP/IP tenga el nombre de host correcto cuando se inicializa. Ello implica que los distintos archivos de configuración de TCP/IP y del resolvente estén configurados correctamente. Hallará más información sobre la configuración de TCP/IP y el resolvente en la sección Apéndice B, "Configurar TCP/IP", en la página 309. Verifique los valores actuales ejecutando el siguiente mandato:

```
fekfivpt
```

**Nota:** Este IVP emite el mandato **netstat** de TCPIP, que puede estar protegido contra ejecución por el software de seguridad.

El mandato debe devolver datos de salida parecidos a los de este ejemplo (\$ es el indicador de z/OS UNIX):

```
$ fekfivpt
```

```

Wed Jul  2 13:11:54 EDT 2008
uid=1(USERID) gid=0(GROUP)
using /etc/rdz/rsed.envvars

```

```

el límite de tamaño del espacio de direcciones actual es 1914675200 (1826.0 MB)
el límite de tamaño del espacio de direcciones máximo es 2147483647 (2048.0 MB)

```

```
-----
configuración del resolvente TCP/IP (orden de búsqueda de z/OS UNIX):
-----
```

```
Inicialización de rastreo de resolvente completada -> 2008/07/02 13:11:54.745964
```

```

Valores de resolvente res_init:
Conjunto de datos Tcp/Ip global      = Ninguno
Conjunto de datos Tcp/Ip predeterminado = Ninguno
Conjunto de datos Tcp/Ip local       = /etc/resolv.conf
Tabla de conversión                  = Predeterminada
IDusuario/NombreTrabajo              = USERID
API llamante                         = LE C Sockets
Modalidad llamante                   = EBCDIC
(L) DataSetPrefix = TCPIP
(L) HostName      = CDFMVS08
(L) TcpIpJobName  = TCPIP
(L) DomainOrigin  = RALEIGH.IBM.COM
(L) NameServer    = 9.42.206.2
                      9.42.206.3
(L) NsPortAddr    = 53          (L) ResolverTimeout      = 10
(L) ResolveVia    = UDP         (L) ResolverUdpRetries = 1
(*) Options NDots = 1
(*) SockNoTestStor
(*) AlwaysWto     = NO          (L) MessageCase       = MIXED
(*) LookUp        = DNS LOCAL

```

```
res_init Satisfactoria
```

```
res_init Iniciada: 2008/07/02 13:11:54.755363
```

```
res_init Finalizada: 2008/07/02 13:11:54.755371
```

```
*****
```

```
MVS TCP/IP NETSTAT CS V1R9      Nombre TCPIP: TCPIP      13:11:54
```

```
Tcpip iniciado a las 01:28:36 el 06/23/2008 con IPv6 habilitado
```

```
-----
```

dirección IP del host:

```
-----  
hostName=CDFMVS08  
hostAddr=9.42.112.75  
bindAddr=9.42.112.75  
localAddr=9.42.112.75
```

Satisfactorio, coincidencia de direcciones

## 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 como los que aparecen en el ejemplo siguiente (\$ es el indicador de z/OS UNIX):

```
$ fekfivpd 4035 USERID
```

```
Wed Jul  2 15:00:27 EDT 2008  
uid=1(USERID) gid=0(GROUP)  
using /etc/rdz/rsed.envvars
```

el límite de tamaño del espacio de direcciones actual es 1914675200 (1826.0 MB)  
el límite de tamaño del espacio de direcciones máximo es 2147483647 (2048.0 MB)

```
Contraseña:  
SSL está inhabilitado  
conectado  
8108  
570655399  
Éxito
```

**Nota:** Al probar una conexión habilitada para SSL, verifique que ha especificado el puerto correcto si obtiene este mensaje de error: 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
```

```
Wed Jul  2 15:00:27 EDT 2008  
uid=1(USERID) gid=0(GROUP)  
using /etc/rdz/rsed.envvars
```

el límite de tamaño del espacio de direcciones actual es 1914675200 (1826.0 MB)  
el límite de tamaño del espacio de direcciones máximo es 2147483647 (2048.0 MB)

```
hostName=CDFMVS08  
hostAddr=9.42.112.75  
En espera de una respuesta del supervisor de trabajos JES...  
ACKNOWLEDGE01v03
```

Éxito

## Conexión del daemon de bloqueo

Verifique la conexión del daemon de bloqueo ejecutando el siguiente mandato.

```
fekfivpl
```

El mandato debe devolver datos de salida parecidos a los de este ejemplo (\$ es el indicador de z/OS UNIX):

```
$ fekfivpl
```

```
Lun 29 Jun 08:00:38 EDT 2009
uid=1(USERID) gid=0(GROUP)
using /etc/rdz/rsed.envvars
```

```
el límite de tamaño del espacio de direcciones actual es 1914675200 (1826.0 MB)
el límite de tamaño del espacio de direcciones máximo es 2147483647 (2048.0 MB)
```

```
hostName=CDFMVS08
hostAddr=9.42.112.75
```

```
Registrando usuario en el Daemon de bloqueo...
En espera de una respuesta del daemon de bloqueo...
```

```
Consultando en el Daemon de bloqueo...
En espera de una respuesta del daemon de bloqueo...
USERID
```

```
Desregistrando usuario del Daemon de bloqueo...
En espera de una respuesta del daemon de bloqueo...
```

```
Consultando en el Daemon de bloqueo...
En espera de una respuesta del daemon de bloqueo...
```

Éxito

## Conexión de la Pasarela de cliente TSO/ISPF de ISPF

Verifique la conexión establecida con la Pasarela de cliente TSO/ISPF de ISPF emitiendo el mandato siguiente:

```
fekfivpi
```

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

```
$ fekfivpi
```

```
Wed Jul 2 15:00:27 EDT 2008
uid=1(USERID) gid=0(GROUP)
using /etc/rdz/rsed.envvars
```

```
el límite de tamaño del espacio de direcciones actual es 1914675200 (1826.0 MB)
el límite de tamaño del espacio de direcciones máximo es 2147483647 (2048.0 MB)
```

```
-----
contenido de /etc/rdz/ISPF.conf:
-----
```

```
isplib=ISP.SISPMENU
isptlib=ISP.SISPTENU
ispplib=ISP.SISPPENU
ispslib=ISP.SISPSLIB
sysproc=ISP.SISPCLIB,FEK.SFEKPROC
```

```
-----
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/rdz/lib:/usr/lpp/ispf/bin:
/bin:/usr/sbin:.
```

```
STEPLIB              = FEK.SFEKAUTH:FEK.SFEKLOAD
```

```
_CMDSERV_BASE_HOME   = /usr/lpp/ispf
_CMDSERV_CONF_HOME   = /etc/rdz
_CMDSERV_WORK_HOME    = /var/rdz
-----
```

\*\*\* CHECK : USS MODULES

```
Comprobando directorio de ISPF : /usr/lpp/ispf
Comprobando módulos en el directorio /usr/lpp/ispf/bin directory
Comprobando el archivo de configuración de ISPF ISPF.conf
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 de ISPF, se mostrará información más detallada.

fekfivpi tiene los siguientes parámetros opcionales que no dependen de la posición:

**-file** fekfivpi puede producir grandes cantidades de datos de salida (cientos de líneas). El parámetro -file envía esta salida a un archivo, userlog/.eclipse/RSE/\$LOGNAME/fekfivpi, donde userlog es el valor de la directiva user.log de rsed.envvars, y \$LOGNAME es el ID de usuario (en mayúsculas). Si la directiva user.log no tiene caracteres de comentario o no está presente, se utiliza la vía de acceso inicial. La vía de acceso inicial se define en el segmento de seguridad OMVS.

**-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.

## (Opcional) Conexión del servicio de mandatos TSO mediante APPC

Verifique la conexión establecida con el servidor de mandatos TSO (utilizando APPC), emitiendo el siguiente mandato. Sustituya IDUSUARIO por 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:

```
Wed Jul  2 15:00:27 EDT 2008
uid=1(USERID) gid=0(GROUP)
using /etc/rdz/rsed.envvars
```

el límite de tamaño del espacio de direcciones actual es 1914675200 (1826.0 MB)  
el límite de tamaño del espacio de direcciones máximo es 2147483647 (2048.0 MB)

```
20070607 13:57:18.584060 /usr/lpp/rdz/bin/fekfscmd: version=Oct 2003
20070607 13:57:18.584326 Input parms: 1.2.3.4 * NOTRACE USERID *****
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
```

## (Opcional) Conexión SCLMDT

Verifique la conexión establecida con SCLM Developer Toolkit emitiendo el mandato siguiente:

```
fekfivps
```

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):

```
$ fekfivps
```

```
Wed Jul  2 15:00:27 EDT 2008
uid=1(USERID) gid=0(GROUP)
using /etc/rdz/rsed.envvars
```

el límite de tamaño del espacio de direcciones actual es 1914675200 (1826.0 MB)  
el límite de tamaño del espacio de direcciones máximo es 2147483647 (2048.0 MB)

```
-----
contenido de /etc/rdz/ISPF.conf:
```

```
-----  
isplib=ISP.SISPMENU  
isptlib=ISP.SISPTENU  
ispllib=ISP.SISPPENU  
ispslib=ISP.SISPSLIB  
sysproc=ISP.SISPCLIB,FEK.SFEKPROC  
-----
```

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

```
-----
```

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

Server PATH = /usr/lpp/java/J5.0/bin:/usr/lpp/rdz/lib:/usr/lpp/ispf/bin:  
/bin:/usr/sbin:.

STEPLIB = FEK.SFEKAUTH:FEK.SFEKLOAD

\_CMDSERV\_BASE\_HOME = /usr/lpp/ispf  
\_CMDSERV\_CONF\_HOME = /etc/rdz  
\_CMDSERV\_WORK\_HOME = /var/rdz  
\_SCLMDT\_CONF\_HOME = /var/rdz/sclmdt  
\_SCLMDT\_WORK\_HOME = /var/rdz  
\_SCLMDT\_TRANTABLE = FEK.#CUST.LSTRANS.FILE

```
-----  
*** CHECK : JAVA PATH SETUP VERIFICATION
```

RC=0

MSG: SATISFACTORIO

```
-----  
*** CHECK : USS MODULES
```

Comprobando directorio de ISPF: /usr/lpp/ispf  
Comprobando módulos en el directorio /usr/lpp/ispf/bin  
Comprobando el archivo de configuración de ISPF ISPF.conf  
Comprobando directorio bin de instalación : /usr/lpp/rdz/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.

fekfivps tiene los siguientes parámetros opcionales que no dependen de la posición:

**-file** fekfivps puede producir grandes cantidades de datos de salida (cientos de líneas). El parámetro **-file** envía esta salida a un archivo, `userlog/.eclipse/RSE/$LOGNAME/fekfivps`, donde `userlog` es el valor de la directiva `user.log` de `rsed.envvars`, y `$LOGNAME` es el ID de usuario (en mayúsculas). Si la directiva `user.log` no tiene caracteres de comentario o no está presente, se utiliza la vía de acceso inicial. La vía de acceso inicial se define en el segmento de seguridad OMVS.

**-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.

## (Opcional) 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:

```
Wed Jul 2 15:00:27 EDT 2008
uid=1(USERID) gid=0(GROUP)
using /etc/rdz/rsed.envvars
```

```
el límite de tamaño del espacio de direcciones actual es 1914675200 (1826.0 MB)
el límite de tamaño del espacio de direcciones máximo es 2147483647 (2048.0 MB)
```

```
$ EYZRC01I Llamando a la función rexec_af con:
EYZRC02I Host: CDFMVS08, user USERID, cmd cd /etc/rdz;export RSE_USER_ID=USERI
D;./server.zseries -ivp, port 512
EYZRC19I Socket de datos = 4, Socket de control = 6.
```

Prueba IVP del servidor RSE

```
CDFMVS08 -- Mie 2 Jul 15:00:27 EDT 2008
uid=1(USERID) gid=0(GROUP)
```

Archivos de configuración de RSE ubicados en `/etc/rdz` --valor predeterminado

El ID de usuario de RSE es USERID --valor predeterminado

-----  
Límites de tamaño del espacio de direcciones

-----  
el límite de tamaño del espacio de direcciones actual es 2147483647 (2048.0 MB)  
el límite de tamaño del espacio de direcciones máximo es 2147483647 (2048.0 MB)

-----  
historial de servicio

-----  
Vie 19 Jun 00:01:00 2009 -- COPY -- HHOP760 v7600 creado 18 Jun 2009

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

-----  
iniciando servidor RSE en el fondo -- Vie 19 Jun 15:59:05 EDT 2009

```
-----
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 j9vmnz3123-20070201 (JIT
enabled)
J9VM - 20070131_11312_bHdSMr
JIT - 20070109_1805ifx1_r8
GC - 200701_09)
JCL - 20070126
```

```
Servidor DStore iniciándose...
El servidor se ha iniciado satisfactoriamente
8108
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).
- Puede probar el script de shell utilizado por REXEC por separado, como se describe en la siguiente prueba de IVP, "(Opcional) Script de shell REXEC/SSH".
- El servidor se inicia sin un cliente que intente conectarse, por lo que se agotará el tiempo de espera (al cabo de 5 segundos). Esto provocará un mensaje de error de conexión como el del ejemplo siguiente:

```
Error de conexión
Servidor: error al inicializar el socket: java.net.SocketTimeoutException:
La aceptación ha agotado el tiempo de espera
```

## (Opcional) Script de shell REXEC/SSH

Es posible saltarse esta prueba de IVP si la prueba anterior (descrita someramente en: "(Opcional) Conexión REXEC" en la página 120) se llevó a cabo satisfactoriamente.

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

```
fekfivpz
```

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):

```
$ fekfivpz
```

```
Wed Jul 2 15:00:27 EDT 2008
uid=1(USERID) gid=0(GROUP)
```

```
using /etc/rdz/rsed.envvars
```

```
el límite de tamaño del espacio de direcciones actual es 1914675200 (1826.0 MB)
el límite de tamaño del espacio de direcciones máximo es 2147483647 (2048.0 MB)
```

```
-----
Prueba IVP del servidor RSE
```

```
CDFMVS08 -- Mie 2 Jul 15:00:27 EDT 2008
uid=1(USERID) gid=0(GROUP)
```

Archivos de configuración de RSE ubicados en /etc/rdz - valor predeterminado

El ID de usuario de RSE es USERID --valor predeterminado

-----  
Límites de tamaño del espacio de direcciones  
-----

el límite de tamaño del espacio de direcciones actual es 2147483647 (2048.0 MB)  
el límite de tamaño del espacio de direcciones máximo es 2147483647 (2048.0 MB)

-----  
historial de servicio  
-----

Vie 19 Jun 00:01:00 2009 -- COPY -- HHOP760 v7600 creado 18 Jun 2009

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

-----  
iniciando servidor RSE en el fondo -- Vie 19 Jun 15:59:05 EDT 2009  
-----

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

Servidor DStore iniciándose...  
El servidor se ha iniciado satisfactoriamente  
8108  
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 de 2096128).
- El servidor se inicia sin un cliente que intente conectarse, por lo que se agotará el tiempo de espera (al cabo de 5 segundos). Esto provocará un mensaje de error de conexión como el del ejemplo siguiente:

Error de conexión

Servidor: error al inicializar el socket: java.net.SocketTimeoutException:  
La aceptación ha agotado el tiempo de espera

---

## Parte 2. Información de Developer for System z



---

## Capítulo 8. Mandatos de operador

Este capítulo ofrece una visión general de los mandatos de operador (o consola) disponibles para Developer for System z. Consulte la sección “Cómo leer un diagrama de sintaxis” en la página 134 si no está familiarizado con los diagramas de sintaxis utilizados para describir el formato de los mandatos.

---

### Iniciar (S)

Utilice el mandato Iniciar (**START**) para iniciar dinámicamente una tarea iniciada (STC). La versión abreviada del mandato es la letra S.

### Supervisor de trabajos JES

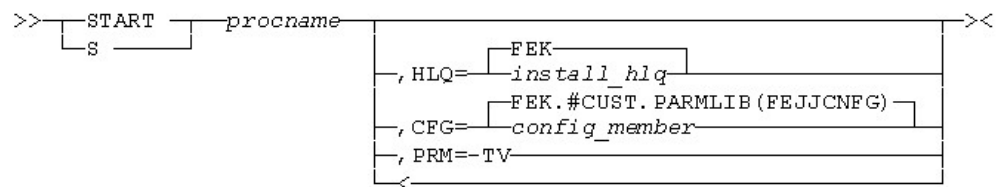


Figura 33. Mandato de operador START JMON

#### **nombreproc**

El nombre del miembro de una biblioteca de procedimientos utilizado para iniciar el servidor. El nombre predeterminado utilizado durante la configuración del host es JMON.

#### **HLQ=hlq\_instalación**

Calificador de alto nivel utilizado para instalar Developer for System z. El valor predeterminado es FEK.

#### **CFG=miembro\_config**

Nombre absoluto de conjunto de datos y miembro del archivo de configuración del Supervisor de trabajos JES. El valor predeterminado es FEK.#CUST.PARMLIB (FEJJCNFG).

#### **PRM=-TV**

Habilitar la modalidad verbosa (rastreo). El rastreo afectará negativamente al rendimiento y solo debe realizarse bajo indicación del centro de soporte de IBM.

## Daemon RSE

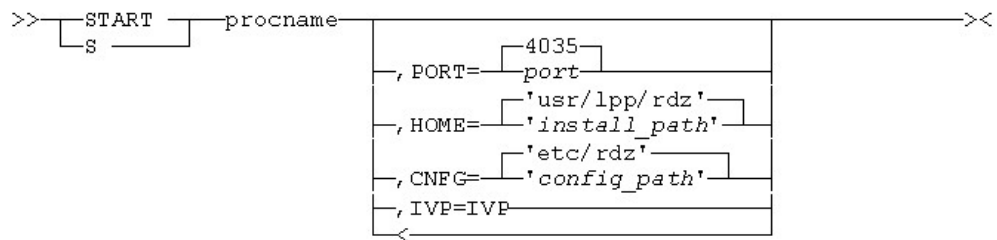


Figura 34. Mandato de operador `START RSED`

### **nombreproc**

El nombre del miembro de una biblioteca de procedimientos utilizado para iniciar el servidor. El nombre predeterminado utilizado durante la configuración del host es `RSED`.

### **PORT=puerto**

El puerto utilizado por el daemon RSE para que se conectan los clientes. El valor predeterminado es 4035.

### **HOME='vía\_instalación'**

Prefijo de vía de acceso y el `/usr/lpp/rdz` obligatorio utilizados para instalar Developer for System z. El valor predeterminado es `'/usr/lpp/rdz'`. Tenga en cuenta que la vía de acceso de z/OS UNIX es sensible a las mayúsculas y minúsculas y debe especificarse entre comillas simples (') para conservar los caracteres en minúsculas.

### **CNFG='vía\_config'**

Ubicación absoluta de los archivos de configuración almacenados en z/OS UNIX. El valor predeterminado es `'/etc/rdz'`. Tenga en cuenta que la vía de acceso de z/OS UNIX es sensible a las mayúsculas y minúsculas y debe especificarse entre comillas simples (') para conservar los caracteres en minúsculas.

### **IVP=IVP**

No iniciar el servidor, pero iniciar el programa de verificación de instalación (IVP) del daemon RSE.

## Daemon de bloqueo

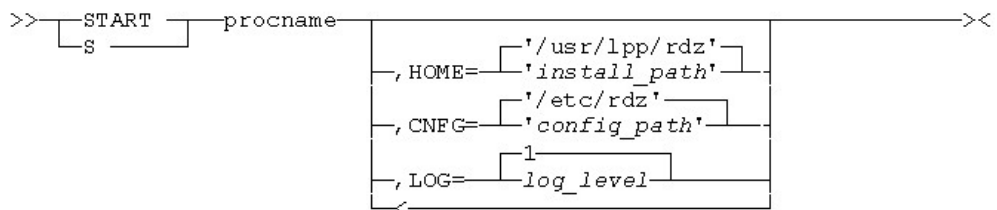


Figura 35. Mandato de operador `START LOCKD`

### **nombreproc**

El nombre del miembro de una biblioteca de procedimientos utilizado para iniciar el servidor. El nombre predeterminado utilizado durante la configuración del host es `LOCKD`.

**HOME='vía\_instalación'**

Prefijo de vía de acceso y el /usr/lpp/rdz obligatorio utilizados para instalar Developer for System z. El valor predeterminado es '/usr/lpp/rdz'. Tenga en cuenta que la vía de acceso de z/OS UNIX es sensible a las mayúsculas y minúsculas y debe especificarse entre comillas simples (') para conservar los caracteres en minúsculas.

**CNFG='vía\_config'**

Ubicación absoluta de los archivos de configuración almacenados en z/OS UNIX. El valor predeterminado es '/etc/rdz'. Tenga en cuenta que la vía de acceso de z/OS UNIX es sensible a las mayúsculas y minúsculas y debe especificarse entre comillas simples (') para conservar los caracteres en minúsculas.

**LOG=nivel\_ anotaciones**

El nivel de detalle de salida en STDOUT de DD.

- 0 : anotar sólo mensajes de error.
- 1 : anotar mensajes de error y aviso (valor predeterminado).
- 2 : anotar mensajes de error, aviso e informativos.

---

## Modificar (F)

El mandato Modificar (**MODIFY**) permite consultar y cambiar dinámicamente las características de una tarea activa. La versión abreviada del mandato es la letra F.

## Supervisor de trabajos JES

```
>> [MODIFY | F] [procname] [, APPL=-TV | , APPL=-TN] ><
```

Figura 36. Mandato de operador MODIFY JMON

**nombreproc**

El nombre del miembro de una biblioteca de procedimientos utilizado para iniciar el servidor. El nombre predeterminado utilizado durante la configuración del host es JMON.

- TV Habilitar la modalidad verbosa (rastreo). El rastreo afectará negativamente al rendimiento y solo debe realizarse bajo indicación del centro de soporte de IBM.
- TN Inhabilitar la modalidad verbosa (rastreo).

## Daemon RSE

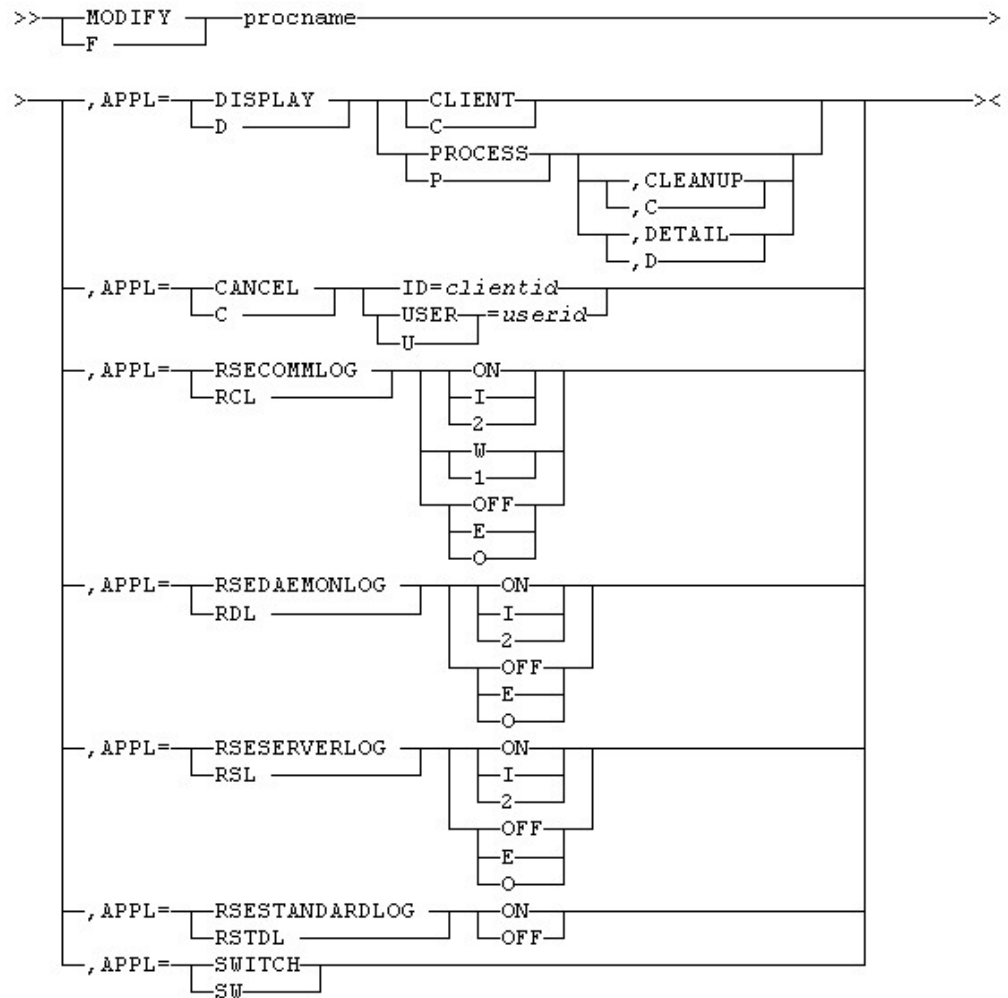


Figura 37. Mandato de operador MODIFY RSED

### nombrepoc

El nombre del miembro de una biblioteca de procedimientos utilizado para iniciar el servidor. El nombre predeterminado utilizado durante la configuración del host es RSED.

### DISPLAY CLIENT

Visualizar los clientes activos.

<IDcliente> : <IDusuario> : <conectado desde>

### DISPLAY PROCESS[,CLEANUP,DETAIL]

Visualizar los procesos de agrupaciones de hebras RSE. Puede haber varios procesos, utilizados para el equilibrio de carga de los usuarios conectados.

ProcessId(<IDproceso>) Memory Usage(<utilización almacenamiento dinámico java>%)  
Clients(<número de clientes>) Order(<orden de inicio>) <estado de error>

### Nota:

- <processid> se puede utilizar en los mandato de operador de z/OS UNIX.

- Cada proceso tiene su propio almacenamiento dinámico Java, cuyo tamaño puede establecerse en `rsed.envvars`.
- `<orden de inicio>` es un número secuencial que indica el orden por el que se iniciaron las agrupaciones de hebras. El número corresponde al número utilizado en el nombre de archivo de los archivos `stderr.*.log` y `stdout.*.log`.

En los casos normales, `<estado de error>` aparece en blanco. Tabla 18 documenta los posibles valores que no estén en blanco para el `<estado de error>`.

Tabla 18. Estado de error de la agrupación de hebras

Estado	Descripción
*error grave*	El proceso de agrupaciones de hebras ha encontrado un error no recuperable y ha interrumpido las operaciones. El resto de campos de estado muestran los últimos valores conocidos. Utilice la opción <b>CLEANUP</b> del mandato de modificación <b>DISPLAY PROCESS</b> para eliminar esta entrada de la tabla.
*proceso desactivado*	Java, z/OS UNIX o un mandato de operador ha desactivado el proceso de agrupaciones de hebras. El resto de campos de estado muestran los últimos valores conocidos. Utilice la opción <b>CLEANUP</b> del mandato de modificación <b>DISPLAY PROCESS</b> para eliminar esta entrada de la tabla.
*tiempo de espera*	El proceso de agrupaciones de hebras no ha respondido a tiempo al daemon RSE durante una petición de conexión de cliente. El resto de campos de estado muestran los últimos valores conocidos. La agrupación de hebras queda excluida de futuras peticiones de conexión de clientes. El estado <code>*tiempo de espera*</code> se restablece cuando un cliente que se sirve de esta agrupación de hebras finaliza la sesión.

Cuando se utiliza la opción **DETAIL** del mandato de modificación **DISPLAY PROCESS**, se proporciona más información:

```
ProcessId(33555087) ASId(002E) JobName(RSED8) Order(1)
PROCESS LIMITS:  CURRENT  HIGHWATER  LIMIT
JAVA HEAP USAGE(%)  10      56      100
CLIENTS             0       25      60
MAXFILEPROC         83     103    64000
MAXPROCUSER         97     99     200
MAXTHREADS          9      14     1500
MAXTHREADTASKS      9      14     1500
```

El campo **ASId** es el ID de espacio de direcciones en notación hexadecimal. La tabla de límites de proceso muestra la utilización de recursos actual, la marca de límite superior de la utilización de recursos y el límite de recursos. Fíjese en que, debido a otros factores de limitación, es posible que nunca se alcance el límite definido.

**CANCEL ID=idcliente**

Cancelar una conexión de cliente en función del ID de cliente, que se muestra en el mandato de modificación **DISPLAY CLIENT**.

**CANCEL USER=idusuario**

Cancelar una conexión de cliente en función del ID del cliente, que se muestra en el mandato de modificación **DISPLAY CLIENT**.

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

Controlar el nivel de detalle de rastreo del servidor RSE (rsecomm.log) y los servicios de conjunto de datos de MVS (lock.log y ffs\*.log). El valor de inicio predeterminado se define en rsecomm.properties. Existen tres niveles de detalle disponibles:

E o 0 u OFF	Sólo mensajes de error.
W o 1	Mensajes de error y aviso. Este es el valor predeterminado de rsecomm.properties.
I o 2 u ON	Mensajes de error, aviso e informativos.

El rastreo detallado afectará negativamente al rendimiento y solo debe realizarse bajo indicación el centro de soporte de IBM.

**RSEDAEMONLOG {ON,OFF,I,E,2,0}**

Controlar el nivel de detalle de rastreo del daemon RSE (rsedaemon.log). El valor de inicio predeterminado se define en rsecomm.properties. Existen dos niveles de detalle disponibles:

E o 0 u OFF	Sólo mensajes de error.
I o 2 u ON	Mensajes de error, aviso e informativos.

El rastreo detallado afectará negativamente al rendimiento y solo debe realizarse bajo indicación el centro de soporte de IBM.

**RSESERVERLOG {ON,OFF,I,E,2,0}**

Controlar el nivel de detalle de rastreo de las agrupaciones de hebras RSE (rseserver.log). El valor de inicio predeterminado se define en rsecomm.properties. Existen dos niveles de detalle disponibles:

E o 0 u OFF	Sólo mensajes de error.
I o 2 u ON	Mensajes de error, aviso e informativos.

El rastreo detallado afectará negativamente al rendimiento y solo debe realizarse bajo indicación el centro de soporte de IBM.

**RSESTANDARDLOG {ON,OFF}**

Inhabilitar (OFF) o habilitar (ON) la actualización de archivos de anotaciones que contienen las secuencias stdout y stderr de las agrupaciones de hebras (stdout\*.log y stderr\*.log). El valor de inicio predeterminado se define en la directivaenable.standard.log de rsed.envvars.

El rastreo detallado afectará negativamente al rendimiento y solo debe realizarse bajo indicación el centro de soporte de IBM.

**SWITCH**

Conmutar a un nuevo archivo de anotaciones de auditoría.

**Nota:**



iniciar el servidor. Los nombres predeterminados utilizados durante la configuración del host son JMON, RSED y LOCKD para el Supervisor de trabajos JES, el daemon RSE y el daemon de bloqueo, respectivamente.

## Mensajes de consola

### Supervisor de trabajos JES

El Supervisor de trabajos JES no tiene mensajes de consola específicos de producto. El servidor se basa en z/OS y JES para generar mensajes de consola para las acciones realizadas por los clientes de Developer for System z.

### Daemon RSE, servidor de agrupaciones de hebras RSE y daemon de bloqueo

Tabla 19 lista los mensajes de consola específicos de producto generados por el daemon RSE, el servidor de agrupaciones de hebras RSE y el daemon de bloqueo.

Tabla 19. Mensajes de consola de RSE

ID de mensaje	Texto del mensaje
FEK001I	RseDaemon inicializado en modalidad de {0} bits
FEK002I	El daemon RSE se ha inicializado. (puerto={0})
FEK003I	El mandato Detener se está procesando
FEK004I	RseDaemon: tamaño de almacenamiento dinámico={0}MB y tamaño de AS privado={1}MB
FEK005I	El proceso del servidor se ha iniciado. (processId={0})
FEK009I	El daemon RSE está esperando a que se inicie el proceso del servidor.
FEK010I	(ubicación de rsed.envvars = {0})
FEK011I	(directorio de anotaciones = {0})
FEK100E	Valor de puerto/tiempo de espera del daemon debe constar de dígitos
FEK101E	Es necesario JRE {0} o posterior
FEK102E	Se han recibido argumentos no válidos: {0}
FEK103E	Disco casi lleno en {0}
FEK104E	Se ha alcanzado el número máximo de procesos
FEK105E	Error al enviar datos de auditoría (rc={0})
FEK110E	socket() ha fallado. razón=({0})
FEK111E	setsockopt() ha fallado. razón=({0})
FEK112E	bind() ha fallado. razón=({0})
FEK113E	listen() ha fallado. razón=({0})
FEK114E	accept() ha fallado. razón=({0})
FEK115E	write() ha fallado. razón=({0})
FEK116E	pipe() ha fallado. razón=({0})
FEK117E	socketpair() ha fallado. razón=({0})
FEK118E	select() ha fallado. razón=({0})
FEK119E	_console() ha fallado. razón=({0})
FEK130E	gsk_environment_open() ha fallado. razón=({0})

Tabla 19. Mensajes de consola de RSE (continuación)

ID de mensaje	Texto del mensaje
FEK131E	gsk_attribute_set_enum(PROTOCOLO_GSK_SSLV2) ha fallado. razón=({0})
FEK132E	gsk_attribute_set_enum(PROTOCOLO_GSK_SSLV3) ha fallado. razón=({0})
FEK133E	gsk_attribute_set_enum(PROTOCOLO_GSK_TLSV1) ha fallado. razón=({0})
FEK134E	gsk_attribute_set_buffer(ARCHIVO_ANILLO-CLAVES_GSK) ha fallado. razón=({0})
FEK135E	gsk_attribute_set_buffer(CONT_ANILLO-CLAVES_GSK) ha fallado. razón=({0})
FEK136E	gsk_environment_init() ha fallado. razón=({0})
FEK137E	gsk_secure_socket_open() ha fallado. razón=({0})
FEK138E	gsk_attribute_set_numeric_value(FD_GSK) ha fallado. razón=({0})
FEK139E	gsk_attribute_set_buffer(ETIQUETA_ANILLO-CLAVES_GSK) ha fallado. razón=({0})
FEK140E	gsk_attribute_set_enum(TIPO_SESIÓN_GSK) ha fallado. razón=({0})
FEK141E	gsk_attribute_set_callback(DEVOLUCIÓN-LLAMADA_ES_GSK) ha fallado. razón=({0})
FEK142E	gsk_secure_socket_init() ha fallado. razón=({0})
FEK143E	gsk_attribute_set_enum(GSK_CLIENT_AUTH_TYPE) ha fallado. razón=({0})
FEK144E	gsk_get_cert_info ha fallado. razón=({0})
FEK145E	gsk_secure_socket_read() ha fallado. razón=({0})
FEK146E	gsk_secure_socket_write() ha fallado. razón=({0})
FEK150E	El daemon RSE ha terminado anormalmente; {0}
FEK201I	El mandato {0} se ha procesado
FEK202E	Se ha especificado un mandato no válido
FEK203E	Mandato de visualización no válido: Display Process   Client
FEK204E	Mandato de cancelación no válido: Cancel ID=   User=
FEK205E	El mandato no se ha procesado debido a SWITCH consecutivos
FEK206E	El recurso de anotaciones de auditoría no está activo
FEK207I	No hay ningún cliente que visualizar
FEK208I	Se ha cancelado {0}
FEK209I	No hay ningún proceso que visualizar
FEK210I	{0} cancelado debido a inicio de sesión duplicado
FEK501I	Daemon de bloqueo iniciado, puerto={0}, intervalo de limpieza={1}, nivel de anotaciones={2}
FEK502I	Daemon de bloqueo, finalizando
FEK510E	Daemon de bloqueo, falta el puerto
FEK511E	Daemon de bloqueo, puerto erróneo, puerto={0}
FEK512E	Daemon de bloqueo, error del socket, puerto={0}
FEK513W	Daemon de bloqueo, ha fallado el registro, ASID={0}, TCB={1}, USER={2}

Tabla 19. Mensajes de consola de RSE (continuación)

ID de mensaje	Texto del mensaje
FEK514W	Daemon de bloqueo, nivel de anotaciones erróneo, nivel de anotaciones={0}
BPXM023I	(stclock) dataset[(member)] NOT LOCKED
BPXM023I	(stclock) dataset[(member)] LOCKED BY userid
BPXM023I	(stclock) command, WRONG COMMAND
BPXM023I	(stclock) command, MISSING ARGUMENT
BPXM023I	(stclock) argument, WRONG ARGUMENT

## Cómo leer un diagrama de sintaxis

El diagrama de sintaxis muestra cómo especificar un mandato para que el sistema operativo pueda interpretar correctamente lo que se escribe. Un diagrama de sintaxis se lee de izquierda a derecha y de arriba a abajo, siguiendo la línea horizontal (la vía de acceso principal).

### Símbolos

En los diagramas de sintaxis se utilizan los símbolos siguientes:

Símbolo	Descripción
>>	Marca el principio del diagrama de sintaxis.
>	Indica que el diagrama de sintaxis continúa.
	Marca el principio y el final de un fragmento o parte del diagrama de sintaxis.
><	Marca el final del diagrama de sintaxis.

### Operandos

En los diagramas de sintaxis se utilizan los tipos de operandos siguientes:

- Los operandos obligatorios se visualizan en la línea principal de la vía de acceso:

```
>>—OPERANDO_OBLIGATORIO—><
```

- Los operandos opcionales se visualizan debajo de la línea principal de la vía de acceso:

```
>>└──┬──┐><
      |OPERANDO OPCIONAL|
```

- Los operandos predeterminados se visualizan encima de la línea principal de la vía de acceso:

```
>>┌──┬──┐><
      |OPERANDO_PREDETERMINADO|
```

Los operandos se clasifican como palabras clave o variables:

- Las palabras clave son constantes que deben suministrarse. Si la palabra clave aparece en el diagrama de sintaxis en mayúsculas y minúsculas, la parte en mayúsculas corresponde a la abreviatura de la palabra clave (por ejemplo, PALABRAclave). Las palabras clave no son sensibles a las mayúsculas y minúsculas. Puede codificarlas en minúsculas o en mayúsculas.

- Las variables aparecen en minúscula cursiva y representan nombres o valores suministrados por el usuario. Por ejemplo, un nombre de conjunto de datos es una variable. Las variables pueden ser sensibles a las mayúsculas y minúsculas.

## Ejemplo de sintaxis

En el ejemplo siguiente, el mandato `USER` es una palabra clave. El parámetro de variable obligatorio es `id_usuario` y el parámetro de variable opcional es contraseña. Sustituya los parámetros de variable por sus propios valores:

```
>>—USER—id_usuario—┐—————><
                    └──┴──┘
                    contraseña
```

## Caracteres no alfanuméricos y espacios en blanco

Si un diagrama muestra un carácter que no es alfanumérico (como por ejemplo paréntesis, puntos, comas, signos de igual y espacios en blanco), debe codificar el carácter como parte de la sintaxis. En este ejemplo, debe codificar `OPERAND=(001 0.001)`:

```
>>—OPERAND—==—(—001— —0.001—)—————><
```

## Seleccionar más de un operando

Una flecha que señala hacia la izquierda en un grupo de operandos indica que puede seleccionarse más de uno o que uno de ellos puede repetirse:

```
>>—┐—————><
    └──┴──┘
    └──┴──┘
    <—┐
    OPERANDO_REPETIBLE_1
    OPERANDO_REPETIBLE_2
```

## Longitud superior a una línea

Si un diagrama ocupa más de una línea, la primera línea finaliza con una sola punta de flecha y la segunda línea empieza por una sola punta de flecha:

```
>>—| La primera línea de un diagrama de sintaxis que ocupa más de una línea |—>
>—| La continuación de los submandatos, parámetros o ambos |—————><
```

## Fragmentos de sintaxis

Es posible que algunos diagramas contengan fragmentos de sintaxis, que sirven para dividir los diagramas demasiado largos, complejos o con demasiadas repeticiones. Los nombres de los fragmentos de sintaxis se especifican en mayúsculas/minúsculas mezcladas y se muestran en el diagrama y en la cabecera del fragmento. El fragmento se coloca debajo del diagrama principal:

```
>>—| Fragmento de sintaxis |—————><
```

Fragmento de sintaxis:

```
|—1ER_OPERANDO—,—2º_OPERANDO—,—3ER_OPERANDO—|
```



---

## Capítulo 9. Resolución de problemas de configuración

Este capítulo se propone ayudarle a resolver algunos problemas comunes que pueden surgir durante la configuración de Developer for System z, y tiene las secciones siguientes:

- “Anotar y configurar el análisis mediante FEKLOGS”
- “Archivos de anotaciones” en la página 138
- “Archivos de vuelco” en la página 143
- “Rastreo” en la página 146
- “Bits de permiso de z/OS UNIX” en la página 148
- “Puertos TCP/IP reservados” en la página 151
- “Tamaño del espacio de direcciones” en la página 153
- “Transacción APPC y el servicio de mandatos TSO” en la página 154
- “Información miscelánea” en la página 156

Encontrará más información en la sección de soporte del sitio Web de Developer for System z (<http://www-306.ibm.com/software/awdtools/rdz/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 (<http://www-306.ibm.com/software/awdtools/rdz/library/>) también hallará la versión más reciente de la documentación de Developer for System z, incluidos los documentos.

Developer for System z Information Center (<http://publib.boulder.ibm.com/infocenter/ratdevz/v7r6/index.jsp>) describe el cliente Developer for System z y cómo interactúa con el host (desde la perspectiva del cliente).

También encontrará información valiosa en la biblioteca Internet de z/OS, disponible en <http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/>.

Infórmenos si opina que a Developer for System z le falta alguna función. Puede abrir una Solicitud de mejora (RFE) en

<https://www.ibm.com/developerworks/support/rational/rfe/>

---

### Anotar y configurar el análisis mediante FEKLOGS

Developer for System z proporciona un trabajo de ejemplo, FEKLOGS, que reúne todos los archivos de anotaciones de z/OS UNIX, así como la información de instalación y configuración de Developer for System z.

El trabajo de ejemplo FEKLOGS se encuentra en FEK.#CUST.JCL, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: “Configuración de la personalización” en la página 15.

La personalización de FEKLOGS se describe en el JCL. La personalización abarca la provisión de algunas variables clave.

**Nota:** Los clientes de SDSF puede utilizar el mandato de línea **XDC** en SDSF para guardar la salida del trabajo en un conjunto de datos, que a su vez se puede entregar al centro de soporte de IBM.

---

## Archivos de anotaciones

Developer for 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 en su sistema host z/OS. 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:

- userlog/\$LOGNAME/

Los archivos de anotaciones específicos del usuario están ubicados en userlog/\$LOGNAME/, donde userlog es el valor combinado de la directivas user.log y DSTORE\_LOG\_DIRECTORY de rsed.envvars, y \$LOGNAME es el ID de usuario de inicio de sesión (en mayúsculas). Si la directiva user.log no tiene caracteres de comentario o no está presente, se utiliza la vía de acceso inicial del usuario. La vía de acceso inicial se define en el segmento de seguridad OMVS del ID de usuario. Si la directiva DSTORE\_LOG\_DIRECTORY no tiene caracteres de comentario o no está presente, se añade .eclipse/RSE/ al valor user.log.

- .dstoreMemLogging - Anotaciones de utilización de memoria de almacén de datos
- .dstoreTrace - Anotaciones de acciones de almacén de datos
- fa.log - Anotaciones de la Integración de analizador de errores
- fekfivpi.log - Anotaciones de la prueba IVP de fekfivpi
- fekfivps.log - Anotaciones de la prueba IVP de fekfivps
- ffs.log - Anotaciones del servidor FFS (Foreign File System), que ejecuta funciones nativas de MVS
- ffsgget.log - Anotaciones del lector de archivos, que lee un conjunto de datos secuencial (SDS) o un miembro PDS
- ffsgput.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 y las anotaciones de comunicación de todos los servicios basados en RSE (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

**Nota:** El directorio .eclipse y los archivos de anotaciones .dstore\* empiezan por un punto (.), que hace que queden ocultos. Utilice el mandato de z/OS UNIX **ls -IA** para listar los archivos y directorios ocultos. Al utilizar el cliente Developer for System z, seleccione la página de preferencias **Ventana > Preferencias... > Sistemas remotos > Archivos** y habilite "Mostrar archivos ocultos".

- **daemon-home**

Los archivos de anotaciones específicos del daemon RSE y de la agrupación de hebras RSE se encuentran en `daemon-home`, donde `daemon-home` es el valor de la directiva `daemon.log` de `rsed.envvars`. Si la directiva `daemon.log` no tiene caracteres de comentario o no está presente, se utiliza el directorio inicial del ID de usuario asignado a la tarea iniciada RSED. El directorio inicial se define en el segmento de seguridad OMVS del ID de usuario.

- `rsedaemon.log` - Anotaciones del daemon RSE
- `rseserver.log` - Anotaciones de las agrupaciones de hebras RSE
- `audit.log` - Seguimiento de auditoría de RSE
- `serverlogs.count` - Contador para anotar las secuencias de agrupaciones de hebras RSE
- `stderr.*.log` - Secuencia de error estándar de agrupaciones de hebras RSE
- `stdout.*.log` - Secuencia de salida estándar de agrupaciones de hebras RSE

**Nota:** Existen mandatos de operador disponibles para controlar la cantidad de datos grabados en algunos de los archivos de anotaciones mencionados. Hallará más información en: Capítulo 8, “Mandatos de operador”, en la página 125.

## Anotaciones del Supervisor de trabajos JES

- **DD SYSOUT**

Anotaciones de las operaciones normales. El valor predeterminado del JCL de ejemplo `FEK.#CUST.PROCLIB(JMON)` es `SYSOUT=*`.

- **DD SYSPRINT**

Anotaciones de rastreo. El valor predeterminado del JCL de ejemplo `FEK.#CUST.PROCLIB(JMON)` es `SYSOUT=*`. El rastreo se activa con el parámetro `-TV`; hallará más detalles en: “Rastreo del Supervisor de trabajos JES” en la página 146.

## Anotaciones del daemon de bloqueo

- **STDOUT DD**

Los datos redirigidos de `stdout`, la salida estándar de Java. El valor predeterminado del JCL de ejemplo `FEK.#CUST.PROCLIB(LOCKD)` es `SYSOUT=*`.

- **STDERR DD**

Los datos redirigidos de `stderr`, la salida de errores estándar de Java. El valor predeterminado del JCL de ejemplo `FEK.#CUST.PROCLIB(LOCKD)` es `SYSOUT=*`.

## Daemon RSE y anotaciones de la agrupaciones de hebras

- **STDOUT DD**

Los datos redirigidos de `stdout`, la salida estándar de Java del daemon RSE. El valor predeterminado del JCL de ejemplo `FEK.#CUST.PROCLIB(RSED)` es `SYSOUT=*`.

- **STDERR DD**

Los datos redirigidos de `stderr`, la salida de error estándar de Java del daemon RSE. El valor predeterminado del JCL de ejemplo `FEK.#CUST.PROCLIB(RSED)` es `SYSOUT=*`.

- **daemon-home**

Los archivos de anotaciones específicos del daemon RSE y de la agrupación de hebras RSE se encuentran en `daemon-home`, donde `daemon-home` es el valor de la

directiva `daemon.log` de `rsed.envvars`. Si la directiva `daemon.log` no tiene caracteres de comentario o no está presente, se utiliza el directorio inicial del ID de usuario asignado a la tarea iniciada RSED. El directorio inicial se define en el segmento de seguridad OMVS del ID de usuario.

- `rsedaemon.log` - Anotaciones del daemon RSE
- `rserver.log` - Anotaciones de las agrupaciones de hebras RSE
- `audit.log` - Seguimiento de auditoría de RSE
- `serverlogs.count` - Contador para anotar las secuencias de agrupaciones de hebras RSE
- `stderr.*.log` - Secuencia de error estándar de agrupaciones de hebras RSE
- `stdout.*.log` - Secuencia de salida estándar de agrupaciones de hebras RSE

**Nota:**

- `serverlogs.count`, `stderr.*.log` y `stdout.*.log` solamente se crean si la directiva `enable.standard.log` de `rsed.envvars` está activa, o si la función está activada dinámicamente con el mandato de operador **modify rsestandardlog on**.
- El \* de `stderr.*.log` y `stdout.*.log` es 1 de forma predeterminada. Sin embargo, puede haber varias agrupaciones de hebras RSE, en cuyo caso este número aumentará por cada nueva agrupación de hebras RSE a fin de asegurar que los nombres de los archivos sean exclusivos.
- No hay archivos de anotaciones `stdout.log` y `stderr.log` específicos del usuario cuando la directiva `enable.standard.log` está activa. Ahora se escriben datos específicos del usuario en la secuencia de agrupación de hebras RSE coincidente.
- Existen mandatos de operador disponibles para controlar la cantidad de datos grabados en algunos de los archivos de anotaciones mencionados. Hallará más información en: Capítulo 8, “Mandatos de operador”, en la página 125.

## Anotaciones de usuario de RSE

- `userlog/$LOGNAME/`

Los componentes relacionados con RSE crean varios archivos de anotaciones. Todos están ubicados en `userlog/$LOGNAME/`, donde `userlog` es el valor combinado de la directivas `user.log` y `DSTORE_LOG_DIRECTORY` de `rsed.envvars`, y `$LOGNAME` es el ID de usuario de inicio de sesión (en mayúsculas). Si la directiva `user.log` no tiene caracteres de comentario o no está presente, se utiliza la vía de acceso inicial del usuario. La vía de acceso inicial se define en el segmento de seguridad OMVS del ID de usuario. Si la directiva `DSTORE_LOG_DIRECTORY` no tiene caracteres de comentario o no está presente, se añade `.eclipse/RSE/` al valor `user.log`.

- `.dstoreMemLogging` - Anotaciones de utilización de memoria de almacén de datos
- `.dstoreTrace` - Anotaciones de acciones de almacén de datos
- `ffs.log` - Anotaciones del servidor FFS (Foreign File System), 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 o 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 y las anotaciones de comunicación de todos los servicios basados en RSE (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

**Nota:**

- El directorio .eclipse y los archivos de anotaciones .dstore\* empiezan por un punto (.), que hace que queden ocultos. Utilice el mandato de z/OS UNIX **ls -lA** para listar los archivos y directorios ocultos. Al utilizar el cliente Developer for System z, seleccione la página de preferencias **Ventana > Preferencias... > Sistemas remotos > Archivos** y habilite “Mostrar archivos ocultos”.
- La creación de los archivos de anotaciones .dstore\* está controlada por las opciones de inicio **>-DDSTORE\_\*** de Java, como se describe en la sección “Definir parámetros de inicio Java adicionales con \_RSE\_JAVAOPTS” en la página 42.
- Los archivos de anotaciones .dstore\* se crean en ASCII. Utilice el mandato de z/OS UNIX **iconv -f ISO8859-1 -t IBM-1047 .dstore\*** para visualizarlos en EBCDIC (al utilizar la página de códigos IBM-1047).
- No hay archivos de anotaciones stdout.log y stderr.log específicos del usuario cuando la directiva enable.standard.log está activa. Ahora se escriben datos específicos del usuario en la secuencia de agrupación de hebras RSE coincidente.
- Existen mandatos de operador disponibles para controlar la cantidad de datos grabados en algunos de los archivos de anotaciones mencionados. Hallará más información en: Capítulo 8, “Mandatos de operador”, en la página 125.

## Anotaciones de Integración del analizador de errores

- userlog/\$LOGNAME/

Anotaciones de Integración del analizador de errores, donde userlog es el valor combinado de la directivas user.log y DSTORE\_LOG\_DIRECTORY de rsed.envvars, y \$LOGNAME es el ID de usuario de inicio de sesión (en mayúsculas). Si la directiva user.log no tiene caracteres de comentario o no está presente, se utiliza la vía de acceso inicial del usuario. La vía de acceso inicial se define en el segmento de seguridad OMVS del ID de usuario. Si la directiva DSTORE\_LOG\_DIRECTORY no tiene caracteres de comentario o no está presente, se añade .eclipse/RSE/ al valor user.log.

- fa.log - Anotaciones de la Integración de analizador de errores
- rsecomm.log - Anotaciones de comunicación de la Integración de analizador de errores

## Anotaciones de Integración del gestor de archivos

- userlog/\$LOGNAME/rsecomm.log

Anotaciones de comunicación de la Integración del gestor de archivos, donde userlog es el valor combinado de la directivas user.log y DSTORE\_LOG\_DIRECTORY

de `rsed.envvars`, y `$LOGNAME` es el ID de usuario de inicio de sesión (en mayúsculas). Si la directiva `user.log` no tiene caracteres de comentario o no está presente, se utiliza la vía de acceso inicial del usuario. La vía de acceso inicial se define en el segmento de seguridad OMVS del ID de usuario. Si la directiva `DSTORE_LOG_DIRECTORY` no tiene caracteres de comentario o no está presente, se añade `.eclipse/RSE/` al valor `user.log`.

## Anotaciones de SCLM Developer Toolkit

- **userlog/\$LOGNAME/rsecomm.log**

Anotaciones de comunicación de SCLM Developer Toolkit, donde `userlog` es el valor combinado de la directivas `user.log` y `DSTORE_LOG_DIRECTORY` de `rsed.envvars`, y `$LOGNAME` es el ID de usuario de inicio de sesión (en mayúsculas). Si la directiva `user.log` no tiene caracteres de comentario o no está presente, se utiliza la vía de acceso inicial del usuario. La vía de acceso inicial se define en el segmento de seguridad OMVS del ID de usuario. Si la directiva `DSTORE_LOG_DIRECTORY` no tiene caracteres de comentario o no está presente, se añade `.eclipse/RSE/` al valor `user.log`.

## Anotaciones de CARMA

- **Trabajo servidor de CARMA**

Al abrir una conexión con CARMA mediante la interfaz de proceso por lotes, `FEK.#CUST.SYSPROC(CRASUBMT)` 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 el método de inicio de CARMA elegido, las anotaciones de CARMA se redirigen a esta sentencia `DD` en el trabajo servidor; de lo contrario, van a `SYSPRINT`.

- **DD SYSPRINT**

El `SYSPRINT` del trabajo servidor contiene las anotaciones de CARMA, si la sentencia `DD CARMALOG` no está definida.

- **userlog/\$LOGNAME/rsecomm.log**

Anotaciones de comunicación de CARMA, donde `userlog` es el valor combinado de la directivas `user.log` y `DSTORE_LOG_DIRECTORY` de `rsed.envvars`, y `$LOGNAME` es el ID de usuario de inicio de sesión (en mayúsculas). Si la directiva `user.log` no tiene caracteres de comentario o no está presente, se utiliza la vía de acceso inicial del usuario. La vía de acceso inicial se define en el segmento de seguridad OMVS del ID de usuario. Si la directiva `DSTORE_LOG_DIRECTORY` no tiene caracteres de comentario o no está presente, se añade `.eclipse/RSE/` al valor `user.log`.

## Anotaciones de 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 `ATBSDFMU`. El valor predeterminado en el JCL de

ejemplo FEK.SFEKSAMP(FEKAPPC) es SYSOUT=\*. Consulte la publicación *MVS Planning: APPC/MVS Management* (SA22-7599) para 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 FEK.SFEKSAMP(FEKAPPC) es &SYSUID.FEKFRSRV.&TPDATE.&TPTIME.LOG. Consulte la publicación *MVS Planning: APPC/MVS Management* (SA22-7599) para 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. Consulte la publicación *MVS Planning: APPC/MVS Management* (SA22-7599) para obtener más información.

## Anotaciones de prueba IVP de fekfivpi

- **userlog/\$LOGNAME/fekfivpi.log**

Salida del mandato fekfivpi -file (Prueba IVP relacionada con la Pasarela de cliente TSO/ISPF), donde userlog es el valor combinado de la directivas user.log y DSTORE\_LOG\_DIRECTORY de rsed.envvars, y \$LOGNAME es el ID de usuario de inicio de sesión (en mayúsculas). Si la directiva user.log no tiene caracteres de comentario o no está presente, se utiliza la vía de acceso inicial del usuario. La vía de acceso inicial se define en el segmento de seguridad OMVS del ID de usuario. Si la directiva DSTORE\_LOG\_DIRECTORY no tiene caracteres de comentario o no está presente, se añade .eclipse/RSE/ al valor user.log.

## Anotaciones de prueba IVP de fekfivps

- **userlog/\$LOGNAME/fekfivps.log**

Salida del mandato fekfivps -file (Prueba IVP relacionada con SCLMDT), donde userlog es el valor combinado de la directivas user.log y DSTORE\_LOG\_DIRECTORY de rsed.envvars, y \$LOGNAME es el ID de usuario de inicio de sesión (en mayúsculas). Si la directiva user.log no tiene caracteres de comentario o no está presente, se utiliza la vía de acceso inicial del usuario. La vía de acceso inicial se define en el segmento de seguridad OMVS del ID de usuario. Si la directiva DSTORE\_LOG\_DIRECTORY no tiene caracteres de comentario o no está presente, se añade .eclipse/RSE/ al valor user.log.

---

## 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 las publicaciones *MVS JCL Reference* (SA22-7597) y *Language Environment Debugging Guide* (GA22-7560) para obtener más información acerca de estas sentencias DD.

## Vuelcos de Java

En z/OS UNIX, la mayoría de vuelcos de Developer for 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 los siguientes:

### 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 `%uid.JVM.TDUMP.%job.D%y%m%d.T%H%M%S`, 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`.

**Nota:** `JAVA_DUMP_TDUMP_PATTERN` permite el uso de variables, que se convierten en un valor real en el momento del volcado de la transacción.

Tabla 20. Variables de `JAVA_DUMP_TDUMP_PATTERN`

Variable	Uso
<code>%uid</code>	ID de usuario
<code>%job</code>	Nombre de trabajo
<code>%y</code>	Año (2 dígitos)
<code>%m</code>	Mes (2 dígitos)
<code>%d</code>	Día (2 dígitos)
<code>%H</code>	Hora (2 dígitos)
<code>%M</code>	Minuto (2 dígitos)
<code>%S</code>	Segundo (2 dígitos)

### 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: “Ubicaciones de vuelcos de z/OS UNIX”.

### 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: “Ubicaciones de vuelcos de z/OS UNIX”.

### 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: “Ubicaciones de vuelcos de z/OS UNIX”.

Consulte las publicaciones *Java Diagnostic Guide* (SC34-6358), para obtener más información acerca de los vuelcos de JVM, y *Language Environment Debugging Guide* (GA22-7560) para obtener información específica acerca de LE.

## Ubicaciones de vuelcos de 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 el valor `/tmp` en el archivo `rsed.envvars`. Se puede cambiar a `/dev/null` para no tener que crear los archivos de vuelco.
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.

---

## Rastreo

### Rastreo del Supervisor de trabajos JES

El rastreo del Supervisor de trabajos JES está controlado por el operador del sistema, como se describe en la sección Capítulo 8, “Mandatos de operador”, en la página 125.

- Al iniciar la tarea iniciada JMON con el parámetro PRM=-TV, se activa la modalidad verbosa (rastreo)
- Los mandatos **modify -TV** y **modify -TN** activan y desactivan el rastreo.

### Rastreo de RSE

Los componentes relacionados con RSE crean varios archivos de anotaciones. La mayoría están ubicados en `userlog/$LOGNAME/`, donde `userlog` es el valor combinado de la directivas `user.log` y `DSTORE_LOG_DIRECTORY` de `rsed.envvars`, y `$LOGNAME` es el ID de usuario de inicio de sesión (en mayúsculas). Si la directiva `user.log` no tiene caracteres de comentario o no está presente, se utiliza la vía de acceso inicial del usuario. La vía de acceso inicial se define en el segmento de seguridad OMVS del ID de usuario. Si la directiva `DSTORE_LOG_DIRECTORY` no tiene caracteres de comentario o no está presente, se añade `.eclipse/RSE/` al valor `user.log`.

La cantidad de datos escritos en `ffs*.log`, `lock.log` y `rsecomm.log` se controla mediante el mandato del operador **modify rsecommlog** o estableciendo `debug_level` en `rsecomm.properties`. Consulte las secciones Capítulo 8, “Mandatos de operador”, en la página 125 y “(Opcional) Rastreo de RSE” en la página 96 para obtener más detalles.

La creación de los archivos de anotaciones `.dstore*` está controlada por las opciones de inicio `-DDSTORE_*` de Java, como se describe en la sección “Definir parámetros de inicio Java adicionales con `_RSE_JVAOPTS`” en la página 42.

#### Nota:

- El directorio `.eclipse` y los archivos de anotaciones `.dstore*` empiezan por un punto (`.`), que hace que queden ocultos. Utilice el mandato de `z/OS UNIX` **ls -lA** para listar los archivos y directorios ocultos. Al utilizar el cliente Developer for System z, seleccione la página de preferencias **Ventana > Preferencias... > Sistemas remotos > Archivos** y habilite “Mostrar archivos ocultos”.
- Los archivos de anotaciones `.dstore*` se crean en ASCII. Utilice el mandato de `z/OS UNIX` **iconv -f ISO8859-1 -t IBM-1047 .dstore\*** para visualizarlos en EBCDIC (al utilizar la página de códigos IBM-1047).

Los archivos de anotaciones específicos del daemon RSE y de la agrupación de hebras RSE se encuentran en `daemon-home`, donde `daemon-home` es el valor de la directiva `daemon.log` de `rsed.envvars`. Si la directiva `daemon.log` no tiene caracteres de comentario o no está presente, se utiliza el directorio inicial del ID de usuario asignado a la tarea iniciada RSED. El directorio inicial se define en el segmento de seguridad OMVS del ID de usuario.

La cantidad de datos escritos en `rsedaemon.log` y `rserver.log` se controla mediante los mandatos del operador **modify rsedaemonlog** y **modify rserverlog** o estableciendo `debug_level` en `rsecomm.properties`. Consulte las secciones

Capítulo 8, “Mandatos de operador”, en la página 125 y “(Opcional) Rastreo de RSE” en la página 96 para obtener más detalles.

`serverlogs.count`, `stderr.*.log` y `stdout.*.log` solamente se crean si la directiva `enable.standard.log` de `rsed.envvars` está activa, o si la función está activada dinámicamente con el mandato de operador **`modify rsestandardlog on`**.

## Rastreo del daemon de bloqueo

Las anotaciones específicas del daemon de bloqueo están ubicadas en `STDOUT DD` de la tarea iniciada `LOCKD`. La cantidad de datos escritos en las anotaciones se controla mediante el parámetro de inicio `LOG`. Consulte las secciones Capítulo 8, “Mandatos de operador”, en la página 125 y “(Opcional) Rastreo de RSE” en la página 96 para obtener más detalles.

## Rastreo de CARMA

El usuario puede controlar la cantidad de información de rastreo generada por CARMA, estableciendo el Nivel de rastreo en la pestaña de 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 el siguiente:

Anotaciones de error

Para obtener más información sobre la ubicación de los archivos de anotaciones, consulte: “Archivos de anotaciones” en la página 138.

## Rastreo de información de retorno de errores

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 for 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: “Procedimientos de construcción remota `ELAXF*`” en la página 25.
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 el programa COBOL del que desea obtener el rastreo detallado.
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.SF.Z682746.XML  
23 //COBOL.WSEDSF2 DD DISP=MOD,  
    // DSN=uid.ERRCOB.member.SF.Z682747.XML
```

5. Copie estos cuatro conjuntos de datos en el PC, creando por ejemplo un proyecto COBOL local en Developer for 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 for System z y guardándola en el proyecto local, seleccionado **Archivo > Guardar como...**
7. Restaure el procedimiento ELAXFC0C 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.

---

## Bits de permiso de z/OS UNIX

Developer for System z requiere que el sistema de archivos de z/OS UNIX y algunos archivos de z/OS UNIX tengan establecidos determinados bits de permiso.

### Atributo del sistema de archivos SETUID

El Explorador de sistemas remotos (RSE) es el componente de Developer for System z que proporciona servicios centrales como los de conectar el cliente al host. Debe permitírsele realizar tareas tales como crear el entorno de seguridad del usuario.

El sistema de archivos (HFS o zFS) en el que se instala Developer for System z debe estar montado con el bit de permiso SETUID (este el valor predeterminado del sistema). El hecho de montar el sistema de archivos con el parámetro NOSETUID impedirá que Developer for System z cree el entorno de seguridad del usuario y la solicitud de conexión fallará.

Utilice el mandato **ISHELL** de TSO para listar el estado actual del bit SETUID. En el panel de ISHELL, seleccione **Sistemas de archivos > 1. Tabla de montaje...** para

listar los sistemas de archivos montados. El mandato abreviado **a** mostrará los atributos del sistema de archivos seleccionado, donde el campo “Ignorar SETUID” debe ser 0.

## Autorización de control de programa

El Explorador de sistemas remotos (RSE) es el componente de Developer for 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 de z/OS UNIX se establece durante la instalación de SMP/E cuando es necesario, excepto para la interfaz de Java del producto de seguridad, tal como se documenta en la sección Capítulo 10, “Consideraciones relativas a la seguridad”, en la página 159. Este bit de permiso puede perderse si no lo ha conservado durante una copia manual de los directorios de Developer for System z.

Los siguientes archivos de Developer for System z deben estar controlados por programa:

- /usr/lpp/rdz/bin/
  - fekfdivp
  - fekfomvs
  - fekfrivp
- /usr/lpp/rdz/lib/
  - fekfdir.dll
  - libfekdcore.so
  - libfekfmain.so
- /usr/lpp/rdz/lib/icuc/
  - libicudata.dll
  - libicudata40.1.dll
  - libicudata40.dll
  - libicudata64.40.1.dll
  - libicudata64.40.dll
  - libicudata64.dll
  - libicuuc.dll
  - libicuuc40.1.dll
  - libicuuc40.dll
  - libicuuc64.40.1.dll
  - libicuuc64.40.dll
  - libicuuc64.dll

**Nota:** Los archivos libicu\*64.\* sólo están presentes si aplicó el PTF de Developer for System z que trata el APAR AM07305 para mejorar el soporte de 64 bits.

Utilice el mandato de z/OS UNIX **ls -E** para listar los atributos ampliados, en los que el bit de control de programa está marcado con la letra **p**, tal como se muestra en el siguiente ejemplo (\$ es la solicitud de z/OS UNIX):

```
$ cd /usr/lpp/rdz
$ ls -E lib/fekf*
-rwxr-xr-x -ps- 2 user      group      94208 Jul  8 12:31 lib/fekfdir.dll
```

Utilice el mandato de z/OS UNIX **extattr +p** para establecer el bit de control de programa manualmente, como se muestra en el ejemplo siguiente (\$ y # son la solicitud de z/OS UNIX):

```
$ cd /usr/lpp/rdz
$ su
# extattr +p lib/fekf*
# exit
$ ls -E lib/fekf*
-rwxr-xr-x  -ps-  2 user      group      94208 Jul  8 12:31 lib/fekfdir.dll
```

**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, o ser un superusuario (UID 0) si este perfil no está definido. Para obtener más información, consulte la publicación *UNIX System Services Planning* (GA22-7800).

## Autorización de APF

El Explorador de sistemas remotos (RSE) es el componente de Developer for System z que proporciona servicios centrales como los de conectar el cliente al host. Se debe ejecutar con autorización de APF para poder realizar tareas como por ejemplo visualizar la utilización de recursos de proceso detallados.

El bit de z/OS UNIX APF se establece durante la instalación de SMP/E, cuando sea necesario. Este bit de permiso puede perderse si no lo ha conservado durante una copia manual de los directorios de Developer for System z.

Los siguientes archivos de Developer for System z deben estar autorizados por APF:

- /usr/lpp/rdz/bin/
  - fekfomvs
  - fekfriwp

Utilice el mandato z/OS UNIX **ls -E** para obtener una lista de los atributos ampliados, en la que el bit de APF está marcado con la letra a, tal como se muestra en el ejemplo siguiente (\$ es la solicitud de z/OS UNIX):

```
$ cd /usr/lpp/rdz
$ ls -E bin/fekfriwp
-rwxr-xr-x  aps-  2 user      group      114688 Sep 17 06:41 bin/fekfriwp
```

Utilice el mandato z/OS UNIX **extattr +a** para establecer manualmente el bit APF, tal como se muestra en el ejemplo siguiente (\$ y # son las solicitudes de z/OS UNIX):

```
$ cd /usr/lpp/rdz
$ su
# extattr +a bin/fekfriwp
# exit
$ ls -E bin/fekfriwp
-rwxr-xr-x  aps-  2 user      group      114688 Sep 17 06:41 bin/fekfriwp
```

**Nota:** Para poder utilizar el mandato **extattr +a**, debe tener como mínimo acceso de lectura (READ) al perfil BPX.FILEATTR.APF en la clase FACILITY del software de seguridad, o ser un superusuario (UID 0) si este perfil no está definido. Para obtener más información, consulte *UNIX System Services Planning* (GA22-7800).

## Sticky bit

Algunos de los servicios opcionales de Developer for System z requieren que los módulos de carga de MVS estén disponibles para z/OS UNIX. Esta operación se realiza creando un apéndice (un archivo ficticio) en z/OS UNIX con el "sticky" bit activado. Al ejecutar el apéndice, z/OS UNIX buscará un módulo de carga de MVS con el mismo nombre y ejecutará el módulo de carga en su lugar.

El sticky bit de z/OS UNIX se establece durante la instalación de SMP/E, cuando es necesario. Estos bits de permiso pueden perderse si no los ha conservado durante una copia manual de los directorios de Developer for System z.

Los siguientes archivos de Developer for System z deben tener activado el sticky bit:

- /usr/lpp/rdz/bin/
  - BWBTSOW
  - CRASTART

Utilice el mandato de z/OS UNIX **ls -l** para listar los permisos, en los que el sticky bit está marcado con la letra **t**, como se muestra en el ejemplo siguiente (\$ es la solicitud de z/OS UNIX):

```
$ cd /usr/lpp/rdz
$ ls -l bin/CRA*
-rwxr-xr-t  2 user      group          71 Jul  8 12:31 bin/CRASTART
```

Utilice el mandato de z/OS UNIX **chmod +t** para establecer el sticky bit manualmente, como se muestra en el ejemplo siguiente (\$ y # son la solicitud de z/OS UNIX):

```
$ cd /usr/lpp/rdz
$ su
# chmod +t bin/CRA*
# exit
$ ls -l bin/CRA*
-rwxr-xr-t  2 user      group          71 Jul  8 12:31 bin/CRASTART
```

**Nota:** Para poder utilizar el mandato **chmod**, debe tener como mínimo acceso de lectura (READ) al perfil SUPERUSER.FILESYS.CHANGPERMS en la clase UNIXPRIV del software de seguridad, o ser un superusuario (UID 0) si este perfil no está definido. Para obtener más información, consulte la publicación *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 del ejemplo que sigue. Los puertos utilizados son el último número (a continuación de "..") de la columna "Socket Local". Como estos puertos ya se están utilizando, no se pueden utilizar para la configuración de Developer for System z.

### IPv4

MVS TCP/IP	NETSTAT CS VxRy	TCP/IP Nombre	TCP/IP: TCP/IP	16:36:42
ID us.	Conexión	Socket Local	Socket Foráneo	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
RSED	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 TCP/IP NETSTAT CS VxRy	Nombre TCPIP: TCPIP	12:46:25
ID usuario Conexión Estado		
-----		
BPXOINIT 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		
RSED 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.

Consulte la publicación *Communications Server: IP Configuration Guide* (SC31-8775) para obtener más información acerca de estas sentencias.

### • 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 (varias pilas TCP/IP activas en un único host).
- 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 (varias pilas TCP/IP activas en un único host).

Consulte las publicaciones *UNIX System Services Planning* (GA22-7800) y *MVS Initialization and Tuning Reference* (SA22-7592) para obtener más información sobre estas sentencias.

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 del ejemplo siguiente:

MVS TCP/IP NETSTAT CS VxRy	Nombre TCPIP: TCPIP	17:08:32
NºPto Prot Usuario Dstivos	Rango Dirección IP	
-----	-----	
00007 TCP MISC SERV DA		
00009 TCP MISC SERV DA		
00019 TCP MISC SERV 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

Consulte la publicación *Communications Server: IP System Administrator's Commands* (SC31-8781) para obtener más información acerca del 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.

---

## Tamaño del espacio de direcciones

Para el daemon RSE, que es un proceso z/OS UNIX Java 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 de JCL de inicio

El daemon RSE se inicia mediante JCL utilizando **BPXBATSL**, cuyo tamaño de región debe ser 0.

### 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, en 2G. Es el tamaño máximo permitido. Este es un límite a escala del sistema y, por ello, está activo para todos los espacios de direcciones z/OS UNIX. Si no desea este límite, puede establecer el límite únicamente para Developer for System z en el software de seguridad.

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* (GC28-1781):

1. **DISPLAY OMVS,0**
2. **SETOMVS MAXASSIZE=2G**

### Limitaciones almacenadas en el perfil de seguridad

Compruebe **ASSIZEMAX**, en el segmento OMVS del ID de usuario del daemon, y establézcalo en 2147483647 o, preferiblemente, en **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 aplicadas 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* (GC28-1781):

1. DISPLAY SMF,0
2. SET SMF=xx

## Limitaciones para el direccionamiento de 64 bits

La palabra clave MEMLIMIT en SYS1.PARMLIB(SMFPRMxx) limita el almacenamiento virtual que una tarea de 64 bits puede asignar por encima de la barra de 2GB. Al contrario que el parámetro REGION en JCL, MEMLIMIT=0M significa que el proceso no puede utilizar almacenamiento virtual por encima de la barra.

Si no se especifica MEMLIMIT en SMFPRMxx, el valor predeterminado es 0M, con lo que las tareas están limitadas a los 2GB (31 bits) por debajo de la barra. El valor predeterminado cambió en z/OS 1.10 a 2G, permitiendo que las tareas de 64 bits utilicen hasta 4GB (los 2GB por debajo de la barra y los 2GB por encima de la barra otorgados por MEMLIMIT).

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* (GC28-1781):

1. DISPLAY SMF,0
2. SET SMF=xx

MEMLIMIT también se puede especificar como parámetro en una tarjeta EXEC en JCL. Si no se especifica ningún parámetro MEMLIMIT, el valor predeterminado es el valor definido por SMF, excepto cuando se especifica REGION=0M, en cuyo caso el valor predeterminado es NOLIMIT.

---

## 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 habituales de este tipo de problemas son las siguientes:
  - No posee autorización de lectura sobre el conjunto de datos FEK.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 las siguientes:
  - 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 FEK.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 suministrado en "(Opcional) Transacción APPC para el servicio de mandatos TSO" en la página 104 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 APPC 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/rdz/support/>). Si desea información adicional, consulte 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 los mensajes siguientes:
  - 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:** Esta lista no es definitiva. Consulte el sitio web de soporte para acceder a más notas técnicas.

---

## 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 for System z activos. La mayoría de los valores BPXPRMxx se pueden cambiar dinámicamente con los mandatos de consola **SETOMVS** y **SET OMVS**.

Utilice el mandato de consola **SETOMVS LIMMSG=ALL** para que z/OS UNIX muestre los mensajes de consola (BPXI040I) cuando se está a punto de alcanzar alguno de los límites de BPXPRMxx.

### 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 IDs 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 de los requisitos

### La apertura de los conjuntos de datos de MVS falla

Al utilizar APPC para el servicio de mandatos TSO, 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) ENTRYPOINT(BPXTUINT)
NETWORK DOMAINNAME(AF_UNIX)
        DOMAINNUMBER(1)
        MAXSOCKETS(2000)
        TYPE(UDS)

```

Otra causa probable de este problema, al utilizar APPC para el servicio de mandatos TSO, es que el Resolvente de TCP/IP no pueda resolver la dirección de host correctamente debido a un archivo de configuración de resolvente faltante o incompleto. Una indicación clara de este problema es el mensaje siguiente en lock.log:

```
clientip(0.0.0.0) <> callerip(<dirección IP host>)
```

Ejecute el IVP de TCP/IP fekfivpt, como se describe en Capítulo 7, “Verificación de la instalación”, en la página 109. La sección de configuración del resolvente de la salida será como la del ejemplo siguiente:

```
Inicialización de rastreo de resolvente completada -> 2008/07/02 13:11:54.745964
```

```

Valores de resolvente res_init:
Conjunto de datos Tcp/Ip global           = Ninguno
Conjunto de datos Tcp/Ip predeterminado   = Ninguno
Conjunto de datos Tcp/Ip local            = /etc/resolv.conf
Tabla de conversión                      = Predeterminada
IDusuario/NombreTrabajo                  = USERID
API llamante                             = LE C Sockets
Modalidad llamante                       = EBCDIC

```

Asegúrese de que las definiciones del archivo (o del conjunto de datos) a las que hace referencia “Conjunto de datos Tcp/Ip local” sean correctas.

Este campo estará en blanco si no utiliza un nombre predeterminado para el archivo resolvente de IP (utilizando el orden de búsqueda de z/OS UNIX). Si es así, añada la sentencia siguiente al archivo rsed.envvars, donde <archivo resolvente> o <datos resolventes> representa el nombre del archivo resolvente de IP:

```
RESOLVER_CONFIG=<archivo resolvente>
```

o bien

```
RESOLVER_CONFIG='<conjunto de datos resolventes>'
```

## Emulador de conexión de host

- 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 conocida, cada cliente debe añadir el certificado de la CA a su lista de CA de confianza del emulador de conexión de host.
- 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.



---

## Capítulo 10. Consideraciones relativas a la seguridad

Developer for System z proporciona a los usuarios acceso al sistema central en una estación de trabajo que no es del sistema central. Algunos aspectos importantes de la configuración del producto son: validar las solicitudes de conexión, proporcionar una comunicación segura entre el host y la estación de trabajo, y autorizar y auditar la actividad.

Los mecanismos de seguridad utilizados por los servidores y servicios de Developer for System z se basan en que el sistema de archivos en el que residen sea seguro. Esto implica que sólo los administradores del sistema que sean de confianza puedan actualizar las bibliotecas de programa y los archivos de configuración.

En este capítulo se tratan estos temas:

- “Métodos de autenticación”
- “Seguridad de conexión” en la página 161
- “Puertos TCP/IP” en la página 162
- “Utilizar PassTickets” en la página 164
- “Anotaciones de auditoría” en la página 165
- “Seguridad de JES” en la página 166
- “Comunicación cifrada con SSL” en la página 170
- “Autenticación de cliente mediante certificados X.509” en la página 171
- “Comprobación de puerto de entrada (POE)” en la página 174
- “Seguridad de CICSTS” en la página 175
- “Seguridad de SCLM” en la página 175
- “Archivos de configuración de Developer for System z” en la página 176
- “Definiciones de seguridad” en la página 177

**Nota:** El Explorador de sistemas remotos (RSE), que proporciona servicios del núcleo como los de conectar el cliente al host, está formado por 2 entidades lógicas.

- El daemon RSE, que gestiona la configuración de conexiones y se inicia como tarea iniciada o como trabajo de usuario de larga ejecución.
- El servidor RSE, que maneja las solicitudes de clientes individuales y se inicia como una hebra en uno o varios procesos hijo del daemon RSE.

Consulte Capítulo 11, “Qué es Developer for System z”, en la página 191 para conocer los conceptos de diseño básicos de Developer for System z.

---

### Métodos de autenticación

Developer for System z admite varias formas de autenticar un ID de usuario facilitado por un cliente durante la conexión.

- ID de usuario y contraseña
- ID de usuario y contraseña para una sola vez
- certificado X.509

Tenga en cuenta que los datos de autenticación facilitados por el cliente solamente se utilizan una vez, durante la configuración inicial de la conexión. Una vez se autentica un ID de usuario, este y los PassTickets autogenerados se utilizan para todas las acciones que requieren autenticación.

## **ID de usuario y contraseña**

El cliente facilita un ID de usuario y una contraseña coincidente durante la conexión. El ID de usuario y la contraseña se utilizan para autenticar al usuario en el producto de seguridad.

## **ID de usuario y contraseña para una sola vez**

Basándose en una única señal, un producto externo puede generar una contraseña para una sola vez. Las contraseñas para una sola vez mejoran la configuración de seguridad, ya que la señal exclusiva no se puede copiar y utilizar sin el conocimiento del usuario, y una contraseña interceptada no sirve para nada porque solamente es válida una vez.

El cliente facilita un ID de usuario y una contraseña para una sola vez durante la conexión que se utiliza para autenticar el ID de usuario con la salida de seguridad proporcionada por un programa externo. Se espera que esta salida de seguridad ignore los PassTickets utilizados para satisfacer las solicitudes de autenticación durante el proceso normal. Su software de seguridad debe procesar los PassTickets.

## **Certificado X.509**

Un tercer puede proporcionar uno o varios certificados X.509 que se pueden utilizar en la autenticación de un usuario. Cuando están almacenados en dispositivos seguros, los certificados X.509 combinan una configuración segura con un uso sencillo para el usuario (no son necesarios ni ID de usuario ni contraseña).

Durante la conexión, el cliente facilita un certificado seleccionado y, opcionalmente, una extensión seleccionada, que se utiliza para autenticar el ID de usuario con su producto de seguridad.

Tenga en cuenta que este método de autenticación solamente está soportado por el método de conexión del daemon RSE y que la SSL debe estar habilitada.

## **Autenticación del Supervisor de trabajos JES**

El daemon RSE (o REXEC/SSH) realiza la autenticación de clientes como parte de la solicitud de conexión del cliente. Una vez se autentica el usuario, se utilizan PassTickets autogenerados para las solicitudes de autenticación que se realicen en el futuro, incluido el inicio de sesión automático en el Supervisor de trabajos JES.

Para que el Supervisor de trabajos JES valide el ID de usuario y el PassTicket presentado por RSE, el Supervisor de trabajos JES debe poder evaluar el PassTicket. Ello implica:

- El módulo de carga FEJMON, ubicado, por omisión, en la biblioteca de carga FEK.SFEKAUTH, debe estar autorizado APF.
- Tanto RSE como el Supervisor de trabajos JES deben utilizar el mismo ID de aplicación (APPLID). Por omisión, ambos servidores utilizan FEKAPPL como APPLID, pero esto puede verse modificado por la directiva APPLID de rsed.envvars para RSE y de FEJCNFG para el Supervisor de trabajos JES.

**Nota:** Los clientes anteriores (versión 7.0 y anteriores) se comunican directamente con el supervisor de trabajos JES. Para estas conexiones, solamente está soportado el método de autenticación de ID de usuario y contraseña.

---

## Seguridad de conexión

El servidor RSE, que controla toda la comunicación entre el cliente y los servicios de Developer for System z, da soporte a varios niveles de seguridad de comunicaciones:

- La comunicación externa (cliente-host) puede limitarse a puertos especificados. Esta característica está inhabilita por omisión.
- La comunicación externa (cliente-host) puede cifrarse mediante SSL. Esta característica está inhabilita por omisión.
- Puede utilizarse la comprobación de puerto de entrada (POE) para permitir el acceso de host sólo a las direcciones TCP/IP de confianza. Esta característica está inhabilita por omisión.

### Limitar la comunicación externa a puertos especificados

El programador del sistema puede especificar los puertos en los que el servidor RSE se puede comunicar con el cliente. Por omisión, se utiliza cualquier puerto disponible. Este rango de puertos no tiene conexión con el puerto 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 del host 4035, el daemon RSE.
2. El puerto del daemon RSE crea una hebra de servidor RSE.
3. El servidor 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 daemon RSE devuelve el número de puerto al cliente.
5. El cliente se conecta al puerto del host.

**Nota:** El proceso es similar para el método de conexión alternativo (opcional) mediante REXEC/SSH, que se describe en “(Opcional) Utilizar REXEC (o SSH)” en la página 102.

### Cifrado de comunicaciones mediante SSL

Todas las corrientes de datos externas de Developer for System z que pasan a través de RSE pueden cifrarse mediante SSL (Capa de sockets seguros). La utilización de SSL está controlada por los valores del archivo de configuración `ssl.properties`, como se describe en la sección “Comunicación cifrada con SSL” en la página 170.

El Emulador de conexión de host del cliente se conecta a un servidor TN3270 del host. La utilización de SSL está controlada por TN3270, como se describe en la publicación *Communications Server IP Configuration Guide* (SC31-8775).

El cliente Gestor de despliegue de aplicaciones utiliza el servicio Web de CICS TS de la interfaz RESTful para invocar los servicios de host del Gestor de despliegue de aplicaciones. La utilización de SSL está controlada por CICS TS, tal como se describe en la publicación *RACF Security Guide for CICS TS*.

## Comprobación de Puerto de entrada

Developer for System z da soporte a la comprobación de puerto de entrada (POE), que permite el acceso de host sólo a las direcciones TCP/IP de confianza. La utilización de POE está controlada por la definición de perfiles específicos del software de seguridad y por la directiva `enable.port.of.entry` del archivo `rsed.envvars`, como se describe en la sección “Comprobación de puerto de entrada (POE)” en la página 174.

Tenga en cuenta que la activación de POE influirá sobre otras aplicaciones TCP/IP que den soporte a la comprobación de POE, como INETD.

---

## Puertos TCP/IP

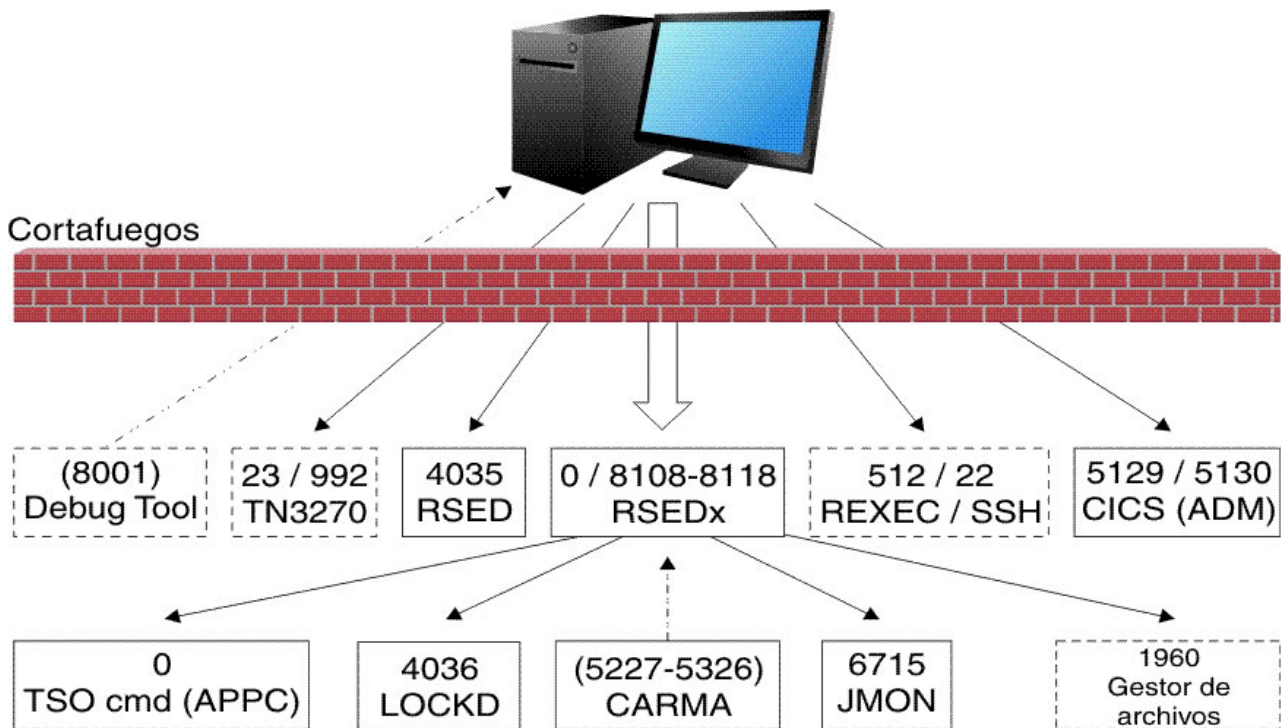


Figura 40. Puertos TCP/IP

Figura 40 muestra los puertos TCP/IP que Developer for System z puede utilizar. Las flechas muestran qué parte realiza el enlace (parte de la punta de la flecha) y qué parte realiza la conexión.

## Comunicación externa

Defina los puertos siguientes para el cortafuegos que protege el host z/OS, ya que se utilizan para la comunicación cliente-host (utilizar el protocolo tcp):

- Daemon RSE para la configuración de la comunicación cliente-host, puerto predeterminado 4035. La comunicación en este puerto puede estar cifrada mediante SSL.
- Servidor RSE para la comunicación entre cliente y host. Por omisión, se utiliza cualquier puerto disponible, pero puede limitarse a un rango especificado con la definición de `_RSE_PORTRANGE` de `rsed.envvars`. La comunicación en este puerto puede estar cifrada mediante SSL.

- (opcional) Servicio INETD para acciones remotas (basadas en host) en subproyectos z/OS UNIX:
  - REXEC (versión z/OS UNIX), puerto predeterminado 512.
  - SSH (versión z/OS UNIX), puerto predeterminado 22. La comunicación en este puerto está cifrada mediante SSL.
- (opcional) Servicio Telnet TN3270 para el Emulador de conexión de host, puerto predeterminado 23. La comunicación 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.
- (opcional) Una de las dos o ambas interfaces de aplicación CICSTS para el Gestor de despliegue de aplicaciones:
  - Interfaz RESTful, puerto predeterminado 5130.
  - Interfaz de servicios Web, puerto predeterminado 5129. La comunicación en este puerto puede estar cifrada mediante SSL.

**Nota:**

- Los clientes anteriores (versión 7.0 y anteriores) se comunican directamente con el Supervisor de trabajos JES, puerto predeterminado 6715.
- 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.

## Comunicación interna

Varios servicios de host de Developer for System z se ejecutan en hebras o espacios de direcciones separados y utilizan sockets TCP/IP como mecanismo de comunicación. Todos estos servicios utilizan RSE para comunicarse con el cliente, confinando con ello su corriente de datos solamente al host. Para algunos servicios se utilizará cualquier puerto disponible, mientras que para otros el programador del sistema puede elegir el puerto o rango de puertos que se utilizará:

- Supervisor de trabajos JES para servicios relacionados con JES, puerto predeterminado 6715. El puerto puede establecerse en el miembro de configuración FEJJCNFG.
- Bloquear daemon para servicios relacionados con bloqueo de conjunto de datos, puerto predeterminado 4036. El puerto se puede establecer en el miembro de configuración rsed.envvars.
- (opcional) Integración del gestor de archivos para interactuar con IBM File Manager, puerto predeterminado 1960.
- (opcional) Comunicación de CARMA, rango de puertos predeterminado 5227-5326 (100 puertos). El rango de puertos puede establecerse en el archivo de configuración CRASRV.properties.
- (opcional) Las versión APPC del servicio de mandatos TSO utiliza cualquier socket disponible para comunicarse con el gestor de bloqueos (que coloca en cola conjuntos de datos de MVS para los clientes). No puede establecerse un rango de puertos específico.

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

## Puertos CARMA y TCP/IP

En la mayoría de los casos, como con el daemon RSE, un servidor se enlaza a un puerto y escucha las solicitudes de conexión. Sin embargo, CARMA utiliza otro procedimiento, dado que el servidor CARMA no está activo cuando el cliente inicia la solicitud de conexión.

Cuando el cliente envía una solicitud de conexión, el miner de CARMA, que está activo como hebra de usuario en una agrupación de hebras RSE, buscará un puerto libre dentro del rango especificado en el archivo de configuración `CRASRV.properties` y se enlaza a dicho puerto. El miner inicia entonces el servidor CARMA y pasa el número de puerto, de manera que el puerto sepa a qué puerto conectarse. Una vez el servidor está conectado, el cliente puede mandar solicitudes al servidor y recibir los resultados.

Así, desde una perspectiva de TCP/IP, RSE (a través del miner de CARMA) es el servidor que se enlaza al puerto, y el servidor CARMA es el cliente que se conecta.

---

## Utilizar PassTickets

Después de del inicio de sesión, se utilizan Pases (PassTickets) para establecer la seguridad de las hebras dentro del servidor RSE. Esta característica no puede inhabilitarse. Los PassTickets son contraseñas generadas por el sistema con un tiempo de vida aproximado de 10 minutos. Los PassTickets generados se basan en el algoritmo de cifrado DES, en el ID de usuario, en el ID de aplicación, en la indicación de fecha y hora, y en una clave secreta. Esta clave secreta es un número de 64 bits (16 caracteres hexadecimales) que deben definirse en el software de seguridad.

Para ayudarle a comprender la utilización de PassTicket, se proporciona esta breve descripción del proceso de seguridad del RSE:

1. El cliente se conecta al puerto del host 4035, el daemon RSE.
2. El daemon RSE autentica el cliente mediante las credenciales presentadas por el éste.
3. El daemon RSE crea un ID de cliente exclusivo y una hebra de servidor RSE.
4. El servidor RSE genera un PassTicket y crea un entorno de seguridad para el cliente utilizando el PassTicket como contraseña.
5. El cliente se conecta al puerto de host devuelto por el daemon RSE.
6. El servidor RSE valida el cliente utilizando el ID de éste.
7. El servidor RSE utiliza un PassTicket generado como contraseña para todas las acciones futuras que la requieran.

La contraseña real del cliente ya no es necesaria después de la autenticación inicial porque los productos de seguridad compatibles con SAF pueden evaluar tanto los PassTickets como las contraseñas habituales. El servidor RSE genera y utiliza un PassTicket cada vez que es necesaria una contraseña, cuyo resultado es una contraseña válida (temporal) para el cliente.

El uso de PassTickets permite a RSE configurar un entorno de seguridad específico del usuario a voluntad, sin necesidad de almacenar todos los ID de usuario y las contraseñas en una tabla, cosa que podría poner en peligro esta información. También lo permite para métodos de autenticación de cliente que no utilizan contraseñas reutilizables, como los certificados X.509.

Los perfiles de seguridad de las clases de APPL y PTKTDATA son necesarios para poder utilizar los PassTickets. Estos perfiles son específicos de la aplicación y, por ello, no afectan a la configuración de su sistema actual.

El hecho de que los PassTickets sean específicos de la aplicación implica que tanto RSE como el Supervisor de trabajos JES deben utilizar el mismo ID de aplicación (APPLID). Por omisión, ambos servidores utilizan FEKAPPL como APPLID, pero esto puede verse modificado por la directiva APPLID de `rsed.envvars` para RSE y de `FEJJCNFG` para el Supervisor de trabajos JES.

Ahora no debe utilizar OMVSAPPL como ID de aplicación ya que abrirá la clave secreta a la mayoría de las aplicaciones z/OS UNIX. Tampoco debe utilizar el ID de aplicación MVS predeterminado, que es MVS seguido por el ID SMF del sistema, porque esto abrirá la clave secreta a la mayoría de aplicaciones MVS (incluyendo trabajos por lotes de usuarios).

**Atención:** La solicitud de conexión del cliente fallará si los PassTickets no están configurados correctamente.

---

## Anotaciones de auditoría

Developer for System z da soporte a las anotaciones de auditoría de acciones gestionadas por el daemon RSE. Las anotaciones de auditoría se almacenan como archivos de texto en el directorio de anotaciones del daemon, utilizando el formato CSV (valores separados por comas).

### Control de auditoría

Varias opciones de `rsed.envvars` influyen sobre la función de auditoría, como se describe en la sección “Definir parámetros de inicio Java adicionales con `_RSE_JAVAOPTS`” en la página 42.

- La función de auditoría se habilita/inhabilita mediante la opción `enable.audit.log`.
- Los valores de auditoría predeterminados están controlados por las opciones `audit.*`.
- La ubicación de los archivos de anotaciones de auditoría está controlada por la opción `daemon.log`.
- La página de códigos utilizada para grabar las anotaciones de auditoría está controlada por la directiva `_RSE_HOST_CODEPAGE`, tal como se documenta en “`rsed.envvars`, archivo de configuración de RSE” en la página 32.

Puede utilizarse el mandato de operador **modify switch** para pasar manualmente a un nuevo archivo de anotaciones de auditoría, como se describe en la sección Capítulo 8, “Mandatos de operador”, en la página 125.

Se envía un mensaje de aviso a la consola cuando el sistema de archivos que contiene los archivos de anotaciones de auditoría se está quedando sin espacio libre. Este mensaje de consola (FEK103E) se repite regularmente hasta que se ha resuelto el problema de falta de espacio. Consulte “Mensajes de consola” en la página 132 para obtener una lista de los mensajes de consola generados por RSE.

### Datos de auditoría

Un archivo de anotaciones de auditoría nuevo se inicia después de un tiempo predeterminado o cuando se emite el mandato de operador **modify switch**. El

archivo de anotaciones antiguo se guarda como `audit.log.aaaammdd.hhmmss`, donde `aaaammdd.hhmmss` es la fecha/indicación de fecha y hora de cierre de las anotaciones. La fecha/indicación de fecha y hora del sistema asignada al archivo indica la creación del archivo de anotaciones. La combinación de las dos fechas muestra el período de tiempo cubierto por este archivo de anotaciones de auditoría.

Se anotan las siguientes acciones:

- Acceso al sistema (conexión, desconexión)
- Acceso al spool de JES (someter, visualizar, retener, liberar, cancelar, depurar)
- Acceso a conjuntos de datos (lectura, grabación, creación, supresión, red denominación, compresión, migración, rellamada)
- Ejecución de mandatos TSO

Cada acción anotada se almacena (con una fecha/indicación de fecha y hora) utilizando el formato CSV (valores separados por comas), que puede leerse mediante una herramienta de análisis de datos o automatización.

Los archivos de anotaciones de auditoría tienen la máscara de bit de permiso 640 (-rw-r-----), lo que significa que el propietario (uid de z/OS UNIX del daemon RSE) tiene acceso de grabación y lectura, y el grupo del propietario (predeterminado) tiene acceso de lectura. Todos los demás intentos de acceso se denegarán, a menos que los realice un superusuario (UID 0) o alguien con permiso suficiente sobre el perfil `SUPERUSER.FILESYS` de la clase `UNIXPRIV`.

---

## Seguridad de JES

Developer for System z permite a los clientes acceder al spool de JES por medio del Supervisor de trabajos JES. El servidor proporciona limitaciones básicas de acceso, que pueden ampliarse con las características estándar de protección de archivos de spool de su producto de seguridad. Las acciones del operador (Retener, Liberar, Cancelar y Depurar) en los archivos de spool se realizan por medio de la consola de EMCS, para la que deben configurarse permisos condicionales.

### Acciones en trabajos - limitaciones de destino

El supervisor de trabajos JES no proporciona a los usuarios de Developer for System z acceso de operador pleno al spool JES. Sólo están disponibles los mandatos Retener, Liberar, Cancelar y Depurar, y, por omisión, sólo para los archivos de spool propiedad del usuario. 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 puede ampliarse utilizando perfiles de seguridad para definir para qué trabajos están disponibles los mandatos.

Parecido a la acción de SDSF **SJ**, el Supervisor de trabajos JES también soporta el mandato **Mostrar JCL** para recuperar el JCL que creó la salida del trabajo seleccionado y visualizarlo en una ventana de editor. El Supervisor de trabajos JES recupera el JCL de JES y lo convierte en una función útil para los casos en que no se puede ubicar el miembro de JCL fácilmente.

Tabla 21. Mandatos de la consola del supervisor de trabajos JES

Acción	JES2	JES3
Retener	\$Hx(idtrabajo) con x = {J, S o T}	*F,J=idtrabajo,H
Liberar	\$Ax(idtrabajo) con x = {J, S o T}	*F,J=idtrabajo,R
Cancelar	\$Cx(idtrabajo) con x = {J, S o T}	*F,J=idtrabajo,C
Purgar	\$Cx(idtrabajo),P con x = {J, S o T}	*F,J=idtrabajo,C
Mostrar JCL	no aplicable	no aplicable

Por omisión, los mandatos de JES disponibles listados en Tabla 21 están limitados a los trabajos que son propiedad del usuario. Esto puede cambiarse mediante la directiva `LIMIT_COMMANDS`, como se describe en la sección “FEJJCNFG, archivo de configuración del supervisor de trabajos JES” en la página 27.

Tabla 22. Matriz de permisos de mandato `LIMIT_COMMANDS`

LIMIT_COMMANDS	Propietario del trabajo	
	Usuario	Otros
USERID (valor predeterminado)	Permitido	No permitido
LIMITED	Permitido	Permitido sólo si lo permiten explícitamente los perfiles de seguridad
NOLIMIT	Permitido	Permitido si lo permiten los perfiles de seguridad o cuando la clase JESSPOOL no está activa

JES utiliza la clase JESSPOOL para proteger los conjuntos de datos SYSIN/SYSOUT. Parecido a SDSF, el Supervisor de trabajos JES amplía la utilización de la clase JESSPOOL para proteger también los recursos de trabajo.

Si `LIMIT_COMMANDS` no es `USERID`, el Supervisor de trabajos JES solicitará el permiso al perfil relacionado con la clase JESSPOOL, tal como se muestra en la tabla siguiente:

Tabla 23. Perfiles JESSPOOL ampliados

Mandato	Perfil JESSPOOL	Acceso necesario
Retener	nodeid.userid.jobname.jobid	ALTER
Liberar	nodeid.userid.jobname.jobid	ALTER
Cancelar	nodeid.userid.jobname.jobid	ALTER
Purgar	nodeid.userid.jobname.jobid	ALTER
Mostrar JCL	nodeid.userid.jobname.jobid.JCL	READ

Utilice la siguientes sustituciones en la tabla anterior:

idnodo	ID del nodo NJE del subsistema JES destino
idusuario	ID de usuario local del propietario del trabajo
nombretabajo	Nombre del trabajo
idtrabajo	ID del trabajo JES

Si la clase JESSPOOL no está activa, se produce un comportamiento diferente para los valores LIMITED y NOLIMIT de LIMIT\_COMMANDS, como se describe en la sección Tabla 9 en la página 30. El comportamiento es idéntico si JESSPOOL está activa, ya que, por omisión, la clase deniega el permiso si un perfil no está definido.

## Acciones en trabajos - limitaciones de ejecución

La segunda fase de la seguridad de mandatos de spool JES, una vez especificados los destinos permitidos, incluye los permisos necesarios para ejecutar realmente el mandato de operador. Las comprobaciones de seguridad de JES y z/OS aplican esta autorización de ejecución.

Tenga en cuenta que Mostrar JCL no es un mandato de operador igual que el resto de mandatos del supervisor de trabajos JES (Retener, Liberar, Cancelar y Depurar), de manera que las siguientes limitaciones no son de aplicación porque no hay ninguna comprobación de seguridad más.

El Supervisor de trabajos JES emite todos los mandatos de operador de JES solicitados por un usuario por medio de una consola de EMCS ampliada (EMCS), cuyo nombre está controlado por la directiva `CONSOLE_NAME`, tal como se describe en “FEJJCNFG, archivo de configuración del supervisor de trabajos JES” en la página 27.

Esta configuración permite al administrador de seguridad definir permisos de ejecución de mandatos granulares mediante las clases `OPERCMDS` y `CONSOLE`.

- A fin de utilizar una consola de EMCS, el usuario debe disponer (como mínimo) de una autorización de LECTURA sobre el perfil `MVS.MCSOPER.console-name` de la clase `OPERCMDS`. Tenga en cuenta que si no se define ningún perfil, el sistema otorgará la petición de autorización.
- A fin de utilizar un mandato de operador de JES, el usuario debe disponer de la autorización suficiente sobre el perfil `JES%.**` (o más concreto) de la clase `OPERCMDS`. Tenga en cuenta que si no se define ningún perfil o la clase `OPERCMDS` no está activa, el mandato fallará a causa de JES.
- El administrador de seguridad también puede requerir que un usuario utilice el supervisor de trabajos JES al ejecutar el mandato de operador especificando `WHEN(CONSOLE(JMON))` en la definición **PERMIT**. La clase `CONSOLE` debe estar activa para esta configuración del trabajo. Tenga en cuenta que es suficiente con que la clase `CONSOLE` esté activa; las consolas de EMCS no comprueban los perfiles.

El software de seguridad impide la asunción de identidad del servidor Supervisor de trabajos JES creando una consola JMON desde una sesión TSO. Aunque la consola se puede crear, el punto de entrada es distinto (supervisor de trabajos JES versus TSO). Los mandatos JES emitidos desde esta consola fallarán la comprobación de seguridad, si la seguridad está configurada según se describe en esta publicación.

Tenga en cuenta que el Supervisor de trabajos JES no puede crear la consola cuando debe ejecutarse un mandato si el nombre de consola ya se está utilizando. Para evitarlo, el programador de sistemas puede establecer la directiva `GEN_CONSOLE_NAME=ON` en el archivo de configuración del supervisor de trabajos JES o bien el administrador de seguridad puede definir perfiles de seguridad para impedir que los usuarios de TSO creen una consola. Los siguientes mandatos RACF de ejemplo impiden que nadie (excepto aquellos que lo tienen permitido) cree una consola TSO o SDSF:

- `RDEFINE TSOAUTH CONSOLE UACC(NONE)`
- `PERMIT CONSOLE CLASS(TSOAUTH) ACCESS(READ) ID(#idusuario)`
- `RDEFINE SDSF ISFCMD.ODSP.ULOG.* UACC(NONE)`
- `PERMIT ISFCMD.ODSP.ULOG.* CLASS(SDSF) ACCESS(READ) ID(#idusuario)`

**Nota:** Aunque no posean autorización sobre estos mandatos de operador, los usuarios todavía pueden someter trabajos y leer la salida de los trabajos por medio del Supervisor de trabajos JES, en caso de que dispongan de la autorización suficiente sobre los perfiles posibles que protegen estos recursos (como los de las clases JESINPUT, JESJOBS y JESSPOOL).

Para obtener más información sobre la protección de los mandatos de operador, consulte el manual *Security Server RACF Security Administrator's Guide* (SA22-7683).

## Acceso a los archivos de spool

Por omisión, el Supervisor de trabajos JES permite acceder a todos los archivos de spool. Esto puede cambiarse mediante la directiva `LIMIT_VIEW`, como se describe en la sección “FEJJCNFG, archivo de configuración del supervisor de trabajos JES” en la página 27.

Tabla 24. matriz de permisos de examen de `LIMIT_VIEW`

LIMIT_VIEW	Propietario del trabajo	
	Usuario	Otros
USERID	Permitido	No permitido
NOLIMIT (valor predeterminado)	Permitido	Permitido si lo permiten los perfiles de seguridad o cuando la clase JESSPOOL no está activa

Para limitar a los usuarios de forma que solo utilicen sus propios trabajos en el spool de JES, defina la sentencia "`LIMIT_VIEW=USERID`" en el archivo de configuración del supervisor de trabajos JES, FEJJCNFG. Si los usuarios 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.

Al definir protección adicional, tenga presente que el supervisor de trabajos JES utiliza 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. Este requisito no es necesario si se ejecuta z/OS 1.7 (z/OS 1.8 para JES3) o superior. En este caso, el permiso de lectura (READ) es suficiente para la función de examen.

Consulte la publicación *Security Server RACF Security Administrator's Guide* (SA22-7683) para obtener más información acerca de la protección de archivos de spool de JES.

## Comunicación cifrada con SSL

La comunicación externa (cliente-host) puede cifrarse mediante SSL (Capa de sockets seguros). Esta característica está inhabilitada por omisión y está controlada por los valores de `ssl.properties`, tal como se describe en "(Opcional) Cifrado SSL de RSE" en la página 93.

El daemon RSE y el servidor RSE soportan distintos mecanismos para almacenar certificados debido a las diferencias de su arquitectura. Esto hace que las definiciones y certificados SSL sean necesarias tanto para el daemon RSE como para el servidor RSE. Se puede utilizar un certificado compartido si el daemon RSE y el servidor RSE utilizan el mismo método de gestión de certificados.

*Tabla 25. Mecanismos de almacenamiento de certificados de SSL*

Almacenamiento de certificados	Creado y gestionado por	daemon RSE	servidor RSE
anillo de claves	producto de seguridad compatible con SAF	soportado	soportado
base de datos de claves	gskkyman de z/OS UNIX	soportado	/
almacén de claves	Keytool de Java	/	soportado

**Nota:** Los anillos de claves compatibles con SAF son el método preferido para gestionar certificados.

Los anillos de claves compatibles con SAF puede almacenar la clave privada del certificado en la base de datos de seguridad o mediante el ICSF (recurso de servicio criptográfico integrado), la interfaz al hardware criptográfico de System z.

Se recomienda el ICSF para el almacenamiento de claves privadas relacionadas con certificados digitales, ya que es una solución más segura que la gestión de claves privadas sin ICSF. ICSF asegura que las claves privadas se cifran con la clave maestra de ICSF y que el acceso a ellas está controlado por los recursos generales de las clases de seguridad CSFKEYS y CSFSERV. Además, el rendimiento operativo mejora porque ICSF utiliza el hardware Coprocesador criptográfico.

El daemon RSE utiliza funciones de SSL del sistema para gestionar las comunicaciones cifradas con SSL. Ello implica que `SYS1.SIEALNKE` debe estar controlado por programa por su software de seguridad y disponible para RSE a través de la directiva `LINKLIST` o `STEPLIB` en `rsed.envvars`.

El ID de usuario de RSE (`STCRSE` en los mandatos de ejemplo siguientes) necesita una autorización para acceder a su anillo de claves y a los certificados relacionados cuando se utilizan anillos de claves compatibles con SAF ya sea para el daemon RSE o para el servidor RSE.

- `RDEFINE FACILITY IRR.DIGTCERT.LIST UACC(NONE)`
- `RDEFINE FACILITY IRR.DIGTCERT.LISTRING UACC(NONE)`

- PERMIT IRR.DIGTCERT.LIST CLASS(FACILITY) ACCESS(READ) ID(stcrse)
- PERMIT IRR.DIGTCERT.LISTRING CLASS(FACILITY) ACCESS(READ) ID(stcrse)
- SETROPTS RACLIST(FACILITY) REFRESH

Consulte el Apéndice A, “Configurar SSL y autenticación de X.509”, en la página 295 para obtener más detalles acerca de la activación de SSL para Developer for System z.

---

## Autenticación de cliente mediante certificados X.509

El daemon RSE admite que los usuarios se autenticuen con un certificado X.509. El uso de una comunicación cifrada con SSL es un requisito previo para utilizar esta función, dado que es una extensión de la autenticación de host con un certificado utilizado en SSL.

El daemon RSE inicia el proceso de autenticación de cliente validando el certificado de cliente. Algunos de los aspectos clave que se comprueban son las fechas de validez del certificado y la fiabilidad de la autoridad certificadora (CA) utilizada para la firma del certificado. Opcionalmente, también se puede consultar una Lista de certificados revocados (CLR) (externa).

Una vez el daemon RSE ha validado el certificado, este es procesado para su autenticación. El certificado pasa a su producto de seguridad para que lo autentique, a menos que la directiva de `rsed.envvars.enable.certificate.mapping` tenga un valor `false`, en cuyo caso el daemon RSE se encargará de la autenticación.

Si se realiza con éxito, el proceso de autenticación determinará el ID de usuario que deberá utilizarse para esta sesión; posteriormente, el daemon RSE lo probará para asegurar que es adecuado para el sistema host donde se está ejecutando el daemon RSE

La última comprobación (que realizar para todos los mecanismos de autenticación, no sólo para los certificados X.509) verifica que el ID de usuario puede utilizar Developer for System z.

Si está familiarizado con las clasificaciones de seguridad SSL utilizadas por TCP/IP, la combinación de estos pasos de validación coinciden con las especificaciones del “Nivel 3 de autenticación de cliente” (el nivel más alto disponible).

## Validación de la autoridad certificadora (CA)

Parte del proceso de validación del certificado incluye la comprobación de que el certificado ha sido firmado por una autoridad certificadora (CA) de confianza. Para ello, el daemon RSE debe tener acceso a un certificado que identifique a la CA.

Al utilizar la base de datos de claves **gskkyman** para su conexión SSL, debe añadirse a la base de datos de claves el certificado de CA.

Al utilizar un anillo de claves de SAF (método recomendado), debe añadir el certificado de CA a su base de datos de seguridad como certificado CERTAUTH con el atributo TRUST o HIGHTRUST, tal como se muestra en este mandato RACF de ejemplo:

- RACDCERT CERTAUTH ADD(dsn) HIGHTRUST WITHLABEL('label')

Tenga en cuenta que la mayor parte de productos de seguridad ya tienen los certificados de conocidas CA disponibles en sus bases de datos con estado NOTRUST. Utilice estos mandatos RACF de ejemplo para enumerar los certificados de CA existentes y marque uno de ellos como De confianza en base a la etiqueta que tiene asignada.

- RACDCERT CERTAUTH LIST
- RACDCERT CERTAUTH ALTER(LABEL('HighTrust CA')) HIGHTRUST

**Nota:** El estado HIGHTRUST es necesario si confía en que RACF autentique al usuario en base a la extensión HostIdMappings del certificado. Para obtener más información, consulte “Autenticación del software de seguridad”.

Una vez el certificado de CA está añadido a su base de datos de seguridad, debe conectarse al anillo de claves del RSE, tal como se muestra en el mandato RACF de ejemplo:

- RACDCERT ID(stcrse) CONNECT(CERTAUTH LABEL('HighTrust CA')  
RING(rdzssl.racf))

Consulte el manual *Security Server RACF Command Language Reference* (SA22-7687) para obtener más información sobre el mandato **RACDCERT**.

**Atención:** si confía más en el daemon RSE que en su software de seguridad para autenticar un usuario, debe tener cuidado de no mezclar las CA con los estados TRUST y HIGHTRUST en su anillo de claves de SAF ni en su base de datos de claves de **gskkyman**. El daemon RSE no los puede diferenciar, por lo que los certificados firmados por una CA con estado TRUST serán válidos para la autenticación del ID de usuario.

## (Opcional) Consulta en una lista de certificados revocados (CRL)

Si lo desea, puede indicar al daemon RSE que compruebe una o varias Listas de certificados revocados (CRL) para hacer más seguro el proceso de validación. Para hacerlo, añada variables de entorno relacionadas con la CRL a `rsed.envvars`. Consulte “`rsed.envvars`, archivo de configuración de RSE” en la página 32 para obtener más información sobre estas variables de ejemplo:

- GSK\_CRL\_SECURITY\_LEVEL
- GSK\_LDAP\_SERVER
- GSK\_LDAP\_PORT
- GSK\_LDAP\_USER
- GSK\_LDAP\_PASSWORD

Consulte el manual *Cryptographic Services System Secure Sockets Layer Programming* (SC24-5901) para obtener más información sobre estas y otras variables de entorno utilizadas por SSL del sistema de z/OS.

**Nota:** Tenga cuidado al especificar otras variables de entorno de SSL del sistema de z/OS (`GSK_*`) en `rsed.envvars`, ya que pueden cambiar la forma en que el daemon RSE maneja las conexiones SSL y la autenticación de certificados.

## Autenticación del software de seguridad

RACF lleva a cabo varias comprobaciones para autenticar un certificado y devolver el ID de usuario asociado. Tenga en cuenta que puede que otros productos de seguridad lo hagan de manera distinta. Consulte la documentación de su producto

de seguridad para obtener información adicional sobre la función `initACEE` utilizada para llevar a cabo la autenticación (modalidad de consulta).

1. RACF comprueba si el certificado está definido en la clase `DIGTCERT`. De ser así, RACF devuelve el ID de usuario que estaba asociado a este certificado cuando se añadió a la base de datos RACF.

Los certificados se definen en RACF mediante el mandato `RACDCERT`, como en el ejemplo siguiente:

```
RACDCERT ID(userid) ADD(dsn) TRUST WITHLABEL('label')
```

2. En caso que el certificado no esté definido, RACF comprueba si hay un filtro de nombre de certificado coincidente en las clases `DIGTNMAP` o `DIGTCRIT`. De ser así, devuelve el ID de usuario asociado al filtro coincidente más específico.

**Nota:** Se recomienda no utilizar filtros de nombre para los certificados utilizados por Developer for System z, ya que estos filtros correlacionan todos los certificados con un único ID de usuario. El resultado es que todos sus usuarios de Developer for System z iniciarán sesión con el mismo ID de usuario.

3. Si no hay ningún filtro de nombre coincidente, RACF ubica la extensión de certificado de `HostIdMappings` y extrae el par de nombre de host e ID de usuario incluido. En caso de que se encuentre y se valide, RACF devuelve el ID de usuario definido dentro de la extensión de `HostIdMappings`.

El par de ID de usuario y nombre de host es válido si todas estas condiciones son true:

- El certificado de CA utilizado para firmar este certificado está marcado como `HIGHTRUST` en la clase `DIGTCERT`.
- El ID de usuario almacenado en la extensión tiene una longitud válida (de 1 a 8 caracteres).
- El ID de usuario asignado al daemon RSE tiene (como mínimo) autorización de `LECTURA` sobre el perfil `IRR.HOST.hostname` en la clase de `SERVAUTH`, donde `hostname` es el nombre de host almacenado en la extensión. Este suele ser un nombre de dominio como, por ejemplo, `CDFMVS08.RALEIGH.IBM.COM`.

La definición de la extensión de `HostIdMappings` en sintaxis ASN.1 es:

```
id-ce-hostIdMappings OBJECT IDENTIFIER ::= { 1 3 18 0 2 18 1 }
HostIdMappings ::= SET OF HostIdMapping
HostIdMapping ::= SEQUENCE {
    hostName          IMPLICIT[1] IA5String,
    subjectId         IMPLICIT[2] IA5String,
    proofOfIdPossession IdProof OPTIONAL
}
IdProof ::= SEQUENCE {
    secret            OCTET STRING,
    encryptionAlgorithm OBJECT IDENTIFIER
}
```

**Nota:** Una extensión de `HostIdMappings` no se respeta si el ID de usuario destino ha sido creado una vez iniciado el período de validez del certificado que incluye la extensión de `HostIdMappings`. Por ello, si está creando ID de usuario concretamente para certificados con extensiones de `HostIdMappings`, asegúrese de que crea el ID de usuario antes de que se envíen las peticiones del certificado.

Consulte el manual *Security Server RACF Security Administrator's Guide* (SA22-7683) para obtener más información sobre certificados X.509, la forma es que estos son gestionados por RACF y cómo definir filtros de nombre de

certificado. Consulte el manual *Security Server RACF Command Language Reference* (SA22-7687) para obtener más información sobre el mandato RACDCERT.

## Autenticación del daemon RSE

Developer for System z puede llevar a cabo la autenticación básica de certificados X.509 sin basarse en su producto de seguridad. La autenticación llevada a cabo por el daemon RSE requiere que se definan un ID de usuario y un nombre de host en una extensión de certificado, y solamente se activa si la directiva `enable.certificate.mapping` de `rsed.envvars` tiene el valor de `FALSE`.

Esta función debe utilizarse cuando su producto de seguridad no admita la autenticación de un usuario en base a un certificado X.509, o bien si en el caso que su certificado fallara la(s) prueba(s) realizadas por el producto de seguridad (por ejemplo, si el certificado tiene un identificador erróneo para la extensión de `HostIdMappings` y no hay ninguna definición ni filtro de nombre en `DIGTCERT`).

El cliente consultará al usuario qué identificador de extensión (OID) utilizar, que es, de forma predeterminada, el OID de `HostIdMappings` {1 3 18 0 2 18 1}.

El daemon RSE le extraerá el ID de usuario y el nombre de host utilizando el formato de la ampliación de `HostIdMappings`. Este formato se describe en “Autenticación del software de seguridad” en la página 172.

El par de ID de usuario y nombre de host es válido si todas estas condiciones son true:

- El ID de usuario almacenado en la extensión tiene una longitud válida (de 1 a 8 caracteres).
- El ID de usuario asignado al daemon RSE tiene (como mínimo) autorización de LECTURA sobre el perfil `IRR.HOST.hostname` en la clase de `SERVAUTH`, donde `hostname` es el nombre de host almacenado en la extensión. Este suele ser un nombre de dominio como, por ejemplo, `CDFMVS08.RALEIGH.IBM.COM`.

**Atención:** Es decisión del administrador de seguridad asegurar que todas las CA reconocidas por el daemon RSE sean de confianza total, dado que el daemon RSE no puede comprobar si la persona que ha firmado el certificado de cliente es de confianza total o es simplemente de confianza. Consulte “Validación de la autoridad certificadora (CA)” en la página 171 para obtener más información sobre los certificados de CA accesibles.

---

## Comprobación de puerto de entrada (POE)

Developer for System z da soporte a la comprobación de puerto de entrada (POE), que permite el acceso de host sólo a las direcciones TCP/IP de confianza. Esta característica está inhabilitada por omisión y requiere la definición del perfil de seguridad `BPX.POE`, como se muestra en los mandatos RACF de ejemplo que figuran a continuación:

- `RDEFINE FACILITY BPX.POE UACC(NONE)`
- `PERMIT BPX.POE CLASS(FACILITY) ACCESS(READ) ID(STCRSE)`
- `SETOPTS RACLIST(FACILITY) REFRESH`

**Nota:**

- El RSE debe configurarse para utilizar POE descomentando la opción “enable.port.of.entry=true” del archivo `rsed.envvars`, como se describe en la sección “Definir parámetros de inicio Java adicionales con `_RSE_JVAOPTS`” en la página 42.
- El ID de usuario de RSE de STCRSE requiere el `UID(0)` si el perfil no está definido y la comprobación de POE está habilitada en `rsed.envvars`.
- El hecho de definir `BPX.POE` influirá sobre otras aplicaciones TCP/IP que den soporte a la comprobación de POE, como `INETD`.
- Deben configurarse zonas de seguridad (perfiles de `EZB.NETACCESS.**`, que son rangos de direcciones IP) en la clase `SERVAUTH` para poder utilizar todas las posibilidades de la comprobación de POE.

Consulte la publicación *Communications Server IP Configuration Guide* (SC31-8775) para obtener más información acerca del control de acceso a la red mediante comprobaciones de POE.

---

## Seguridad de CICS TS

Developer for System z permite, mediante el Gestor de despliegue de aplicaciones, que los administradores de CICS controlen qué definiciones de recursos CICS puede editar el desarrollador, sus valores predeterminados y la visualización de una definición de recurso CICS por medio del servidor CRD (definiciones de recursos CICS). Consulte la sección Capítulo 15, “consideraciones de CICS TS”, en la página 253 para obtener más información acerca de las definiciones de recursos CICS TS necesarias.

### Repositorio CRD

El conjunto de datos VSAM del repositorio del servidor CRD contiene todas las definiciones de recurso predeterminadas y, por tanto, debe protegerse contra actualizaciones, pero los desarrolladores deben poder leer los valores almacenados en él.

### Transacciones CICS

Developer for System z suministra varias transacciones que el servidor CRD utiliza al definir y consultar recursos CICS. Cuando se conecta la transacción, la comprobación de la seguridad de recursos CICS, si está habilitada, se asegura de que el ID de usuario tiene autorización para ejecutar el ID de transacción.

### Comunicación cifrada con SSL

El cliente Gestor de despliegue de aplicaciones utiliza la interfaz RESTful o los servicios Web de CICS TS para invocar el servidor CRD. La utilización de SSL para esta comunicación está controlada por la definición `TCPIPService` de CICS TS, tal como se describe en *RACF Security Guide for CICS TS*.

---

## Seguridad de SCLM

El servicio SCLM Developer Toolkit ofrece funciones de seguridad opcionales para las funciones de construcción, promoción y despliegue.

Si el administrador de SCLM habilita la seguridad para una función, se realizan llamadas SAF para comprobar la autorización para ejecutar la función protegida con el ID de usuario del llamante o uno subordinado.

Consulte la *Guía del administrador de SCLM Developer Toolkit*, SC11-3815 (SC23-9801), para obtener más información acerca de las definiciones de seguridad de SCLM necesarias.

---

## Archivos de configuración de Developer for System z

Existen varios archivos de configuración de Developer for System z, cuyas directivas afectan a la configuración de seguridad. En base a la información de este capítulo, el administrador de seguridad y el programados de sistemas pueden decidir cuáles deberían ser los valores para las directivas siguientes.

### Rastreo del daemon de bloqueo - FEJJCNFG

- `LIMIT_COMMANDS={USERID | LIMITED | NOLIMIT }`  
Definir en qué trabajos pueden realizarse las acciones (excluidas las acciones examinar y someter). Para obtener más información, consulte “Acciones en trabajos - limitaciones de destino” en la página 166.
- `LIMIT_VIEW={USERID | NOLIMIT}`  
Definir qué archivos de spool pueden examinarse. Para obtener más información, consulte “Acceso a los archivos de spool” en la página 169.
- `APPLID={FEKAPPL | *}`  
ID de aplicación utilizado para la creación/validación de PassTicket. Para obtener más información, consulte “Utilizar PassTickets” en la página 164.

**Nota:** Puede obtener detalles sobre estas y otras directivas FEJJCNFG en “FEJJCNFG, archivo de configuración del supervisor de trabajos JES” en la página 27.

### RSE - rsed.envvars

- `( _RSE_JAVAOPTS ) -DDENY_PASSWORD_SAVE={true | false}`  
Denegar a los usuarios que guarden la contraseña del host en el cliente. Para obtener más información, consulte “Definir parámetros de inicio Java adicionales con \_RSE\_JAVAOPTS” en la página 42.
- `( _RSE_JAVAOPTS ) -DDSTORE_IDLE_SHUTDOWN_TIMEOUT=value`  
Temporizador para desconectar a los clientes desocupados. Para obtener más información, consulte “Definir parámetros de inicio Java adicionales con \_RSE\_JAVAOPTS” en la página 42.
- `( _RSE_JAVAOPTS ) -DAPPLID={FEKAPPL | *}`  
ID de aplicación utilizado para la creación/validación de PassTicket. Para obtener más información, consulte “Utilizar PassTickets” en la página 164.
- `( _RSE_JAVAOPTS ) -Denable.port.of.entry={true | false}`  
Habilitar la comprobación de puerto de entrada. Para obtener más información, consulte “Comprobación de puerto de entrada (POE)” en la página 174.
- `( _RSE_JAVAOPTS ) -Denable.certificate.mapping={true | false}`  
Utilizar su producto de seguridad para autenticar a los usuarios con un certificado X.509. Para obtener más información, consulte “Autenticación de cliente mediante certificados X.509” en la página 171.
- `( _RSE_JAVAOPTS ) -Ddaemon.log={/var/rdz/logs | *}`  
Ubicación de los archivos de anotaciones de auditoría. Para obtener más información, consulte “Anotaciones de auditoría” en la página 165.

**Nota:** Puede obtener detalles sobre estas y otras directivas `rsed.envvars` en “`rsed.envvars`, archivo de configuración de RSE” en la página 32.

## RSE - `ssl.properties`

- `daemon_keydb_file={SAF key ring name | gskkyman key database name}`  
Ubicación del certificado del daemon RSE. Para obtener más información, consulte “Comunicación cifrada con SSL” en la página 170.
- `daemon_key_label=certificate label`  
Nombre del certificado del daemon RSE. Para obtener más información, consulte “Comunicación cifrada con SSL” en la página 170.
- `server_keystore_file={SAF key ring name | Java key store name}`  
Ubicación del certificado del servidor RSE. Para obtener más información, consulte “Comunicación cifrada con SSL” en la página 170.
- `server_keystore_label=certificate label`  
Nombre del certificado del servidor RSE. Para obtener más información, consulte “Comunicación cifrada con SSL” en la página 170.
- `server_keystore_type={JKS | JCECARCFKS | JCECCARCFKS}`  
Tipo de almacén de claves utilizado (almacén de claves Java o anillo de claves de SAF). Para obtener más información, consulte “Comunicación cifrada con SSL” en la página 170.

**Nota:** Puede obtener detalles sobre estas y otras directivas `ssl.properties` en “(Opcional) Cifrado SSL de RSE” en la página 93.

---

## Definiciones de seguridad

Personalice y someta el miembro de ejemplo FEKRACF, que contiene mandatos de ejemplo RACF y z/OS UNIX para crear las definiciones básicas de seguridad para Developer for System z.

FEKRACF se encuentra en `FEK.#CUST.JCL`, a menos que haya especificado otra ubicación al personalizar y someter el trabajo `FEK.SFEKSAMP(FEKSETUP)`. Encontrará más detalles en: “Configuración de la personalización” en la página 15.

Para obtener más información sobre los mandatos RACF, consulte el manual *RACF Command Language Reference* (SA22-7687).

### Nota:

- Para los sitios que utilizan CA ACF2™ para z/OS, consulte el enlace siguiente, <https://support.ca.com/irj/portal/kbtech?ipLogNrow=0&docid=492389&searchID=TEC492389>, para obtener detalles sobre los mandatos de seguridad necesarios para configurar adecuadamente Developer for System z.
- Para los sitios que utilizan CA Top Secret® para z/OS, consulte la página del producto de la página de soporte de CA (<https://support.ca.com>) y compruebe el Documento de conocimiento de Developer for System z relacionado. Este Documento de conocimiento contiene detalles sobre los mandatos de seguridad necesarios para configurar correctamente Developer for System z.

Las secciones que siguen describen los pasos necesarios, la configuración opcional y las posibles alternativas.

## Requisitos y lista de comprobación

Para completar la configuración de seguridad, el administrador de seguridad necesita conocer los valores enumerados en Tabla 26. Estos valores se han definido durante los pasos anteriores de la instalación y personalización de Developer for System z.

Tabla 26. Variables de configuración de seguridad

Descripción	<ul style="list-style-type: none"> <li>• Valor predeterminado</li> <li>• Dónde encontrar la respuesta</li> </ul>	Valor
Calificador de producto de alto nivel de Developer for System z	<ul style="list-style-type: none"> <li>• FEK</li> <li>• Instalación de SMP/E</li> </ul>	
Calificador de personalización de alto nivel de Developer for System z	<ul style="list-style-type: none"> <li>• FEK.#CUST</li> <li>• FEK.SFEKSAMP(FEKSETUP), tal como se describe en “Configuración de la personalización” en la página 15.</li> </ul>	
Nombre de tarea iniciada del Supervisor de trabajos JES	<ul style="list-style-type: none"> <li>• JMON</li> <li>• FEK.#CUST.PROCLIB(JMON), tal como se describe en “Cambios de PROCLIB” en la página 21.</li> </ul>	
Nombre de tarea iniciada del daemon RSE	<ul style="list-style-type: none"> <li>• RSED</li> <li>• FEK.#CUST.PROCLIB(RSED), tal como se describe en “Cambios de PROCLIB” en la página 21.</li> </ul>	
Nombre de tarea iniciada del daemon de bloqueo	<ul style="list-style-type: none"> <li>• LOCKD</li> <li>• FEK.#CUST.PROCLIB(LOCKD), tal como se describe en “Cambios de PROCLIB” en la página 21.</li> </ul>	
ID de aplicación	<ul style="list-style-type: none"> <li>• FEKAPPL</li> <li>• /etc/rdz/rsed.envvars, tal como se describe en “Definir parámetros de inicio Java adicionales con _RSE_JAVAOPTS” en la página 42</li> </ul>	

La lista que sigue es una visión general de las acciones necesarias para completar la configuración de seguridad básica de Developer for System z. Tal como se describe en las secciones siguientes, se pueden utilizar distintos métodos para cumplir estos requisitos en función del nivel de seguridad deseado. Consulte las secciones anteriores para obtener información sobre la configuración de seguridad de los servicios de Developer for System z opcionales.

- “Activar valores y clases de seguridad” en la página 179
- “Definir un segmento OMVS para usuarios de Developer for System z” en la página 180

- “Definir perfiles de conjunto de datos” en la página 180
- “Definir las tareas iniciadas de Developer for System z” en la página 183
- “Definir la seguridad de mandatos JES” en la página 184
- “Definir RSE como servidor z/OS UNIX seguro” en la página 186
- “Definir las bibliotecas controladas por programa MVS para RSE” en la página 186
- “Definir la protección de aplicaciones para RSE” en la página 187
- “Definir el soporte de PassTicket para RSE” en la página 187
- “Definir los archivos controlados por programa z/OS UNIX para RSE” en la página 188
- “Verificar valores de seguridad” en la página 189

## Activar valores y clases de seguridad

Developer for System z utiliza diversos mecanismos de seguridad para garantizar un entorno de host seguro y controlado para el cliente. Para ello deben activarse varias clases y valores de seguridad, como se muestra en los siguientes mandatos RACF de ejemplo:

- Visualizar valores actuales
  - SETROPTS LIST
- Activar clases de recurso para perfiles de certificados digitales y de z/OS UNIX
  - SETROPTS GENERIC(FACILITY)
  - SETROPTS CLASSACT(FACILITY) RACLIST(FACILITY)
- Activar definiciones de tareas iniciadas
  - SETROPTS GENERIC(STARTED)
  - RDEFINE STARTED \*\* STDATA(USER(=MEMBER) GROUP(STCGROUP) TRACE(YES))
  - SETROPTS CLASSACT(STARTED) RACLIST(STARTED)
- Activar seguridad de consola para el Supervisor de trabajos JES
  - SETROPTS GENERIC(CONSOLE)
  - SETROPTS CLASSACT(CONSOLE) RACLIST(CONSOLE)
- Activar protección de mandatos de operador para el Supervisor de trabajos JES
  - SETROPTS GENERIC(OPERCMDS)
  - SETROPTS CLASSACT(OPERCMDS) RACLIST(OPERCMDS)
- Activar protección de aplicaciones para el RSE
  - SETROPTS GENERIC(APPL)
  - SETROPTS CLASSACT(APPL) RACLIST(APPL)
- Activar el inicio de sesión seguro mediante PassTickets para el RSE
  - SETROPTS GENERIC(PTKTDATA)
  - SETROPTS CLASSACT(PTKTDATA) RACLIST(PTKTDATA)
- Activar control de programa para garantizar que el RSE sólo pueda cargar código de confianza
  - RDEFINE PROGRAM \*\* ADDMEM('SYS1.CMDLIB'//NOPADCHK) UACC(READ)
  - SETROPTS WHEN(PROGRAM)

**Nota:** No cree el perfil \*\* si ya tiene un perfil \* en la clase PROGRAM. Oscurece y complica la vía de acceso de búsqueda utilizada por el software de seguridad. En este caso, debe fusionar las definiciones \* existentes y las definiciones \*\* nuevas. IBM recomienda utilizar el perfil \*\*, como se describe en la publicación *Security Server RACF Security Administrator's Guide* (SA22-7683).

**Atención:** Algunos productos, por ejemplo FTP, deben estar controlados por programa si "WHEN PROGRAM" está activo. Debe someter a prueba este procedimiento antes de activarlo en un sistema de producción.

- (Opcional) Activar el soporte del puerto de entrada (POE) ampliado y el HostIdMappings X.509
  - SETROPTS GENERIC(SERVAUTH)
  - SETROPTS CLASSACT(SERVAUTH) RACLIST(SERVAUTH)

## Definir un segmento OMVS para usuarios de Developer for System z

Debe definirse un segmento OMVS de RACF (o equivalente) que especifique un ID de usuario (UID) de z/OS UNIX válido que no sea cero, un directorio inicial y un mandato de shell para cada usuario de Developer for System z. Su grupo predeterminado también requiere un segmento OMVS con un id de grupo.

En los mandatos RACF de ejemplo que figuran a continuación, sustituya los espacios reservados #idusuario, #identificador-usuario, #nombre-grupo e #identificador-grupo por los valores reales:

- ALTUSER #idusuario  
OMVS(UID(#identificador-usuario) HOME(/u/#idusuario) PROGRAM(/bin/sh) NOASSIZEMAX)
- ALTGROUP #nombre-grupo OMVS(GID(#identificador-grupo))

Aunque no es aconsejable hacerlo, puede utilizar el segmento OMVS compartido definido en el perfil BPX.DEFAULT.USER de la clase FACILITY para satisfacer el requisito de segmento OMVS de Developer for System z.

## Definir perfiles de conjunto de datos

El acceso de lectura (READ) para los usuarios y de modificación (ALTER) para los programadores de sistemas es suficiente para la mayoría de conjuntos de datos de Developer for System z. Sustituya el espacio reservado #progsis por identificadores de usuario o nombres de grupo de RACF válidos. Solicite al programador del sistema que ha instalado y configurado el producto los nombres de conjunto de datos correctos. FEK es el calificador de alto nivel predeterminado utilizado durante la instalación y FEK.#CUST es el calificador de alto nivel predeterminado para los conjuntos de datos creados durante el proceso de personalización.

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

### Nota:

- Es muy aconsejable proteger FEK.SFEKAUTH contra actualizaciones, ya que este conjunto de datos tiene autorización APF. Lo mismo puede decirse de FEK.SFEKLOAD y FEK.SFEKLPA, pero en este caso debido a que estos conjuntos de datos están controlados por programa.
- En los mandatos de ejemplo se esta publicación y en el trabajo FEKRACF se presupone que EGN (Denominación genérica mejorada) está activa. Esto permite utilizar el calificador \*\* para representar cualquier número de calificadores en la clase DATASET. Sustituya \*\* por \* si EGN no está activa

en el sistema. Consulte la publicación *Security Server RACF Security Administrator's Guide* (SA22-7683) para obtener más información acerca de EGN.

Algunos de los componentes opcionales de Developer for System z requieren perfiles de conjunto de datos de seguridad adicionales. Sustituya los espacios reservados #progsis, #desarrollador-ram y #admindics por identificadores de usuario o nombres de grupo de RACF válidos:

- Si se utiliza la conversión de nombres largos/abreviados de SCLM Developer Toolkit, los usuarios necesitarán acceso de actualización (UPDATE) al VSAM de correlación, FEK.#CUST.LSTRANS.FILE.
  - ADDSD 'FEK.#CUST.LSTRANS.\*.\*' UACC(UPDATE)
  - DATA('RATIONAL DEVELOPER FOR SYSTEM Z - SCLMDT')
  - PERMIT 'FEK.#CUST.LSTRANS.\*.\*' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
  - SETROPTS GENERIC(DATASET) REFRESH
- Los desarrolladores de RAM (Repository Access Manager) de CARMA requieren acceso de actualización (UPDATE) a los VSAM de CARMA, FEK.#CUST.CRA\*.
  - ADDSD 'FEK.#CUST.CRA\*.\*' UACC(READ)
  - DATA('RATIONAL DEVELOPER FOR SYSTEM Z - CARMA')
  - PERMIT 'FEK.#CUST.CRA\*.\*' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
  - 
  - PERMIT 'FEK.#CUST.CRA\*.\*' CLASS(DATASET) ACCESS(UPDATE) ID(#desarrollador-ram)
  - SETROPTS GENERIC(DATASET) REFRESH
- Si se utiliza el servidor CRD (definición de recurso CICS) del Gestor de despliegue de aplicaciones, los administradores de CICS necesitan acceso de actualización (UPDATE) al VSAM del repositorio de CRD.
  - ADDSD 'FEK.#CUST.ADNREP\*.\*' UACC(READ)
  - DATA('RATIONAL DEVELOPER FOR SYSTEM Z - ADN')
  - PERMIT 'FEK.#CUST.ADNREP\*.\*' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
  - PERMIT 'FEK.#CUST.ADNREP\*.\*' CLASS(DATASET) ACCESS(UPDATE) ID(#admindics)
  - SETROPTS GENERIC(DATASET) REFRESH
- Si se define el repositorio de manifiestos del Gestor de despliegue de aplicaciones, todos los usuarios de CICS Transaction Server necesitan acceso de actualización (UPDATE) al VSAM del repositorio de manifiestos.
  - ADDSD 'FEK.#CUST.ADNMAN\*.\*' UACC(UPDATE)
  - DATA('RATIONAL DEVELOPER FOR SYSTEM Z - ADN')
  - PERMIT 'FEK.#CUST.ADNMAN\*.\*' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
  - SETROPTS GENERIC(DATASET) REFRESH

Utilice los siguientes mandatos RACF de ejemplo para establecer una configuración más segura, en la que el acceso de lectura (READ) también esté controlado.

- protección de conjunto de datos uacc(none)
  - ADDGROUP (FEK)
  - DATA('RATIONAL DEVELOPER FOR SYSTEM Z - HLQ STUB')
  - OWNER(IBMUSER) SUPGROUP(SYS1)"
  - ADDSD 'FEK.\*.\*' UACC(NONE)
  - DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
  - ADDSD 'FEK.SFEKAUTH' UACC(NONE)
  - DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
  - ADDSD 'FEK.SFEKLOAD' UACC(NONE)
  - DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
  - ADDSD 'FEK.SFEKPROC' UACC(NONE)
  - DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
  - ADDSD 'FEK.#CUST.PARMLIB' UACC(NONE)
  - DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
  - ADDSD 'FEK.#CUST.CNTL' UACC(NONE)
  - DATA('RATIONAL DEVELOPER FOR SYSTEM Z')

- ADDSD 'FEK.#CUST.LSTRANS.\*.\*' UACC(NONE)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - SCLMDT')
- ADDSD 'FEK.#CUST.CRA\*.\*' UACC(NONE)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - CARMA')
- ADDSD 'FEK.#CUST.ADNREP\*.\*' UACC(NONE)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - ADN')
- ADDSD 'FEK.#CUST.ADNMAN\*.\*' UACC(NONE)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - ADN')
- permitir al programador del sistema gestionar todas las bibliotecas
  - PERMIT 'FEK.\*.\*' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
  - PERMIT 'FEK.SFEKAUTH' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
  - PERMIT 'FEK.SFEKLOAD' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
  - PERMIT 'FEK.SFEKLOAD' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
  - PERMIT 'FEK.SFEKLOAD' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
  - PERMIT 'FEK.SFEKPROC' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
  - PERMIT 'FEK.#CUST.PARMLIB' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
  - PERMIT 'FEK.#CUST.CNTL' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
  - PERMIT 'FEK.#CUST.LSTRANS.\*.\*' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
  - PERMIT 'FEK.#CUST.CRA\*.\*' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
  - PERMIT 'FEK.#CUST.ADNREP\*.\*' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
  - PERMIT 'FEK.#CUST.ADNMAN\*.\*' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
- permitir a los clientes acceder a las bibliotecas de carga y exec
  - PERMIT 'FEK.SFEKAUTH' CLASS(DATASET) ACCESS(READ) ID(\*)
  - PERMIT 'FEK.SFEKLOAD' CLASS(DATASET) ACCESS(READ) ID(\*)
  - PERMIT 'FEK.SFEKPROC' CLASS(DATASET) ACCESS(READ) ID(\*)
  - PERMIT 'FEK.#CUST.CNTL' CLASS(DATASET) ACCESS(READ) ID(\*)

**Nota:** No son necesarios permisos para FEK.SFEKLPA, ya que todos los usuarios pueden acceder a todos los códigos que residen en LPA.

- permitir al Supervisor de trabajos JES acceder a la biblioteca de carga y de parámetros
  - PERMIT 'FEK.SFEKAUTH' CLASS(DATASET) ACCESS(READ) ID(STCJMON)
  - PERMIT 'FEK.#CUST.PARMLIB' CLASS(DATASET) ACCESS(READ) ID(STCJMON)
- (opcional) permitir a los clientes actualizar el VSAM de conversión de nombres largos/abreviados para SCLMDT
  - PERMIT 'FEK.#CUST.LSTRANS.\*.\*' CLASS(DATASET) ACCESS(UPDATE) ID(\*)
- (opcional) permitir a los desarrolladores de RAM actualizar los VSAM de CARMA para CARMA
  - PERMIT 'FEK.#CUST.CRA\*.\*' CLASS(DATASET) ACCESS(UPDATE) ID(#desarrollador-ram)
- (opcional) permitir a los usuarios de CICS leer el VSAM del repositorio de CRD para el Gestor de despliegue de aplicaciones
  - PERMIT 'FEK.#CUST.ADNREP\*.\*' CLASS(DATASET) ACCESS(READ) ID(\*)
- (opcional) permitir a los administradores de CICS actualizar el VSAM del repositorio de CRD para el Gestor de despliegue de aplicaciones
  - PERMIT 'FEK.#CUST.ADNREP\*.\*' CLASS(DATASET) ACCESS(UPDATE) ID(#admincics)
- (opcional) permitir a los usuarios de CICS actualizar el VSAM del repositorio de manifiestos para el Gestor de despliegue de aplicaciones
  - PERMIT 'FEK.#CUST.ADNMAN\*.\*' CLASS(DATASET) ACCESS(UPDATE) ID(\*)
- (opcional) permitir al servidor TS CICS acceder a la biblioteca de carga para bidi y el Gestor de despliegue de aplicaciones
  - PERMIT 'FEK.SFEKLOAD' CLASS(DATASET) ACCESS(READ) ID(#cicsts)
- (opcional) permitir al servidor DB2 acceder a la biblioteca exec para el constructor de procedimientos almacenados DB2

- PERMIT 'FEK.SFEKPROC' CLASS(DATASET) ACCESS(READ) ID(#db2)
- activar perfiles de seguridad
  - SETROPTS GENERIC(DATASET) REFRESH

Al controlar el acceso de lectura (READ) a los conjuntos de datos del sistema, debe otorgar a los servidores y usuarios de Developer for System z el permiso para leer (READ) los conjuntos de datos siguientes:

- CEE.SCEERUN
- CEE.SCEERUN2
- CBC.SCLBDLL
- ISP.SISPLoad
- ISP.SISPLPA
- SYS1.LINKLIB
- SYS1.SIEALNKE
- REXX.V1R4M0.SEAGLPA

**Nota:** Si utiliza la biblioteca alternativa para el paquete de producto REXX, el nombre predeterminado de la biblioteca de tiempo de ejecución de REXX es REXX.\*.SEAGALT. en lugar de REXX.\*.SEAGLPA (como en el ejemplo anterior).

## Definir las tareas iniciadas de Developer for System z

Los siguientes mandatos RACF crean las tareas iniciadas JMON, RSED y LOCKD, con ID de usuario protegidos (STCJMON, STCRSE y STCLOCK, respectivamente) y el grupo STCGROUP asignado a ellas. Sustituya los espacios reservados #id-grupo e #id-usuario-\* por identificadores de OMVS válidos.

- ADDGROUP STCGROUP OMVS(GID(#id-grupo))  
DATA('GROUP WITH OMVS SEGMENT FOR STARTED TASKS')
- ADDUSER STCJMON DFLTGROUP(STCGROUP) NOPASSWORD NAME('RDZ - JES JOBMONITOR')  
OMVS(UID(#user-id-jmon) HOME(/tmp) PROGRAM(/bin/sh) NOASSIZEMAX  
NOTHEADSMAX)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
- ADDUSER STCRSE DFLTGROUP(STCGROUP) NOPASSWORD NAME('RDZ - RSE DAEMON')  
OMVS(UID(#user-id-rse) HOME(/tmp) PROGRAM(/bin/sh) ASSIZEMAX(2147483647)  
NOTHEADSMAX)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
- ADDUSER STCLOCK DFLTGROUP(STCGROUP) NOPASSWORD NAME('RDZ - DAEMON DE BLOQUEO')  
OMVS(UID(#id-usuario-bloqueo) HOME(/tmp) PROGRAM(/bin/sh) NOASSIZEMAX  
NOTHEADSMAX)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
- RDEFINE STARTED JMON.\* DATA('RDZ - JES JOBMONITOR')  
STDATA(USER(STCJMON) GROUP(STCGROUP) TRUSTED(NO))
- RDEFINE STARTED RSED.\* DATA('RDZ - RSE DAEMON')  
STDATA(USER(STCRSE) GROUP(STCGROUP) TRUSTED(NO))
- RDEFINE STARTED LOCKD.\* DATA('RDZ - DAEMON DE BLOQUEO')  
STDATA(USER(STCLOCK) GROUP(STCGROUP) TRUSTED(NO))
- SETROPTS RACLIST(STARTED) REFRESH

### Notas:

1. Asegúrese de que los IDs de usuario de las tareas iniciadas están protegidos especificando la palabra clave NOPASSWORD.
2. Asegúrese de que el servidor RSE tenga un uid OMVS exclusivo debido a los privilegios relacionados con z/OS UNIX otorgados a este uid.
3. El daemon RSE requiere un tamaño de espacio de direcciones grande (2GB) para funcionar adecuadamente. Debe establecer este valor en la variable ASSIZEMAX del segmento OMVS para el ID de usuario STCRSE. Ello asegura que

el daemon RSE conseguirá el tamaño de región necesario, independientemente de los cambios que se realicen en MAXASSIZE de SYS1.PARMLIB(BPXPRMxx).

4. RSE también requiere un gran número de hebras para funcionar adecuadamente. Puede establecer este límite en la variable THREADSMAX del segmento OMVS para el ID de usuario STCRSE. Ello asegura que el RSE conseguirá el límite de hebras necesario, independientemente de los cambios que se realicen en MAXTHREADS o MAXTHREADTASKS de SYS1.PARMLIB(BPXPRMxx). Consulte la sección Capítulo 13, “Consideraciones acerca de los ajustes”, en la página 213 para determinar el valor correcto para el límite de hebras.
5. El ID de usuario STCJMON es otro buen candidato para establecer THREADSMAX en el segmento OMVS, ya que el supervisor de trabajos JES utilice una hebra por cada conexión de cliente.

Puede que desee considerar la posibilidad de que el ID de usuario STCRSE sea restringido. Los usuarios con el atributo RESTRICTED no pueden acceder a recursos protegidos (MVS) a los que no tienen autorización de acceso específica.

ALTUSER STCRSE RESTRICTED

Para asegurarse de que los usuarios restringidos no obtengan acceso a los recursos del sistema de archivos de z/OS UNIX mediante los “otros” bits de permiso, debe definir el perfil RESTRICTED.FILESYS.ACCESS en la clase UNIXPRIV con UACC(NONE). Consulte la publicación *Security Server RACF Security Administrator's Guide* (SA22-7683) para obtener más información acerca de los ID de usuario restringidos.

**Atención:** Si utiliza ID de usuario restringidos, debe añadir explícitamente el permiso de acceso a los recursos con los mandatos **PERMIT** de TSO o **setfacl** de z/OS UNIX. Esto incluye los recursos en los que la documentación de Developer for System z utiliza UACC (como por ejemplo el perfil \*\* de la clase PROGRAM) o los basados en convenciones comunes de z/OS UNIX (como por ejemplo que todos los usuarios tengan permiso de lectura y ejecución sobre las bibliotecas de Java). Debe someter a prueba este procedimiento antes de activarlo en un sistema de producción.

## Definir la seguridad de mandatos JES

El Supervisor de trabajos JES emite todos los mandatos de operador de JES solicitados por un usuario por medio de una consola de EMCS ampliada (EMCS), cuyo nombre está controlado por la directiva CONSOLE\_NAME, tal como se describe en “FEJCNFG, archivo de configuración del supervisor de trabajos JES” en la página 27.

Los siguientes mandatos RACF de ejemplo otorgan a los usuarios de Developer for System z acceso condicional a un conjunto limitado de mandatos JES (Retener, Liberar, Cancelar y Depurar). Los usuarios sólo tienen permiso de ejecución si emiten los mandatos por medio del supervisor de trabajos JES. Sustituya el espacio reservado #console con el nombre de la consola.

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

### Nota:

- El uso de la consola está permitido si no hay ningún perfil MVS.MCSOPER.#console definido.

- La clase CONSOLE debe estar activa para que WHEN(CONSOLE(JMON)) funcione, pero no hay ninguna comprobación real de perfiles en la consola CONSOLE para las consolas de EMCS.
- No sustituya JMON con el nombre real de la consola en la cláusula WHEN(CONSOLE(JMON)). La palabra clave JMON representa la aplicación de punto de entrada, no el nombre de la consola.

**Atención:** 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.

La Tabla 27 y la Tabla 28 muestran los mandatos de operador emitidos para JES2 y JES3 y los perfiles de seguridad específicos que pueden utilizarse para protegerlos.

*Tabla 27. Mandatos de operador del Supervisor de trabajos JES2*

Acción	Mandato	Perfil OPERCMDS	Acceso necesario
Retener	\$Hx(idtrabajo) con x = {J, S o T}	jesname.MODIFYHOLD.BAT jesname.MODIFYHOLD.STC jesname.MODIFYHOLD.TSU	UPDATE
Liberar	\$Ax(idtrabajo) con x = {J, S o T}	jesname.MODIFYRELEASE.BAT jesname.MODIFYRELEASE.STC jesname.MODIFYRELEASE.TSU	UPDATE
Cancelar	\$Cx(idtrabajo) con x = {J, S o T}	jesname.CANCEL.BAT jesname.CANCEL.STC jesname.CANCEL.TSU	UPDATE
Purgar	\$Cx(idtrabajo),P con x = {J, S o T}	jesname.CANCEL.BAT jesname.CANCEL.STC jesname.CANCEL.TSU	UPDATE

*Tabla 28. Mandatos de operador del Supervisor de trabajos JES3*

Acción	Mandato	Perfil OPERCMDS	Acceso necesario
Retener	*F,J=idtrabajo,H	jesname.MODIFY.JOB	UPDATE
Liberar	*F,J=idtrabajo,R	jesname.MODIFY.JOB	UPDATE
Cancelar	*F,J=idtrabajo,C	jesname.MODIFY.JOB	UPDATE
Purgar	*F,J=idtrabajo,C	jesname.MODIFY.JOB	UPDATE

**Nota:**

- Los mandatos del operador JES Retener, Liberar, Cancelar y Depurar, y el mandato Mostrar JCL sólo pueden ejecutarse en los archivos de spool propiedad del ID de usuario cliente, a menos que se especifique LIMIT\_COMMANDS= con el valor LIMITED, o se especifique NOLIMIT en el archivo de configuración del Supervisor de trabajos JES. Consulte la sección “Acciones en trabajos - limitaciones de destino” en la página 166 para obtener más información acerca de este punto.
- Los usuarios pueden examinar cualquier archivos de spool, a menos que se haya definido LIMIT\_VIEW=USERID en el archivo de configuración del Supervisor de trabajos JES. Consulte la sección “Acceso a los archivos de spool” en la página 169 para obtener más información acerca de este punto.
- Aunque no posean autorización sobre estos mandatos de operador, los usuarios todavía pueden someter trabajos y leer la salida de los trabajos por medio del Supervisor de trabajos JES, en caso de que dispongan de la

autorización suficiente sobre los perfiles posibles que protegen estos recursos (como los de las clases JESINPUT, JESJOBS y JESSPOOL).

El software de seguridad impide la asunción de identidad del servidor Supervisor de trabajos JES creando una consola JMON desde una sesión TSO. Aunque la consola se puede crear, el punto de entrada es distinto (supervisor de trabajos JES versus TSO). Los mandatos JES emitidos desde esta consola fallarán la comprobación de seguridad, si la seguridad está configurada según se describe en esta publicación.

## Definir RSE como servidor z/OS UNIX seguro

RSE requiere acceso de actualización (UPDATE) al perfil BPX.SERVER para crear/suprimir el entorno de seguridad de la hebra del cliente. Si este perfil no está definido, RSE requiere el UID(0).

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

**Atención:** Definir el perfil BPX.SERVER hace que z/OS UNIX como un todo cambie de la seguridad a nivel de UNIX a la seguridad a nivel de z/OS UNIX, la cual es más segura. Ello puede afectar a otras operaciones y aplicaciones de z/OS UNIX. Debe someter a prueba este procedimiento antes de activarlo en un sistema de producción. Consulte el manual *UNIX System Services Planning* (GA22-7800) para obtener más información los distintos niveles de seguridad.

## Definir las bibliotecas controladas por programa MVS para RSE

Los servidores con autorización sobre BPX.SERVER deben ejecutarse en un entorno limpio controlado por programa. Esto implica que todos los programas a los que llama RSE también deben estar controlados por programa. Para las bibliotecas de carga MVS, el control de programa se gestiona mediante el software de seguridad.

RSE utiliza la biblioteca de carga del entorno de ejecución Language Environment (CEE.SCEERUN\*) y de la Pasarela de cliente TSO/ISPF de ISPF (ISP.SISPLPA) del sistema.

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

**Nota:** No utilice el perfil \*\* si ya tiene un perfil \* en la clase PROGRAM. Oscurece y complica la vía de acceso de búsqueda utilizada por el software de seguridad. En este caso, debe fusionar las definiciones \* existentes y las definiciones \*\* nuevas. IBM recomienda utilizar el perfil \*\*, como se describe en la publicación *Security Server RACF Security Administrator's Guide* (SA22-7683).

Las siguientes bibliotecas adicionales (prerrequisito) deben estar controladas por programa para dar soporte a la utilización de servicios opcionales. Esta lista no incluye los conjuntos de datos específicos de un producto con el que interactúa Developer for System z, como IBM Debug Tool.

- Biblioteca de tiempo de ejecución REXX alternativa (para SCLM Developer Toolkit)
  - REXX.\*.SEAGALT
- Biblioteca de carga del sistema (para cifrado SSL)
  - SYS1.SIEALNKE
- Biblioteca de carga de escucha de gestor de archivos (para integración de gestor de archivos)
  - FMN.SFMNMODA

**Nota:** Las bibliotecas diseñadas para colocación en LPA también requieren autorizaciones de control de programa si se accede a ellas por medio de LINKLIST o STEPLIB. Esta publicación documenta la utilización de las siguientes bibliotecas de LPA:

- ISPF (para Pasarela de cliente TSO/ISPF)
  - ISP.SISPLPA
- Biblioteca de tiempo de ejecución REXX (para SCLM Developer Toolkit)
  - REXX.\*.SEAGLPA
- Developer for System z (para CARMA)
  - FEK.SFEKLPA

## Definir la protección de aplicaciones para RSE

Durante el inicio de sesión de clientes, el daemon RSE verifica que un usuario pueda utilizar la aplicación.

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

**Nota:**

Tal como se describe más detalladamente en “Definir el soporte de PassTicket para RSE”, RSE soporta el uso de un ID de aplicación distinto de FEKAPPL. La definición de clase APPL debe coincidir con el ID de aplicación real que utiliza RSE.

**Atención:** La solicitud de conexión del cliente fallará si el perfil de la aplicación no está definido, o bien cuando el usuario no tiene acceso de lectura (READ) al perfil.

## Definir el soporte de PassTicket para RSE

La contraseña del cliente (u otras formas de identificación, como un certificado X.509) sólo se utiliza para verificar su identidad durante la conexión. Después de eso, se utilizan Pases (PassTickets) para mantener la seguridad de las hebras.

Los PassTickets son contraseñas generadas por el sistema con un tiempo de vida aproximado de 10 minutos. Los PassTickets generados se basan en una clave secreta. Esta clave es un número de 64 bits (16 caracteres hexadecimales). En los mandatos RACF de ejemplo que figuran a continuación, sustituya el espacio reservado key16 por una serie de 16 caracteres hexadecimales proporcionada por el usuario (caracteres 0-9 y A-F).

- RDEFINE PTKTDATA FEKAPPL UACC(NONE) SSIGNON(KEYMASKED(key16))  
APPLDATA('NO REPLAY PROTECTION – DO NOT CHANGE')  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')

- RDEFINE PTKTDATA IRRPTAUTH.FEKAPPL.\* UACC(NONE)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
- PERMIT IRRPTAUTH.FEKAPPL.\* CLASS(PTKTDATA) ACCESS(UPDATE) ID(STCRSE)
- SETROPTS RACLIST(PTKTDATA) REFRESH

RSE soporta el uso de un ID de aplicación distinto de FEKAPPL. Descomente y personalice la opción "APPLID=FEKAPPL" del archivo `rzed.envvars` para activarla, como se indica en la sección "Definir parámetros de inicio Java adicionales con `_RSE_JAVAOPTS`" en la página 42. Las definiciones de clase PTKTDATA deben coincidir con el ID de aplicación real utilizado por RSE.

No debe utilizar OMVSAPPL como ID de aplicación porque abrirá la clave secreta a la mayoría de aplicaciones z/OS UNIX. Tampoco debe utilizar el ID de aplicación MVS predeterminado, que es MVS seguido por el ID SMF del sistema, porque esto abrirá la clave secreta a la mayoría de aplicaciones MVS (incluyendo trabajos por lotes de usuarios).

**Nota:**

- Si la clase PTKTDATA ya está definida, verifique que lo está como clase genérica antes de crear los perfiles enumerados a continuación. El soporte para los caracteres genéricos de la clase PTKTDATA es nuevo del release 1.7 de z/OS, con la introducción de una interfaz de Java a PassTickets.
- Sustituya el comodín (\*) de la definición IRRPTAUTH.FEKAPPL.\* con una máscara de ID de usuario válida para limitar los ID de usuario para los que RSE puede generar un PassTicket.
- Dependiendo de sus valores RACF, es posible que el usuario que ha definido un perfil aparezca también en la lista de acceso para el perfil en cuestión. Es aconsejable que elimine este permiso para los perfiles PTKTDATA.
- RSE y el supervisor de trabajos JES deben tener el mismo ID de aplicación para que el supervisor de trabajos JES pueda evaluar los PassTickets presentados por RSE.
- Si el sistema tiene un producto criptográfico instalado y disponible, puede cifrar la clave de la aplicación de inicio de sesión seguro para obtener más protección. Para ello, utilice la palabra clave KEYENCRYPTED, en lugar de KEYMASKED. Consulte el manual *Security Server RACF Security Administrator's Guide* (SA22-7683) para obtener más información sobre este tema.

<p><b>Atención:</b> La solicitud de conexión del cliente fallará si los PassTickets no están configurados correctamente.</p>
--

## Definir los archivos controlados por programa z/OS UNIX para RSE

Los servidores con autorización sobre BPX.SERVER deben ejecutarse en un entorno limpio controlado por programa. Esto implica que todos los programas a los que llama RSE también deben estar controlados por programa. Para los archivos z/OS UNIX, el control de programa se gestiona mediante el mandato `extattr`. Para ejecutar este mandato, necesita acceso de lectura (READ) a BPX.FILEATTR.PROGCTL en la clase FACILITY o tener el UID(0).

El servidor RSE utiliza la biblioteca compartida Java de RACF(`/usr/lib/libIRRRacf.so`).

- `extattr +p /usr/lib/libIRRRacf.so`

**Nota:**

- A partir de z/OS 1.9, /usr/lib/libIRRRacf.so se instala de forma controlada por programa durante la instalación de RACF SMP/E.
- A partir de z/OS 1.10, /usr/lib/libIRRRacf.so forma parte de SAF, que se suministra con el producto base z/OS, por lo que también está disponible para los clientes no RACF.
- La configuración puede ser diferente si utiliza un producto distinto de RACF. Consulte la documentación de su producto de seguridad para obtener más información.
- La instalación SMP/E de Developer for System z establece el bit de control de programa para los programas internos de RSE.
- Utilice el mandato de z/OS UNIX **ls -Eog** para visualizar el estado actual del bit de control de programa (el archivo está controlado por programa si aparece la letra **p** en la segunda serie).

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

## Verificar valores de seguridad

Utilice los siguientes mandatos de ejemplo para visualizar los resultados de las personalizaciones relacionadas con la seguridad.

- Valores y clases de seguridad
  - `SETOPTS LIST`
- Segmento OMVS para usuarios
  - `LISTUSER #userid NORACF OMVS`
  - `LISTGRP #group-name NORACF OMVS`
- Perfiles de conjunto de datos
  - `LISTGRP FEK`
  - `LISTDSO PREFIX(FEK) ALL`
- Tareas iniciadas
  - `LISTGRP STCGROUP OMVS`
  - `LISTUSER STCMON OMVS`
  - `LISTUSER STCRSE OMVS`
  - `LISTUSER STCLOCK OMVS`
  - `RLIST STARTED JMON.* ALL STDATA`
  - `RLIST STARTED RSED.* ALL STDATA`
  - `RLIST STARTED LOCKD.* ALL STDATA`
- Seguridad de mandatos JES
  - `RLIST CONSOLE JMON ALL`
  - `RLIST OPERCMDS MVS.MCSOPER.JMON ALL`
  - `RLIST OPERCMDS JES%.* ALL`
- RSE como servidor z/OS UNIX seguro
  - `RLIST FACILITY BPX.SERVER ALL`
- Bibliotecas controladas por programa MVS para RSE
  - `RLIST PROGRAM ** ALL`
- Protección de aplicaciones para el RSE
  - `RLIST APPL FEKAPPL ALL`

- Soporte de PassTicket para RSE
  - RLIST PTKTDATA FEKAPPL ALL SSIGNON
  - RLIST PTKTDATA IRRPTAUTH.FEKAPPL.\* ALL
- Archivos controlados por programa z/OS UNIX para RSE
  - ls -E /usr/lib/libIRRracf.so

## Capítulo 11. Qué es Developer for System z

El host de Developer for System z está formado por varios componentes que interactúan para proporcionar al cliente acceso a los servicios y datos del host. Comprender el diseño de estos componentes puede ayudarle a tomar las decisiones de configuración correctas.

En este capítulo se tratan estos temas:

- “Visión general de los componentes”
- “RSE como aplicación Java” en la página 193
- “Propietarios de tareas” en la página 194
- “Flujo de conexión” en la página 196
- “Daemon de bloqueo” en la página 198
- “Estructura de directorios de z/OS UNIX” en la página 200

### Visión general de los componentes

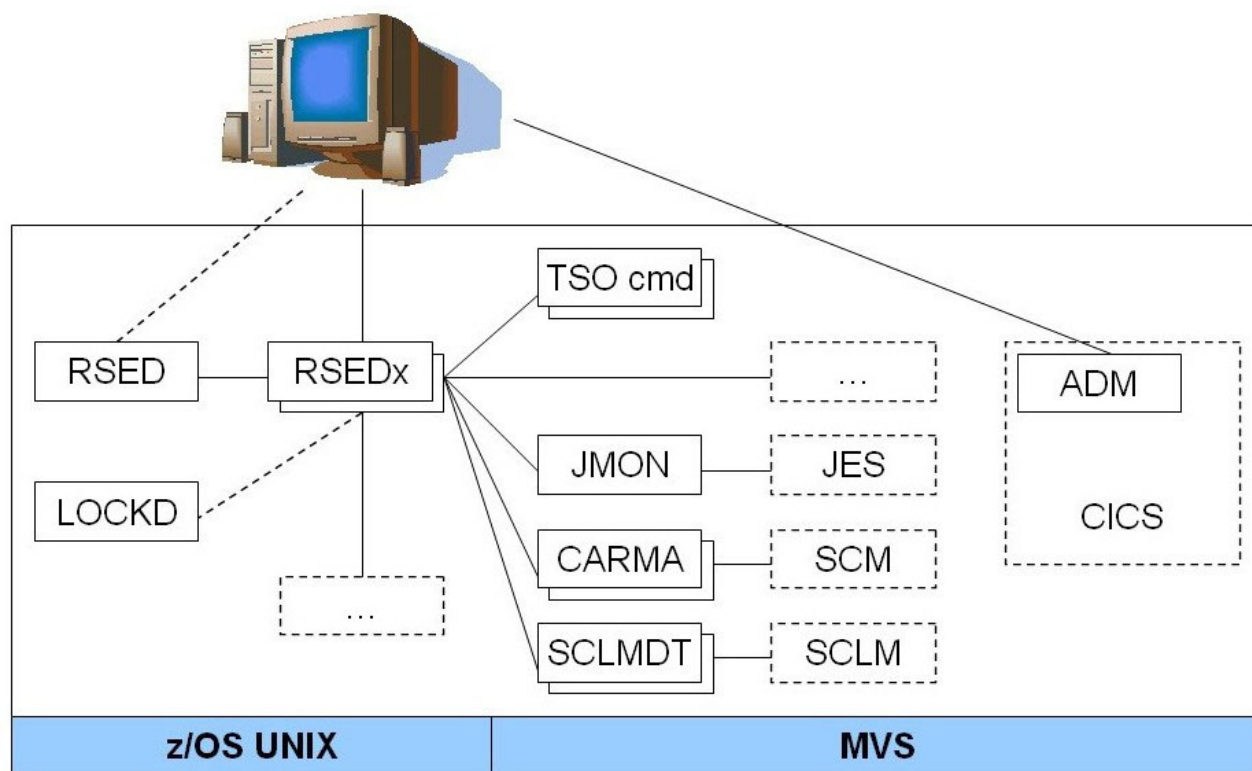


Figura 41. Visión general de los componentes

Figura 41 muestra una visión generalizada del diseño de Developer for System z en el sistema host.

- El Explorador de sistemas remotos (RSE) proporciona servicios del núcleo como los de conectar el cliente al host e iniciar otros servidores para servicios específicos. El RSE consta de dos entidades lógicas:

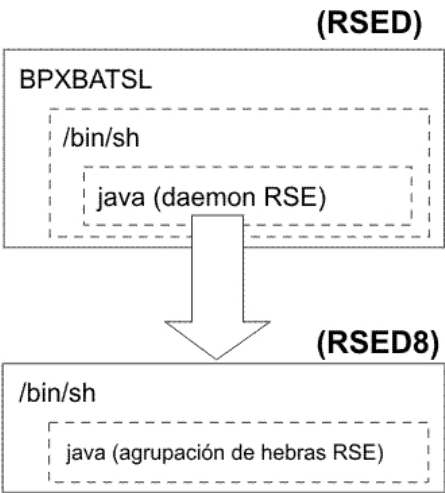
- El daemon RSE (RSED), que gestiona la configuración de conexiones. El daemon RSE es responsable de la ejecución en modalidad de servidor único. Para ello, el daemon RSE crea uno o varios procesos hijo conocidos como agrupaciones de hebras RSE (RSEDx).
- El servidor RSE, que maneja las solicitudes de clientes individuales. Un servidor RSE está activo como hebra dentro de una agrupación de hebras RSE.
- El daemon de bloqueo (LOCKD) proporciona servicios de rastreo para bloqueos de conjuntos de datos.
- El Servicio de mandatos TSO (TSO cmd) proporciona una interfaz de tipo por lotes para los mandatos TSO y ISPF.
- El Supervisor de trabajos JES (JMON) suministra todos los servicios relacionados con JES.
- El CARMA (Common Access Repository Manager) proporciona una interfaz para interactuar con SCM (Software Configuration Manager), tales como el CA Endevor.
- SCLM Developer Toolkit (SCLMDT) proporciona una interfaz para mejorar e interactuar con SCLM.
- El Gestor de despliegue de aplicaciones (ADM) proporciona varios servicios relacionados con CICS.
- Hay más servicios disponibles, proporcionados por el propio Developer for System z o por el software correquisito.

La descripción del párrafo y la lista anteriores muestra el rol central asignado al RSE. Salvo algunas excepciones, toda la comunicación de cliente va a través del RSE. Ello permite una configuración de la red de seguridad sencilla, ya que únicamente se utiliza un conjunto limitado de puertos para la comunicación cliente-host.

Para gestionar las conexiones y cargas de trabajo de los clientes, RSE está formado por un espacio de direcciones de daemon, que controla los espacios de direcciones de agrupaciones de hebras. El daemon actúa como punto focal a efectos de conexión y gestión, mientras que las agrupaciones de hebras procesan las cargas de trabajo del cliente. Basándose en los valores definidos en el archivo de configuración `rsed.envvars` y en la cantidad de conexiones de cliente reales, el daemon puede iniciar varios espacios de direcciones de agrupaciones de hebras.

# RSE como aplicación Java

## Procesos z/OS UNIX



## Utilización del almacenamiento Java

Sistema - compartido
Sistema - privado
Código (z/OS UNIX, Java, RSE)
Almacenamiento dinámico Java
No utilizado

NOM	TRAB	Estado	PID	PPID	Mandato
RSED	FILE SYS	KERNEL WAIT	50331904	1	BPXBATSL
RSED		WAITING FOR CHILD	67109114	50331904	/bin/sh...
RSED	FILE SYS	KERNEL WAIT	50331949	67109114	java...
RSED8		WAITING FOR CHILD	307	50331949	/bin/sh...
RSED8	FILE SYS	KERNAL WAIT	308	307	java...

Figura 42. RSE como aplicación Java

Figura 42 muestra una vista básica del uso de recursos (procesos y almacenamiento) por RSE.

RSE es una aplicación Java, lo que significa que está activo en el entorno z/OS UNIX. Ello permite establecer puertos de forma sencilla a otras plataformas de host y una comunicación directa con el cliente de Developer for System z, que también es una aplicación Java (basada en la infraestructura de Eclipse). Por ello, tener un conocimiento básico de cómo funcionan z/OS UNIX y Java es de gran ayuda para comprender Developer for System z.

En z/OS UNIX un programa se ejecuta en un proceso, identificado por un PID (ID de proceso). Cada programa está activo en su propio proceso, de manera que el hecho de invocar otro programa crea un proceso nuevo. Se hace referencia al proceso que ha iniciado un proceso con un PPID (PID padre), y el proceso nuevo se denomina proceso hijo. El proceso hijo se puede ejecutar en el mismo espacio de direcciones, o bien se puede engendrar (crear) en un espacio de direcciones nuevo. Un proceso nuevo que se ejecuta en el mismo espacio de direcciones puede compararse con la ejecución de un mandato en TSO; mientras que engendrar uno en un espacio de direcciones nuevo es similar a someter un trabajo por lotes.

Tenga un proceso puede tener una sola o varias hebras. En una aplicación de varias hebras (como RSE), las distintas hebras compiten por los recursos del sistema, como si fueran espacios de direcciones separados (con menos sobrecarga).

Al correlacionar esta información de proceso al ejemplo de RSE de Figura 42, obtenemos este flujo:

1. Cuando se inicia la tarea RSED, esta ejecuta BPXBATSL, que invoca z/OS UNIX y crea un entorno de shell – PID 50331904.
2. En este proceso se ejecuta el script de shell `rsed.sh` shell, que se ejecuta en un proceso independiente (`/bin/sh`) – PID 67109114.
3. El script de shell establece las variables de entorno definidas en `rsed.envvars` y ejecuta Java con los parámetros necesarios para iniciar el daemon RSE – PID 50331949.
4. El daemon RSE generará un shell nuevo en un proceso hijo (RSED8) – PID 307.
5. En este shell, se establecen las variables de entorno definidas en `rsed.envvars` y se ejecuta Java con los parámetros necesarios para iniciar la agrupación de hebras RSE – PID 308.

Las aplicaciones Java, tales como RSE, no asignan almacenamiento directamente, sino que utilizan los servicios de gestión de memorias de Java. Estos servicios, como la asignación de almacenamiento, la liberación de almacenamiento, y la recogida de basura, funcionan dentro de los límites del almacenamiento dinámico de Java. El tamaño mínimo y máximo del almacenamiento dinámico se define (ya sea implícita o explícitamente) durante el inicio de Java.

Ello implica que obtener el máximo del tamaño de espacio de direcciones disponible es un acto de equilibrio que consiste en definir un tamaño de almacenamiento dinámico grande y dejar suficiente espacio para que z/OS almacene una cantidad variable de bloques de control del sistema (que depende del número de hebras activas).

---

## Propietarios de tareas

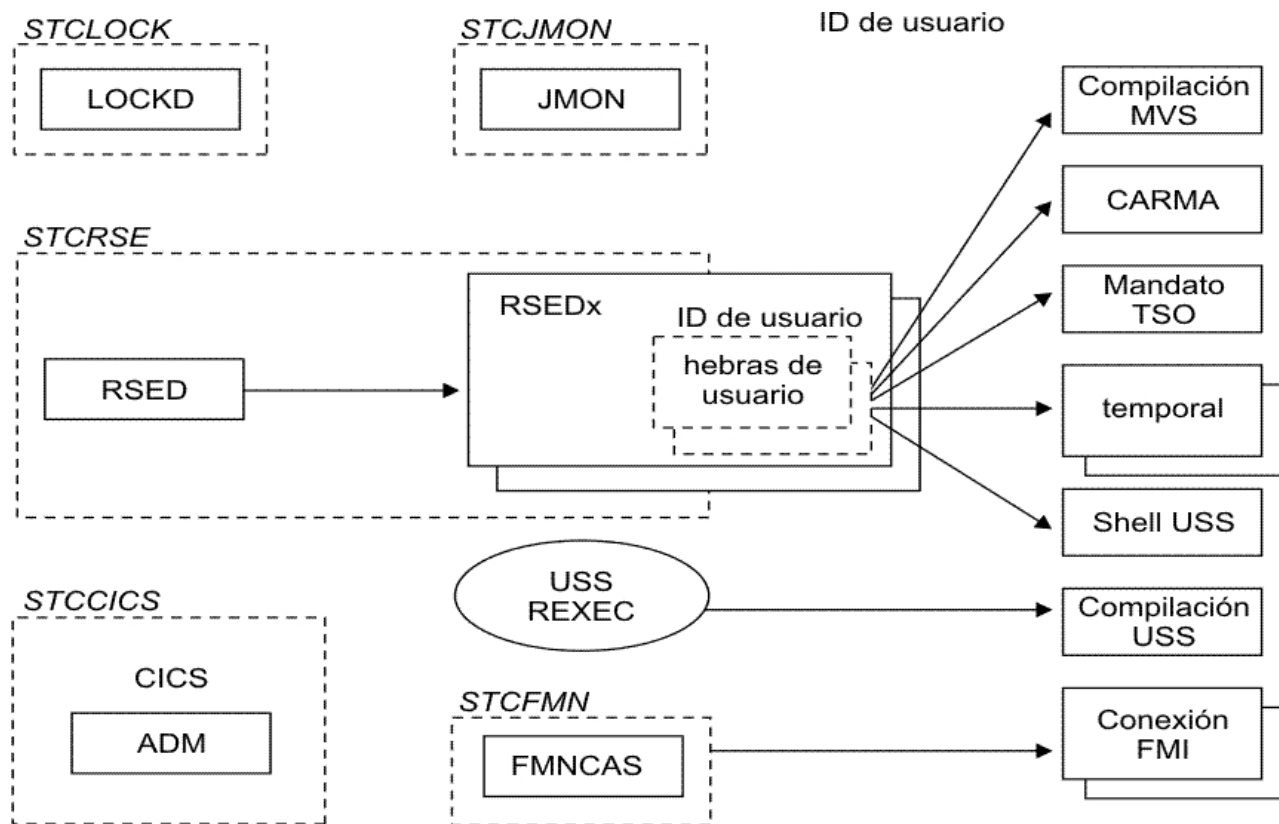


Figura 43. Propietarios de tareas

Figura 43 muestra una visión general básica del propietario de las credenciales de seguridad utilizadas para varias tareas de Developer for System z.

La propiedad de una tarea se puede dividir en dos secciones. Las tareas iniciadas son propiedad del ID de usuario asignado a la tarea iniciada en el software de seguridad. El resto de tareas con las agrupaciones de hebras RSE (RSEDx) como excepción son propiedad del ID de usuario del cliente.

Figura 43 muestra las tareas iniciadas de Developer for system z (LOCKD, JMON y RSED) y las tareas iniciadas de ejemplo y los servicios del sistema con los que Developer for System z se comunica. Application Deployment Manager (ADM) está activo dentro de una región CICS. FMNCAS es la tarea iniciada por el Gestor de archivos. El código USS REXEC representa el servicio z/OS UNIX REXEC (o SSH).

El daemon RSE (RSED) crea uno o varios espacios de direcciones de agrupaciones de hebras RSE (RSEDx) para procesar las peticiones de cliente. Cada agrupación de hebras RSE soporta varios clientes y es propiedad del mismo usuario que el daemon RSE. Cada cliente tiene sus propias hebras dentro de una agrupación de hebras y estas hebras son propiedad del ID de usuario de cliente.

Según las acciones realizadas por el cliente, se pueden iniciar uno o varios espacios de direcciones, todos propiedad del ID de usuario cliente para realizar la acción solicitada. Estos espacios de direcciones pueden estar en un trabajo por lotes MVS, una transacción APPC o un proceso hijo z/OS UNIX. Tenga en cuenta que un proceso hijo z/OS UNIX está activo en un iniciador z/OS UNIX (BPXAS) y que el proceso hijo aparece como una tarea iniciada en JES.

La mayoría de las veces es una hebra de usuario en una agrupación de hebras la que desencadena la creación de estos espacios de direcciones, ya sea directamente o a través de servicios del sistema como por ejemplo ISPF. Sin embargo, el espacio de direcciones también lo puede crear un tercero. Por ejemplo, el Gestor de archivos iniciará un espacio de direcciones nuevo para cada conjunto de datos (o miembro) que debe procesar en nombre de Developer for System z. z/OS UNIX REXEC o SSH están implicados cuando se inician construcciones nuevas en z/OS UNIX.

Los espacios de direcciones específicos del usuario terminan cuando finaliza la tarea o cuando caduca el temporizador de inactividad. Las tareas iniciadas permanecen activas. Los espacios de direcciones que aparecen en la lista de Figura 43 en la página 195 permanecen en el sistema lo suficiente para ser visibles. Sin embargo, debe tener en cuenta que debido al diseño de z/OS UNIX, también hay varios espacios de direcciones temporales de vida breve.

## Flujo de conexión

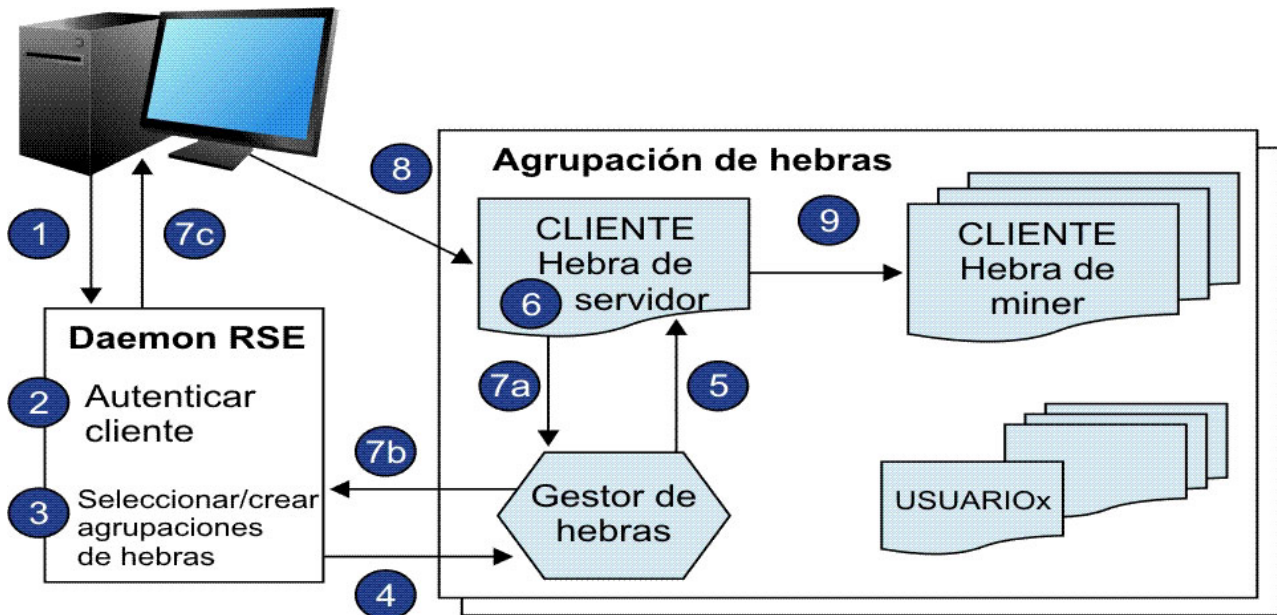


Figura 44. Flujo de conexión

Figura 44 muestra una visión general esquemática de la forma en que un cliente se conecta al host mediante Developer for System z. Además, explica brevemente cómo se utilizan los PassTickets.

1. El cliente inicia sesión en el daemon (puerto 4035).
2. El daemon RSE autentica el cliente mediante las credenciales presentadas por éste.
3. El daemon RSE selecciona una agrupación de hebras existente o bien inicia una agrupación en caso de que todas estén completas.
4. El daemon RSE pasa el ID de usuario del cliente a la agrupación de hebras.
5. La agrupación de hebras crea una hebra de servidor RSE específica del cliente, utilizando el ID de usuario del cliente y un PassTicket para la autenticación.
6. La hebra de servidor de cliente se enlaza a un puerto para la futura comunicación con el cliente.

7. La hebra de servidor de cliente devuelve el número de puerto para que el cliente se conecte.
8. El cliente se desconecta del daemon RSE y se conecta al número de puerto proporcionado.
9. La hebra de servidor de cliente inicia otras hebras específicas del cliente (miners), utilizando el ID de usuario del cliente y un PassTicket para la autenticación. Estas hebras proporcionan los servicios específicos del cliente que el cliente requiere.

La descripción anterior muestra el diseño orientado a hebras de RSE. En lugar de iniciar un espacio de direcciones por usuario, un único espacio de direcciones de agrupaciones de hebras proporciona servicio a varios usuarios. Dentro de la agrupación de hebras, cada miner (un servicio específico del usuario) se activa en su propia hebra con el contexto de seguridad del usuario que tiene asignado, asegurando así una configuración segura. El diseño acomoda un mayor número de usuarios con un uso de recursos limitado, pero conlleva que cada cliente utilice varias hebras (16 o más, dependiendo de las tareas realizadas).

Desde un punto de vista de red, Developer for system z actúa de forma similar al FTP en modalidad pasiva. El cliente se conecta a un punto focal (daemon RSE), descarta la conexión y se reconecta a un número de puerto proporcionado por el punto focal. La siguiente lógica controla la selección del puerto que se utiliza para la segunda conexión:

1. Si el cliente ha especificado un número de puerto distinto de cero en la pestaña de propiedades de subsistema, el servidor RSE utilizará ese número de puerto para el enlace. Si este puerto no está disponible, la conexión falla.
2. Si se especifica `_RSE_PORTRANGE` en `rsed.envvars`, el servidor RSE se enlazará a un puerto de este rango. Si no hay ningún puerto disponible, la conexión falla. Tenga en cuenta que el servidor RSE no necesita el puerto exclusivamente para la duración de la conexión de cliente. Está sólo en el lapso de tiempo entre el enlace (servidor) y la conexión (cliente) que ningún otro servidor RSE puede enlazar al puerto.
3. Si no establece ninguna limitación, el servidor RSE se enlazará al puerto 0. El resultado es que TCP/IP elige el número de puerto.

El uso de PassTickets para todos los servicios de z/OS que requieren autenticación permite a Developer for System z invocar estos servicios cuando sea necesario sin tener que almacenar la contraseña ni solicitarle al usuario continuamente que la introduzca. El uso de PassTickets para todos los servicios de z/OS también permite métodos de autenticación alternativos durante el inicio de sesión, como las contraseñas para una sola vez y los certificados X.509.

## Daemon de bloqueo

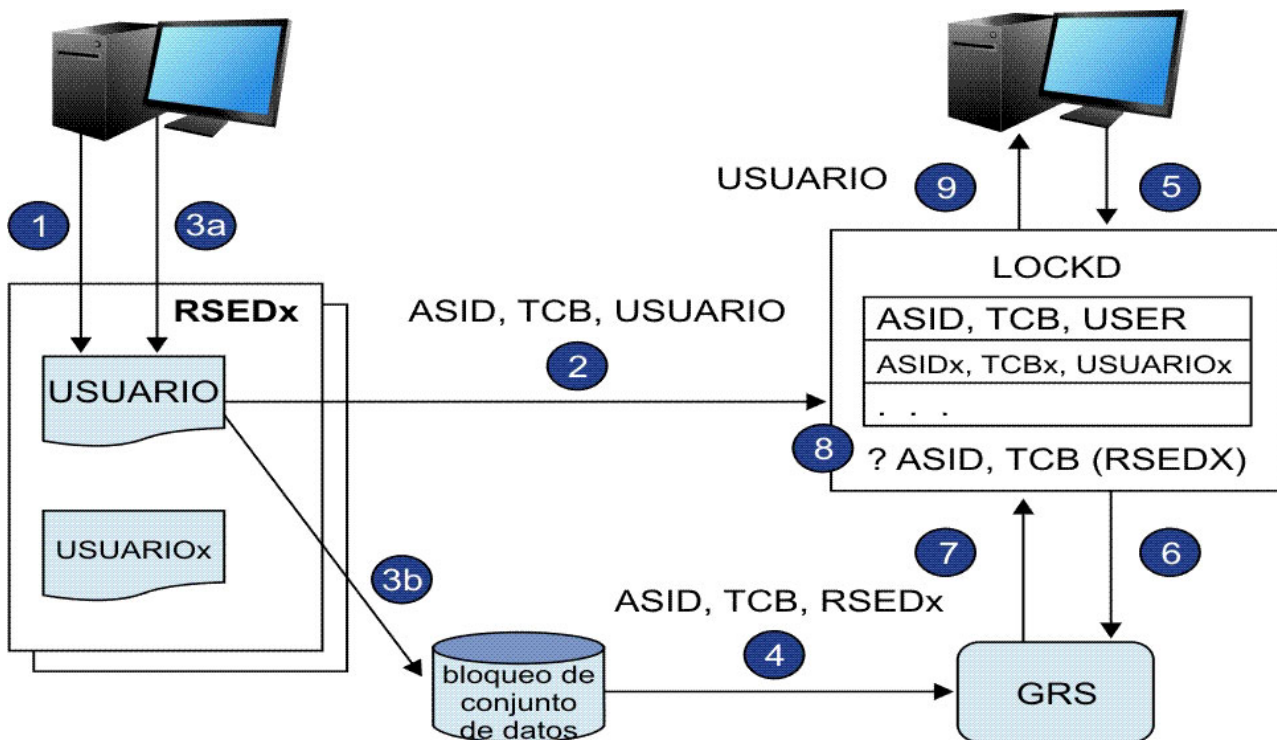


Figura 45. Flujo de daemon de bloqueo

Figura 45 muestra una visión general esquemática de la forma en que el daemon de bloqueo determina qué cliente de Developer for System z es propietario de un bloqueo de conjunto de datos.

1. El cliente inicia sesión, lo que crea una hebra de servidor RSE específica del usuario (USER) dentro de una agrupación de hebras (RSEDx).
2. El servidor RSE registra un usuario recién conectado con el daemon de bloqueo. La información de registro contiene el identificador de espacios de direcciones (que es el ASID de la agrupación de hebras), el identificador del bloque de control de tareas (TCB) (específico del usuario), y el ID de usuario.
3. El cliente abre un conjunto de datos en edición, que especifica al servidor RSE que obtenga un bloqueo exclusivo en el conjunto de datos.
4. El sistema registra el ASID, TCB y el nombre de tarea (RSEDx) del solicitante como parte del proceso de bloqueo. Esta información se almacena en las colas de serialización de recursos globales (GRS).
5. Un operador (o el servidor RSE en nombre de un cliente) consulta el daemon de bloqueo en busca del estado de bloqueo del conjunto de datos.
6. El daemon de bloqueo explora las colas de GRS para saber si el conjunto de datos está bloqueado.
7. El daemon recupera el ASID, TCB y el nombre de tarea del propietario del bloqueo.
8. El ASID y TCB recuperados se comparan con las combinaciones de ASID y TCB de los clientes registrados.
9. El ID de usuario del cliente relacionado se devuelve al solicitante cuando se encuentra una coincidencia. De lo contrario, el nombre de tarea recuperado de la GRS se devuelve.

Con una configuración de Developer for System z con un solo servidor, donde hay varios usuarios asignados a un único espacio de direcciones de agrupaciones de hebras, z/OS pierde la capacidad de rastrear quién es propietario de un bloqueo en un conjunto de daots o un miembro. Los mandatos del sistema se detienen a nivel de espacio de dirección, que es la agrupación de hebras.

Para solucionar este problema, Developer for System z proporciona el daemon de bloqueo. El daemon de bloqueo puede rastrear todos los bloqueos de conjuntos de datos/miembros realizador por los usuarios de RSE, así como los bloqueos realizados por otros productos, como ISPF.

El servidor RSE registra un usuario recién conectado con el daemon de bloqueo. La información de registro contiene el identificador de espacios de direcciones (que es el ASID de la agrupación de hebras), el ID del bloque de control de tareas (TCB) (específico del usuario), y el ID de usuario.

Tenga en cuenta que el registro se realiza únicamente en el momento de conectarse, de manera que todos los usuarios de RSE que estaban activos antes de que se iniciara (o se reiniciara) el daemon de bloqueo no quedarán registrados.

Cuando el daemon de bloqueo recibe una consulta de conjunto de datos (a través de un mandato de operador modify query o desde el cliente a través del servidor RSE), el daemon examina las colas de serialización de recursos globales (GRS) del sistema. Si el ASID y TCB coinciden con los de un usuario registrado, se devuelve el ID de usuario como propietario del bloqueo. De lo contrario, se devuelve el nombre de trabajo/ID de usuario asociado al ASID como propietario del bloqueo.

En caso de que el registro falle, aparece un mensaje de consola (FEK513W) con la información de registro. Ello permite a un operador hacer coincidir los valores con la salida de un mandato de operador **DISPLAY GRS,RES=(\*,dataset\*)** a fin de encontrar el propietario del bloqueo.

**Nota:** Los registros correctos también se enumeran en STDOUT de DD del servidor si log\_level se establece como 2. Esto es útil para realizar la correlación manual de los registros satisfactorios eliminados tras un reinicio del daemon de bloqueo.

## Liberar un bloqueo

En circunstancias normales, un conjunto de datos o un miembro está bloqueado cuando un cliente lo abre en modalidad de edición, y este se libera cuando el cliente cierra la sesión de edición.

Algunas condiciones de error pueden provocar que este mecanismo no funcione tal como debe. En este caso, se puede cancelar el usuario que está manteniendo el bloqueo mediante el mandato de operador **modify cancel** de RSE, tal como se describe en Capítulo 8, “Mandatos de operador”, en la página 125. Los bloqueos de conjuntos de datos activos de este usuario se liberan durante el proceso.

## Estructura de directorios de z/OS UNIX

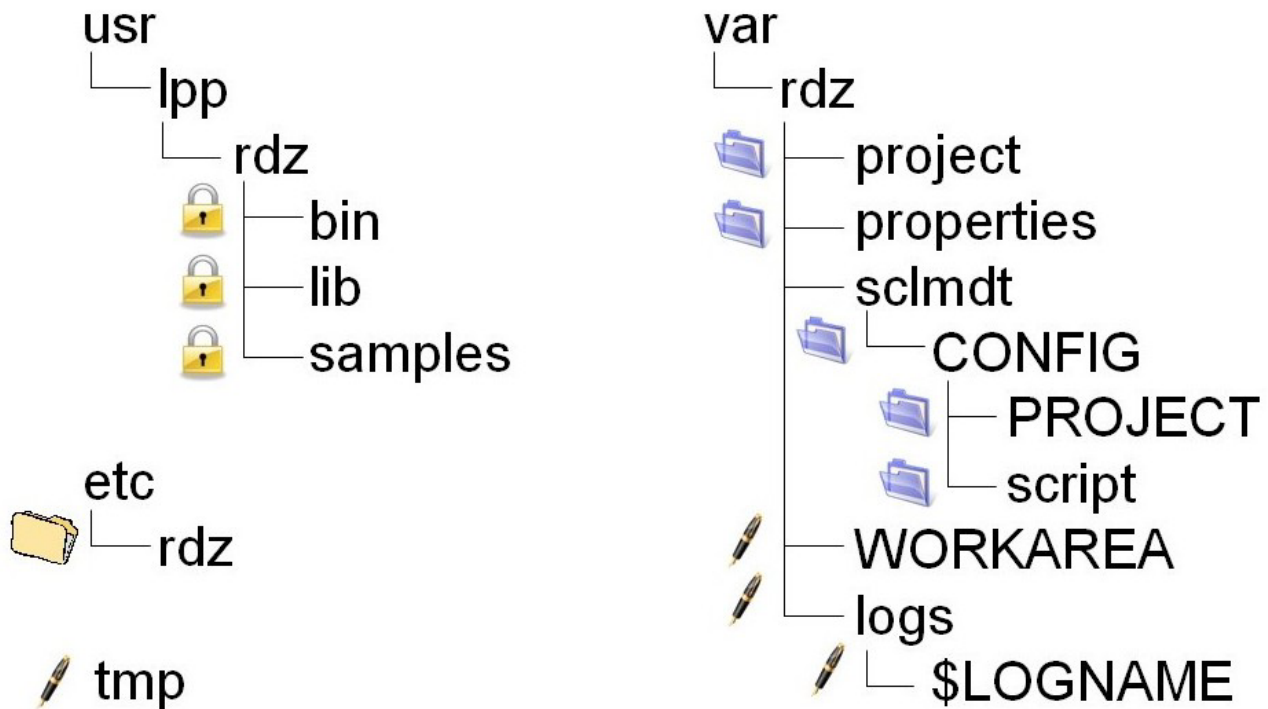


Figura 46. Estructura de directorios de z/OS UNIX

Figura 46 muestra una visión general de los directorios de z/OS UNIX utilizados por Developer for System z. La lista siguiente describe cada directorio tocado por Developer for System z, cómo se puede cambiar la ubicación y quién mantiene los datos que contienen.

- `/usr/lpp/rdz/` es la vía de acceso de raíz para el código de producto de Developer for System z. La ubicación real se especifica en las tareas iniciadas RSED y LOCKD (variable HOME). SMP/E mantiene los archivos que contienen.
- `/etc/rdz/` contiene los archivos de configuración relacionados con RSE y el miner. La ubicación real se especifica en las tareas iniciadas RSED y LOCKD (variable CNFG). El programador del sistema mantiene los archivos que contienen.
- `/var/rdz/sclmdt/CONFIG/` contiene los archivos de configuración SCLMDT generales. La ubicación real se especifica en `rsed.envvars` (variable SCLMDT\_CONF\_HOME). El administrador de SCLM mantiene los archivos que contienen.
- `/var/rdz/sclmdt/CONFIG/PROJECT/` contiene los archivos de configuración de proyectos SCLMDT. La ubicación real se especifica en `rsed.envvars` (variable SCLMDT\_CONF\_HOME). El administrador de SCLM mantiene los archivos que contienen.
- `/var/rdz/sclmdt/CONFIG/script/` contiene los scripts relacionados con SCLMDT que pueden utilizar otros productos. La ubicación actual no se especifica en ningún sitio. El administrador de SCLM mantiene los archivos que contienen.
- `/var/rdz/projects/` contiene los archivos de definición de proyectos basados en host. La ubicación real se especifica en `projectcfg.properties` (variable PROJECT-HOME). El gestor de proyectos o el desarrollador principal mantiene los archivos que contienen.

- `/var/rdz/properties/` contiene los grupos de propiedades basadas en host. La ubicación real se especifica en `propertiescfg.properties` (variables `PROPERTY-GROUP` y `DEFAULT-VALUES`). El gestor de proyectos o el desarrollador principal mantiene los archivos que contienen.
- `/var/rdz/logs/` contiene las anotaciones del daemon RSE y los servidores de agrupaciones de hebras RSE. La ubicación real se especifica en `rsed.envvars` (variable `daemon.log`). RSE mantiene los archivos que contienen.
- `/var/rdz/logs/$LOGNAME/` contiene las anotaciones específicas del usuario del servidor RSE y los miners. La ubicación real se especifica en `rsed.envvars` (variables `user.log` y `DSTORE_LOG_DIRECTORY`). RSE y los miners mantienen los archivos que contienen.

**Nota:** `/var/rdz/logs/` requiere la máscara de bit de permiso 777 para permitir a cada cliente crear directorio `$LOGNAME` y almacenar los archivos de anotaciones específicos del cliente.

- SCLMDT y la Pasarela de cliente TSO/ISPF de ISPF utilizan `/var/rdz/WORKAREA/` para transferir datos entre z/OS UNIX y los espacios de direcciones basados en MVS. La ubicación real se especifica en `rsed.envvars` (variable `SCLMDT_CMDSERV_HOME`). ISPF y SCLMDT mantienen los archivos que contienen.

**Nota:** `/var/rdz/WORKAREA/` requiere la máscara de bit de permiso 777 para permitir a cada cliente crear archivos temporales.

- La Pasarela de cliente TSO/ISPF de ISPF y varios miners utilizan `/tmp/` para almacenar datos temporales. La ubicación no se puede personalizar. ISPF y los miners mantienen los archivos que contienen. También es la ubicación predeterminada para los archivos de vuelco Java, que pueden personalizarse con la variable `_CEE_DUMPTARG` de `rsed.envvars`.

**Nota:** `/tmp/` requiere la máscara de bit de permiso 777 para permitir a cada cliente crear archivos temporales.

## Privilegios de actualización para usuarios no administradores del sistema

Los datos de algunos directorios, como `/var/rdz/projects/`, los mantienen usuarios no administradores del sistema, como gestores de proyectos, y es posible que no dispongan de demasiados privilegios de actualización en z/OS UNIX. Si sólo hay un ID de usuario que mantiene los archivos, no hay ningún problema una vez el ID de usuario se haya convertido en propietario del directorio y de todo lo que contiene.

```
chown -R IBMUSER /var/rdz/projects/
```

Cuando hay varios ID de usuario que requieren permisos de actualización para el directorio, puede trabajar con los bits de permiso de grupo.

1. Cree un grupo en el software de seguridad que tenga un segmento OMVS válido y conecte todos los ID de usuario que requieran acceso de actualización. Este es preferiblemente su grupo predeterminado.  

```
ADDGROUP RDZPROJ OMVS(GID(1200))
CONNECT IBMUSER GROUP(RDZPROJ)
ALTUSER IBMUSER DFLTGRP(RDZPROJ)
```
2. Utilice el mandato de z/OS UNIX **chgrp** para asignar el grupo al directorio y todos sus archivos. Este mandato debe repetirse cada vez que se añada un archivo nuevo y el ID de grupo deseado no es el grupo predeterminado del ID de usuario que ha añadido el archivo.

```
chgrp -R IBMUSER /var/rdz/projects/
```

3. Utilice el mandato de z/OS UNIX **chmod** para proporcionar a todo el grupo permiso de actualización en el directorio y todos sus archivos.

```
chmod -R 775 /var/rdz/projects/
```

## Capítulo 12. Consideraciones de WLM

Al contrario que las aplicaciones z/OS tradicionales, Developer for System z no es una aplicación monolítica que se pueda identificar fácilmente para el Gestor de carga de trabajo (WLM). Developer for System z está formado por varios componentes que interactúan para proporcionar al cliente acceso a los servicios y datos del host. Tal como se describe en Capítulo 11, “Qué es Developer for System z”, en la página 191, algunos de estos servicios están activos en diferentes espacios de direcciones, lo que resulta en diferentes clasificaciones WLM.

En este capítulo se tratan estos temas:

- “Clasificación de carga de trabajo”
- “Establecimiento de objetivos” en la página 205

### Clasificación de carga de trabajo

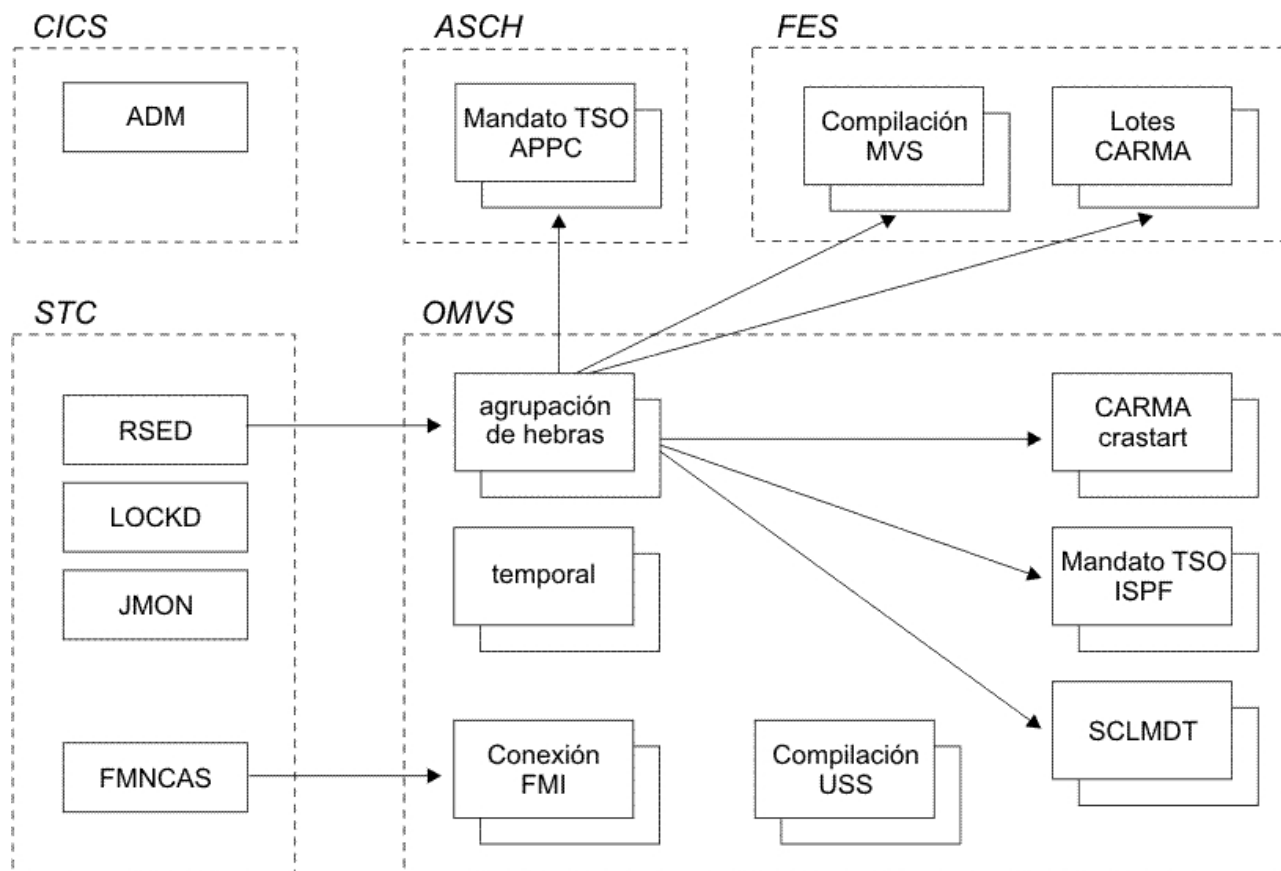


Figura 47. Clasificación de WLM

Figura 47 muestra una visión general básica de los subsistemas a través de los cuales se presentan las cargas de trabajo de Developer for System z a WLM.

El Gestor de despliegue de aplicaciones (ADM) está activo en una región CICS y por lo tanto seguirá las reglas de clasificación CICS en WLM.

Daemon RSE (RSED), daemon de bloqueo (LOCKD) y Supervisor de trabajos JES (JMON) son tareas iniciadas por Developer for System z (o trabajos por lotes de larga ejecución), cada uno con su espacio de direcciones individual.

Tal como se explica en “RSE como aplicación Java” en la página 193, el daemon RSE genera un proceso hijo para cada servidor de agrupaciones de hebras RSE (que soporta un número variable de clientes). Cada agrupación de hebras está activo en un espacio de direcciones aparte (utilizando un iniciador z/OS UNIX, BPXAS). Como se trata de procesos generados, se clasifican mediante las reglas de clasificación WLM OMVS, no las reglas de clasificación de tareas iniciadas.

Los clientes activos en una agrupación de hebras pueden crear muchos otros espacios de direcciones, según las acciones realizadas por los usuarios. Dependiendo de la configuración de Developer for System z, algunas cargas de trabajo, como por ejemplo el servicio de mandatos TSO (TSO cmd) o CARMA, se pueden ejecutar en subsistemas diferentes.

Los espacios de direcciones que aparecen en la lista de Figura 47 en la página 203 siguen permaneciendo en el sistema durante tiempo suficiente para ser visibles pero debe tener en cuenta que debido al diseño de z/OS UNIX, también hay varios espacios de direcciones temporales de vida breve. Estos espacios de direcciones temporales están activos en el subsistema OMVS.

Tenga en cuenta que mientras que las agrupaciones de hebras de RSE utilizan el mismo ID de usuario y un nombre de trabajo parecido al del daemon RSE, todos los espacios de direcciones iniciados por una agrupación de hebras son propiedad del IDE de usuario del cliente que solicita la acción. El ID de usuario cliente también se utiliza como (es parte del) nombre de trabajo para todos los espacios de direcciones basados en OMVS declarados por la agrupación de hebras.

Otros servicios han creado más espacios de direcciones que Developer for System z utiliza, como por ejemplo Gestor de archivos (FMNCAS) o z/OS UNIX REXEC (construcción USS).

## Reglas de clasificación

WLM utiliza reglas de clasificación para correlacionar el trabajo que entra en el sistema con una clase de servicio. Esta clasificación se basa en calificadores de trabajo. El primer calificador (obligatorio) es el tipo de subsistema que recibe la petición de trabajo. Tabla 29 enumera los tipos de subsistema que pueden recibir cargas de trabajo de Developer for System z.

*Tabla 29. Subsistemas de punto de entrada de WLM*

Tipo de subsistema	Descripción de trabajo
ASCH	Las peticiones de trabajo incluyen todos los programas de transacción APPC planificados por el planificador de transacciones APPC/MVS proporcionados por IBM, ASCH.
CICS	Las peticiones de trabajo incluyen todas las transacciones procesadas por CICS.
JES	Las solicitudes de trabajo incluyen todos los trabajos iniciados por JES2 o JES3.
OMVS	Las peticiones de trabajo incluyen el trabajo procesado en espacios de direcciones hijo bifurcados de z/OS UNIX System Services.

Tabla 29. Subsistemas de punto de entrada de WLM (continuación)

Tipo de subsistema	Descripción de trabajo
STC	Las peticiones de trabajo incluyen todo el trabajo iniciado por los mandatos START y MOUNT. STC también incluye espacios de direcciones de componentes del sistema.

Tabla 30La tabla 2 enumera calificadores adicionales que se pueden utilizar para asignar una carga de trabajo a una clase de servicio específica. Consulte la planificación de MVS: Workload Management (SA22-7602) para obtener más detalles sobre los calificadores de trabajo de la lista.

Tabla 30. Calificadores de trabajo de WLM

		ASCH	CICS	JES	OMVS	STC
AI	Información de contabilidad	x		x	x	x
LU	Nombre de LU (*)		x			
PF	Realizar (*)			x		x
PRI	Prioridad			x		
SE	Nombre de entorno de planificación			x		
SSC	Nombre de recogida de subsistema			x		
SI	Instancia de subsistema (*)		x	x		
SPM	Parámetro de subsistema					x
PX	Nombre de Sysplex	x	x	x	x	x
SY	Nombre de sistema (*)	x			x	x
TC	Clase de transacción/trabajo (*)	x		x		
TN	Nombre de transacción/trabajo (*)	x	x	x	x	x
UI	ID de usuario (*)	x	x	x	x	x

**Nota:** Para los calificadores marcados con (\*), puede especificar grupos de clasificación añadiendo una G a la abreviación de tipo. Por ejemplo, un grupo de nombres de transacción sería TNG.

## Establecimiento de objetivos

Tal como se explica en “Clasificación de carga de trabajo” en la página 203, Developer for System z crea varios tipos de cargas de trabajo en el sistema. Estas diferentes tareas se comunican entre sí, lo que implica que el tiempo transcurrido real se vuelve importante para evitar los problemas de tiempo de espera para las conexiones entre las tareas. Como resultado, las tareas de Developer for System z deben colocarse en clases de servicio de alto rendimiento o en clases de servicio de rendimiento moderado con una alta prioridad.

Por lo tanto, es recomendable revisar y posiblemente actualizar sus objetivos de WLM. Esto es especialmente cierto para las tiendas MVS tradicionales sin experiencia con cargas de trabajo OMVS para las que el tiempo es muy importante.

#### Nota:

- La información de objetivo de esta sección se mantiene deliberadamente a un nivel descriptivo porque los objetivos de rendimiento reales son muy específicos del sitio.
- Para ayudarle a comprender el impacto de una tarea específica en el sistema, se utilizan términos como utilización de recursos mínima, moderada y sustancial. Todos ellos son relativos a la utilización de recursos total de Developer for System z, no de todo el sistema.

Tabla 31 enumera los espacios de direcciones utilizados por Developer for System z. z/OS UNIX sustituirá "x" en la columna "Nombre de tarea" por un número aleatorio de 1 dígito.

*Tabla 31. Cargas de trabajo WLM*

Descripción	Nombre de tarea	Carga de trabajo
Supervisor de trabajos JES	JMON	STC
Daemon de bloqueo	LOCKD	STC
Daemon RSE	RSED	STC
Agrupación de hebras RSE	RSEDx	OMVS
Pasarela de cliente ISPF (Servicio de mandatos TSO y SCLMDT)	<userid>x	OMVS
Servicio de mandatos TSO (APPC)	FEKFRSRV	ASCH
CARMA (por lotes)	CRA<port>	JES
CARMA (crastart)	<userid>x	OMVS
CARMA (Pasarela de cliente ISPF)	<userid> y <userid>x	OMVS
Construcción de MVS (trabajo por lotes)	*	JES
Construcción de z/OS UNIX (mandatos de shell)	<userid>x	OMVS
Shell z/OS UNIX	<userid>	OMVS
Tarea de Gestor de archivos	<userid>x	OMVS
Gestor de despliegue de aplicaciones	CICSTS	CICS

## Consideraciones para la selección de objetivos

Las consideraciones de WLM generales siguientes le pueden ayudar a definir adecuadamente las definiciones de objetivos correctas para Developer for System z:

- Debe basar los objetivos en lo que se puede conseguir realmente, no en lo que desea que ocurra. Si establece los objetivos más arriba de lo necesario, WLM mueve recursos de trabajos menos importantes a trabajos más importantes que realmente no necesitan los recursos.
- Limite la cantidad de trabajo asignada a las clases de servicio SYSTEM y SYSSTC, porque estas clases tienen una prioridad de despacho más alta que cualquier clase gestionada por WLM. Utilice estas clases para el trabajo que tenga más importancia pero que utilice menos CPU.
- El trabajo que queda entre las reglas de clasificación termina en la clase SYSOTHER, que tiene un objetivo discrecional. Un objetivo discrecional indica a WLM que haga lo mejor que pueda cuando el sistema tenga recursos de sobra.

Cuándo utilizar los objetivos de tiempo de respuesta:

- Debe haber una cadencia de llegada de tareas constante (al menos 10 tareas en 20 minutos) para que WLM gestione adecuadamente un objetivo de tiempo de respuesta.
- Utilice los objetivos de tiempo de respuesta sólo para cargas de trabajo bien controladas porque una sola transacción larga tiene un impacto grande en el tiempo de respuesta promedio y puede hacer que WLM reaccione de manera exagerada.

Cuándo utilizar métodos de velocidad:

- Normalmente no puede conseguir un método de velocidad por encima del 90% por varias razones. Por ejemplo, todos los espacios de direcciones SYSTEM y SYSSTC tienen una prioridad de despacho mayor que cualquier objetivo de tipo de velocidad.
- WLM utiliza un número mínimo de ejemplos (utilizar y retardar) sobre los que basar sus decisiones de objetivo de velocidad. Así, cuanto menos trabajo se esté ejecutando en una clase de servicio, más se tardará en recoger el número necesario de ejemplos y en ajustar la política de despacho.
- Vuelva a evaluar los métodos de velocidad cuando cambie el hardware. Además, pasar a menos y más rápidos procesadores necesita cambios en los objetivos de velocidad.

## STC

Todas las tareas iniciadas por Developer for System z, daemon de RSE, daemon de bloqueo y Supervisor de trabajos JES están dando servicio en tiempo real a peticiones de cliente.

*Tabla 32. Cargas de trabajo WLM - STC*

Descripción	Nombre de tarea	Carga de trabajo
Supervisor de trabajos JES	JMON	STC
Daemon de bloqueo	LOCKD	STC
Daemon RSE	RSED	STC

- Supervisor de trabajos JES  
El Supervisor de trabajos JES proporciona todos los servicios relacionados con JES como por ejemplo el sometimiento de trabajos, el examen de archivos en spool y la ejecución de mandatos de operador JES. Debe especificar un objetivo de velocidad de un periodo y alto rendimiento porque la tarea no comunica las transacciones individuales a WLM. La utilización de recursos depende mucho de las acciones de usuario y por lo tanto fluctuará, pero se espera que sea entre mínima y moderada.
- Daemon de bloqueo  
El daemon de bloqueo consulta las tablas de cola GRS a petición del cliente y del operador y empareja el resultado con los usuarios de Developer for System z conocidos. Debe especificar un objetivo de velocidad de un periodo y alto rendimiento porque la tarea no comunica las transacciones individuales a WLM. Se espera que la utilización de recursos sea mínima.
- Daemon RSE  
El daemon RSE maneja el inicio de sesión y la autenticación de clientes y gestiona las diferentes agrupaciones de hebras de RSE. Debe especificar un objetivo de velocidad de un periodo y alto rendimiento porque la tarea no comunica las transacciones individuales a WLM. Se espera que la utilización de recursos sea moderada, con un pico al principio del día de trabajo.

## OMVS

Las cargas de trabajo de OMVS pueden dividirse en dos grupos, agrupaciones de hebras RSE y todo lo demás. Esto es porque todas las cargas de trabajo excepto las agrupaciones de hebras RSE utilizan el ID de usuario de cliente como base para el nombre del espacio de direcciones. (z/OS UNIX sustituirá "x" en la columna "Nombre de tarea" por un número aleatorio de 1 dígito.)

Tabla 33. Cargas de trabajo WLM - OMVS

Descripción	Nombre de tarea	Carga de trabajo
Agrupación de hebras RSE	RSEDx	OMVS
Pasarela de cliente ISPF (Servicio de mandatos TSO y SCLMDT)	<userid>x	OMVS
CARMA (crastart)	<userid>x	OMVS
CARMA (Pasarela de cliente ISPF)	<userid> y <userid>x	OMVS
Construcción de z/OS UNIX (mandatos de shell)	<userid>x	OMVS
Shell z/OS UNIX	<userid>	OMVS
Tarea de Gestor de archivos	<userid>x	OMVS

- Agrupación de hebras RSE

Una agrupación de hebras RSE es como el corazón y el cerebro Developer for System z. Casi todos los datos fluyen por aquí, y los mineros (hebras específicas de usuario) de dentro de la agrupación de hebras controlan las acciones para la mayoría de las otras tareas relacionadas de Developer for System z. Debe especificar un objetivo de velocidad de un periodo y alto rendimiento porque la tarea no comunica las transacciones individuales a WLM. La utilización de recursos depende mucho de las acciones de usuario y por lo tanto fluctuará, pero se espera que sea sustancial.

Las cargas de trabajo restantes finalizarán todas en la misma clase de servicio debido a un convenio de denominación de espacios de direcciones común. Debe especificar un objetivo de varios periodos para esta clase de servicio. Los primeros periodos deberían ser objetivos de tiempo de respuesta percentil de alto rendimiento, mientras que el último periodo debería tener un objetivo de velocidad de rendimiento moderado. Algunas cargas de trabajo como por ejemplo la Pasarela de cliente ISPF informarán de transacciones individuales a WLM, mientras que otras no lo harán.

- Pasarela de cliente ISPF

La Pasarela de cliente ISPF es un servicio ISPF invocado por Developer for System z para ejecutar mandatos TSO y ISPF no interactivos. Esto incluye mandatos explícitos emitidos por el cliente así como mandatos implícitos emitidos por Developer for System z, como por ejemplo obtener una lista de miembros de PDS. La utilización de recursos depende mucho de las acciones de usuario y por lo tanto fluctuará, pero se espera que sea mínima.

- CARMA

CARMA es un servidor Developer for System z opcional utilizado para interactuar con Gestores de configuraciones de software (SCM), como por ejemplo CA Endevor® SCM. Developer for System z permite diferentes métodos de inicio para un servidor CARMA, algunos de los cuales se convierten en una

carga de trabajo OMVS. La utilización de recursos depende mucho de las acciones de usuario y por lo tanto fluctuará, pero se espera que sea mínima.

- Construcción z/OS UNIX

Cuando un cliente inicia una construcción para un proyecto z/OS UNIX, z/OS UNIX REXEC (o SSH) iniciará una tarea que ejecuta un número de mandatos de shell z/OS UNIX para realizar la construcción. La utilización de recursos depende mucho de las acciones de usuario y por lo tanto fluctuará, pero se espera que sea entre moderada y sustancial, dependiendo del tamaño del proyecto.

- Shell z/OS UNIX

Esta carga de trabajo procesa mandatos shell z/OS UNIX emitidos por el cliente. La utilización de recursos depende mucho de las acciones de usuario y por lo tanto fluctuará, pero se espera que sea mínima.

- IBM File Manager

Aunque no se trata de espacios de direcciones de Developer for System z, los procesos hijo del Gestor de archivos generados se incluyen en esta lista porque se pueden iniciar a petición de un cliente de Developer for System z y estas tareas utilizan el mismo convenio de denominación que las tareas de Developer for System z. Estas tareas de Gestor de archivos procesan acciones de conjunto de datos MVS no triviales, como por ejemplo la edición formateada de un archivo VSAM. La utilización de recursos depende mucho de las acciones de usuario y por lo tanto fluctuará, pero se espera que sea entre mínima y moderada.

## JES

Los procesos por lotes gestionados por JES los utiliza Developer for System z de varias maneras. La utilización más común es para construcciones MVS, donde un trabajo se somete y supervisa para determinar cuándo finaliza. Pero Developer for System z también podría iniciar un servidor CARMA por lotes y comunicarse con el mediante TCP/IP.

Tabla 34. Cargas de trabajo WLM - JES

Descripción	Nombre de tarea	Carga de trabajo
CARMA (por lotes)	CRA<port>	JES
Construcción de MVS (trabajo por lotes)	*	JES

- CARMA

CARMA es un servidor Developer for System z opcional utilizado para interactuar con Gestores de configuraciones de software (SCM), como por ejemplo CA Endevor® SCM. Developer for System z permite diferentes métodos de inicio para un servidor CARMA, algunos de los cuales se convierten en una carga de trabajo JES. Debe especificar un objetivo de velocidad de un periodo y alto rendimiento porque la tarea no comunica las transacciones individuales a WLM. La utilización de recursos depende mucho de las acciones de usuario y por lo tanto fluctuará, pero se espera que sea mínima.

- Construcción de MVS

Cuando un cliente inicia una construcción para un proyecto MVS, Developer for System z iniciará un trabajo por lotes para realizar la construcción. La utilización de recursos depende mucho de las acciones de usuario y por lo tanto fluctuará, pero se espera que sea entre moderada y sustancial, dependiendo del tamaño del proyecto. En función de sus circunstancias locales, puede ser aconsejable seguir diferentes estrategias de objetivo de rendimiento moderado.

- Debe especificar un objetivo de varios periodos con un periodo de tiempo de respuesta percentil y un periodo de velocidad final. En este casos, sus desarrolladores deben utilizar principalmente el mismo procedimiento de construcción y archivos de entrada de tamaños parecidos para crear trabajos con tiempos de respuesta uniformes. Debe haber también una cadencia de llegada de trabajos constante (al menos 10 trabajos en 20 minutos) para que WLM gestione adecuadamente un objetivo de tiempo de respuesta.
- Un objetivo de velocidad se ajusta mejor a los trabajos por lotes porque este objetivo puede manejar tiempos de ejecución y cadencias de llegada muy variables.

## ASCH

En las versiones actuales Developer for System z, la Pasarela de cliente ISPF se utiliza para ejecutar mandatos TSO e ISPF no interactivos. Por razones históricas, Developer for System z también soporta la ejecución de estos mandatos a través de una transacción APPC.

Tabla 35. Cargas de trabajo WLM - ASCH

Descripción	Nombre de tarea	Carga de trabajo
Servicio de mandatos TSO (APPC)	FEKFRSRV	ASCH

- Servicio de mandatos TSO  
El servicio de mandatos TSO puede iniciarse como una transacción APPC por parte de Developer for System z para ejecutar mandatos TSO e ISPF no interactivos. Esto incluye mandatos explícitos emitidos por el cliente así como mandatos implícitos emitidos por Developer for System z, como por ejemplo obtener una lista de miembros de PDS. Debe especificar un objetivo de varios periodos para esta clase de servicio. Para los primeros periodos debe especificar objetivos de tiempo de respuesta percentil de alto rendimiento. Para el último periodo debe especificar un objetivo de velocidad de rendimiento moderado. La utilización de recursos depende mucho de las acciones de usuario y por lo tanto fluctuará, pero se espera que sea mínima.

## CICS

El Gestor de despliegue de aplicaciones es un servidor de Developer for System z opcional que está activo dentro de una región de CICS Transaction Server.

Tabla 36. Cargas de trabajo WLM - CICS

Descripción	Nombre de tarea	Carga de trabajo
Gestor de despliegue de aplicaciones	CICSTS	CICS

- Gestor de despliegue de aplicaciones  
El servidor Gestor de despliegue de aplicaciones que está activo dentro de una región CICSTS, permite descargar de forma segura las tareas de gestión CICSTS seleccionadas para los desarrolladores. La utilización de recursos depende mucho de las acciones de usuario y por lo tanto fluctuará, pero se espera que sea mínima. El tipo de clase de servicio que utilice depende del resto de transacciones activas en esta región CICS y por lo tanto no se trata en detalle.

WLM soporta varios tipos de gestión que puede utilizar para CICS:

- Gestionar CICS hacia un objetivo de región

| El objeto se establece en una clase de servicio que gestiona los espacios de  
| dirección CICS. Sólo puede utilizar un objetivo de velocidad de ejecución para  
| esta clase de servicio. WLM utiliza las reglas de clasificación de JES o STC para  
| los espacios de dirección pero no utiliza las reglas de clasificación de  
| subsistemas CICS para transacciones.

- Gestionar CICS hacia un objetivo de tiempo de respuesta

| Se puede establecer un objetivo de tiempo de respuesta en una clase de servicio  
| asignada a una sola transacción o a un grupo de transacciones. WLM utiliza las  
| reglas de clasificación JES o STC para los espacios de dirección y las reglas de  
| clasificación de subsistema CICS para transacciones.



---

## Capítulo 13. Consideraciones acerca de los ajustes

Tal como se explica en Capítulo 11, “Qué es Developer for System z”, en la página 191, RSE (Explorador de sistemas remotos) es el núcleo de Developer for System z. Para gestionar las conexiones y cargas de trabajo de los clientes, RSE está formado por un espacio de direcciones de daemon, que controla los espacios de direcciones de agrupaciones de hebras. El daemon actúa como punto focal a efectos de conexión y gestión, mientras que las agrupaciones de hebras procesan las cargas de trabajo del cliente.

Ello hace que RSE sea el destino principal para ajustar la configuración de Developer for System z. Sin embargo, mantener a cientos de usuarios, cada uno de los cuales utiliza 16 hebras o más, una cantidad determinada de almacenamiento y, posiblemente, 1 o más espacios de direcciones, requiere que la configuración de Developer for System z y z/OS sea la adecuada.

En este capítulo se tratan estos temas:

- “Uso de recursos”
- “Uso de almacenamiento” en la página 224
- “Uso de espacio del sistema de archivos de z/OS UNIX” en la página 230
- “Definiciones de recursos clave” en la página 233
- “Definiciones de varios recursos” en la página 236
- “Supervisión” en la página 238
- “Configuración de ejemplo” en la página 241

---

### Uso de recursos

Utilice la información de esta sección para estimar el uso normal y máximo de recursos por parte de Developer for System z, de manera que pueda planificar acorde la configuración del sistema.

Al utilizar los números y las fórmulas presentadas en esta sección para definir los valores para los límites del sistema, tenga en cuenta que está trabajando con estimaciones bastante precisas. Deje un margen suficiente al establecer los límites del sistema para permitir el uso de recursos por las tareas temporales o por otras tareas, o por usuarios que se conecten varias veces al host simultáneamente. (Por ejemplo, a través de RSE y TN3270).

#### Nota:

- El ámbito de la información está limitado a los servicios a los que se accede a través de RSE proporcionados por el propio Developer for System z. Por ejemplo, el uso de recursos de TN3270 no está documentado (no se accede a través de RSE), como tampoco lo está el uso de recursos de los programas llamados durante construcciones remotas (basadas en host) de proyectos de MVS o z/OS UNIX (no proporcionados por Developer for System z).
- Añadir extensiones externas a Developer for System z puede aumentar los contadores de uso de recursos.

- Todos los servicios tienen tareas "de mantenimiento" cortas, que utilizan recursos durante su ejecución, y que pueden ejecutarse secuencialmente o paralelamente. Los recursos utilizados por estas tareas no están documentados.
- El uso de recursos específicos del usuario del software requisito, como la Pasarela de cliente ISPF, se documenta cuando es útil.
- Los números que se presentan aquí pueden cambiar sin notificación previa.

## Visión general

Las siguientes tablas proporcionan una visión general del número de espacios de direcciones, procesos y hebras utilizados por Developer for System z. En las siguientes secciones puede encontrar más detalles acerca de los números que se presentan aquí:

- "Recuento de espacios de direcciones" en la página 215
- "Recuento de procesos" en la página 218
- "Recuento de hebras" en la página 221

Tabla 37 proporciona una visión general de los recursos clave utilizados por las tareas iniciadas de Developer for System z. Estos recursos se asignan únicamente una vez. Todos los clientes de Developer for System z los comparten.

*Tabla 37. Uso de recursos comunes*

Tarea iniciada	Espacios de direcciones	Procesos	Hebras
JMON	1	1	3
LOCKD	1	3	10
RSED	1	3	11
RSEDx	(a)	2	10

**Nota:** (a) Hay, como mínimo, un espacio de direcciones de agrupaciones de hebras RSE activo. Consulte "Recuento de espacios de direcciones" en la página 215 para determinar el número real de espacios de direcciones de agrupaciones de hebras RSE.

Tabla 38 proporciona una visión general de los recursos clave utilizados por el software de seguridad. Estos recursos se asignan para cada cliente de Developer for System z que invoque la función relacionada.

*Tabla 38. Uso de recursos requisito específicos del usuario*

Software requisito	Espacios de direcciones	Procesos	Hebras
Pasarela de cliente ISPF	1	2	4
APPC	1	1	2
Gestor de archivos	1	1	2

Tabla 39 en la página 215 proporciona una visión general de los recursos clave utilizados por cada cliente de Developer for System z al ejecutar la función especificada. Los valores no numéricos, como ISPF, son una referencia al valor correspondiente de Tabla 38.

Tabla 39. Uso de recursos específicos del usuario

Acción de usuario	Espacios de direcciones	Procesos	Hebras		
	ID de usuario	ID de usuario	ID de usuario	RSEDx	JMON
Inicio de sesión	-	-	-	16	1
Temporizador para tiempo de espera desocupado	-	-	-	1	-
Ampliar PDS(E)	ISPF	ISPF	ISPF	-	-
Abrir conjunto de datos	ISPF	ISPF	ISPF	-	-
Mandato TSO	ISPF	ISPF	ISPF	-	-
Shell de z/OS UNIX	1	1	1	6	-
Construcción de MVS	1	-	-	-	-
Construcción de z/OS UNIX	3	3	3	-	-
CARMA (por lotes)	1	1	2	1	-
CARMA (crastart)	1	1	2	4	-
CARMA (ispf)	4	4	7	5	-
SCLMDT	ISPF	ISPF	ISPF	-	-
Integración del gestor de archivos	ISPF + FM	ISPF + FM	ISPF + FM	-	-
Integración del analizador de errores	-	-	-	-	-

**Nota:** ISPF se puede sustituir por APPC, excepto por SCLM Developer Toolkit.

## Recuento de espacios de direcciones

Tabla 40 enumera los espacios de direcciones utilizados por Developer for System z, donde “u” de la columna “Recuento” indica que la cantidad debe multiplicarse por el número de usuarios activos simultáneamente que están utilizando la función. z/OS UNIX sustituirá “x” de la columna “Nombre de tarea” por un número de 1 dígito aleatorio.

Tabla 40. Recuento de espacios de direcciones

Recuento	Descripción	Nombre de tarea	Compartido	Finaliza tras
1	Supervisor de trabajos JES	JMON	Sí	Nunca

Tabla 40. Recuento de espacios de direcciones (continuación)

Recuento	Descripción	Nombre de tarea	Compartido	Finaliza tras
1	Daemon de bloqueo	LOCKD	Sí	Nunca
1	Daemon RSE	RSED	Sí	Nunca
(a)	Agrupación de hebras RSE	RSEDx	Sí	Nunca
1u	Pasarela de cliente ISPF (Servicio de mandatos TSO y SCLMDT)	<userid>x	No	15 minutos o fin de sesión de usuario
1u	Servicio de mandatos TSO (APPC)	FEKFRSRV	No	60 minutos o fin de sesión de usuario
1u	CARMA (por lotes)	CRA<port>	No	7 minutos o fin de sesión de usuario
1u	CARMA (crastart)	<userid>x	No	7 minutos o fin de sesión de usuario
4u	CARMA (ispf)	(1)<userid> o (3)<userid>x	No	7 minutos o fin de sesión de usuario
(b)	Uso simultáneo de la Pasarela de cliente ISPF por 1 usuario	<userid>x	No	Compleción de la tarea
1u	Construcción de MVS (trabajo por lotes)	*	No	Compleción de la tarea
3u	Construcción de z/OS UNIX (mandatos de shell)	<userid>x	No	Compleción de la tarea
1u	Shell de z/OS UNIX	<userid>	No	Fin de sesión de usuario
(c)	Gestor de archivos	<userid>x	No	Compleción de la tarea

**Nota:**

- (a) Hay, como mínimo, un espacio de direcciones de agrupaciones de hebras RSE activo. EL número real depende de:
  - La directiva `minimum.threadpool.process` de `rsed.envvars`. El valor predeterminado es 1.
  - El número de usuarios a los que una agrupación de hebras puede proporcionar servicios. El objetivo de los valores predeterminados es de 60 usuarios por agrupación de hebras.
  - La marca de límite superior de usuarios activos simultáneamente, puesto que las agrupaciones de hebras desocupadas no se detienen automáticamente.
- (b) Developer for System z tiene varias hebras activas por usuario. En caso de que el espacio de direcciones de la Pasarela de cliente ISPF no haya terminado de servir la solicitud de una hebra cuando otra hebra manda una solicitud, ISPF iniciará una Pasarela de cliente nueva para procesar la nueva solicitud. Este espacio de direcciones finaliza tras la compleción de tareas.
- (c) La escucha del Gestor de archivos inicia un espacio de direcciones por cada objeto que se debe manipular, por ejemplo un VSAM. Este espacio de direcciones permanece activo hasta que Developer for System z indica que el objeto ya no es necesario, por ejemplo cerrando el VSAM.
- SCLMDT requiere un espacio de direcciones de Pasarela de cliente ISPF. SCLMDT comparte el espacio de direcciones con el servicio de mandatos TSO.

- La mayor parte de acciones relacionadas con conjuntos de datos de MVS utilizan el servicio de mandatos TSO, que puede estar activo en la Pasarela de cliente ISPF o en una transacción APPC, respectivamente.

Utilice la fórmula de Figura 48 para estimar el número máximo de espacios de direcciones utilizados por Developer for System z.

$$3 + A + N*(x + y + z) + (2 + N*0.01)$$

Figura 48. Número máximo de espacios de direcciones

Donde

- “3” iguala el número de espacios de direcciones del servidor activo permanente.
- “A” representa el número de espacios de direcciones de agrupaciones de hebras RSE.
- “N” representa el número máximo de usuarios simultáneos.
- “x” es uno de los siguientes valores, dependiendo de las opciones de configuración seleccionadas.

X	SCLMDT	TSO a través de la Pasarela de cliente	TSO a través de APPC
1	No	No	Sí
1	No	Sí	No
1	Sí	Sí	No

- “y” es uno de los siguientes valores, dependiendo de las opciones de configuración seleccionadas.

Y	
0	No CARMA
1	CARMA (por lotes)
1	CARMA (crastart)
4	CARMA (ispf)

- “z” es 0 de forma predeterminada, pero puede aumentar dependiendo de las acciones de usuario:
  - Añada 1 cuando se realice una construcción de MVS. Estos espacios de direcciones finalizan cuando se completa la tarea de construcción relacionada (un trabajo por lotes).
  - Añada 3 cuando se realice una construcción de z/OS UNIX. Tenga en cuenta que el número real puede ser superior, dependiendo de las necesidades de los programas invocados. Estos espacios de direcciones finalizan cuando se completa la tarea de construcción relacionada.
  - Añada 1 por cada interacción simultánea con el Gestor de archivos de IBM. Estos espacios de direcciones finalizan cuando el objeto solicitado ya no es necesario.
- “2 + N\*0.01” añade un almacenamiento intermedio para los espacios de direcciones temporales. El tamaño del almacenamiento intermedio necesario puede ser distinto en su sitio.

Utilice la fórmula de Figura 49 en la página 218 para estimar el número máximo de espacios de direcciones utilizados por un cliente de Developer for System z (sin

contar los espacios de direcciones temporales no documentados).

$$x + y + z$$

Figura 49. Número de espacios de direcciones por cliente

Donde

- "x" depende de las opciones de configuración seleccionadas y se documenta para que la fórmula calcule el número máximo de espacios de direcciones (Figura 48 en la página 217).
- "y" depende de las opciones de configuración seleccionadas y se documenta para que la fórmula calcule el número máximo de espacios de direcciones (Figura 48 en la página 217).
- "z" es 0 de forma predeterminada, pero puede aumentar dependiendo de las acciones de usuario, ya que se documenta para que la fórmula calcule el número máximo de espacios de direcciones (Figura 48 en la página 217).

Las definiciones de Tabla 41 pueden limitar el número real de espacios de direcciones.

Tabla 41. Límites de espacios de direcciones

Ubicación	Límite	Recursos afectados
rsed.envvars	maximum.threadpool.process	Limita el número de agrupaciones de hebras RSE
IEASYMxx	MAXUSER	Limita el número de espacios de direcciones
ASCHPMxx	MAX	Limita el número de iniciadores APPC para el servicio de mandatos TSO (APPC)

## Recuento de procesos

Tabla 42 enumera el número de procesos por espacio de direcciones utilizados por Developer for System z. "u" de la columna "Espacios de direcciones" indica que la cantidad debe multiplicarse por el número de usuarios activos simultáneamente que están utilizando la función.

Tabla 42. Recuento de procesos

Procesos	Espacios de direcciones	Descripción	ID de usuario
1	1	Supervisor de trabajos JES	STCJMON
3	1	Daemon de bloqueo	STCLOCK
3	1	Daemon RSE	STCRSE
2	(a)	Agrupación de hebras RSE	STCRSE
2	(b)	Pasarela de cliente ISPF (Servicio de mandatos TSO y SCLMDT)	<userid>
1	1u	Servicio de mandatos TSO (APPC)	<userid>
1	1u	CARMA (por lotes)	<userid>
1	1u	CARMA (crastart)	<userid>
1	1u	CARMA (ispf)	<userid>

Tabla 42. Recuento de procesos (continuación)

Procesos	Espacios de direcciones	Descripción	ID de usuario
1	3u	Construcción de z/OS UNIX (mandatos de shell)	<userid>
1	1u	Shell de z/OS UNIX	<userid>
1	(c)	Gestor de archivos	<userid>
(5)	(u)	SCLM Developer Toolkit	<userid>

**Nota:**

- (a) Hay, como mínimo, un espacio de direcciones de agrupaciones de hebras RSE activo. Consulte “Recuento de espacios de direcciones” en la página 215 para determinar el número real de espacios de direcciones de agrupaciones de hebras RSE.
- El daemon RSE y todas las agrupaciones de hebras RSE utilizan el mismo ID de usuario.
- (b) En los casos normales, y cuando se utilizan las opciones de configuración predeterminadas, hay 1 Pasarela de cliente ISPF activa por usuario. El número real puede variar, tal como se describe en “Recuento de espacios de direcciones” en la página 215.
- (c) La escucha del Gestor de archivos utiliza un proceso por cada objeto que se debe manipular, por ejemplo un VSAM. Este proceso permanece activo hasta que Developer for System z indica que el objeto ya no es necesario, por ejemplo cerrando el VSAM.
- SCLMDT requiere un espacio de direcciones de Pasarela de cliente ISPF. SCLMDT comparte el espacio de direcciones con el servicio de mandatos TSO.
- (u) Los procesos SCLMDT se ejecutan en el espacio de direcciones de la Pasarela de cliente ISPF, por lo que no tienen un valor para el recuento de espacios de direcciones.
- Los procesos SCLMDT son temporales y finalizan cuando se completan las tareas, pero puede haber varios procesos activos simultáneamente para un único usuario. Tabla 42 en la página 218 enumera el número máximo de procesos SCLMDT simultáneos.
- La mayor parte de acciones relacionadas con conjuntos de datos de MVS utilizan el servicio de mandatos TSO, que puede estar activo en la Pasarela de cliente ISPF o en una transacción APPC, respectivamente.
- Una construcción de z/OS UNIX utiliza tres procesos en total, y cada uno de ellos se ejecuta en su propio espacio de direcciones.
- Todos los procesos enumerados permanecen activos hasta que el espacio de direcciones relacionado finaliza, a menos que se indique lo contrario.

Utilice la fórmula de Figura 50 para estimar el número máximo de procesos utilizados por Developer for System z.

$$7 + 2 * A + N * (x + y + z) + (10 + N * 0.05)$$

Figura 50. Número máximo de procesos

Donde

- "7" iguala el número de procesos utilizados por los espacios de direcciones del servidor activo permanente.
- "A" representa el número de espacios de direcciones de agrupaciones de hebras RSE.
- "N" representa el número máximo de usuarios simultáneos.
- "x" es uno de los siguientes valores, dependiendo de las opciones de configuración seleccionadas.

X	SCLMDT	TSO a través de la Pasarela de cliente	TSO a través de APPC
1	No	No	Sí
2	No	Sí	No
7	Sí	Sí	No

- "y" es uno de los siguientes valores, dependiendo de las opciones de configuración seleccionadas.

Y	
0	No CARMA
1	CARMA (por lotes)
1	CARMA (crastart)
4	CARMA (ispf)

- "z" es 0 de forma predeterminada, pero puede aumentar dependiendo de las acciones de usuario:
  - Añada 1 cuando se abra un shell de z/OS UNIX. Este proceso permanece activo hasta que el usuario finaliza la sesión.
  - Añada 3 cuando se realice una construcción de z/OS UNIX. Tenga en cuenta que el número real puede ser superior, dependiendo de las necesidades de los programas invocados. Estos procesos finalizan cuando se completa la tarea de construcción relacionada.
  - Añada 1 por cada interacción simultánea con el Gestor de archivos de IBM. Estos procesos finalizan cuando el objeto solicitado ya no es necesario.
- "10 + N\*0.05" añade un almacenamiento intermedio para los procesos temporales. El tamaño del almacenamiento intermedio necesario puede ser distinto en su sitio.

Utilice la fórmula de Figura 51 para estimar el número máximo de procesos utilizados por un cliente de Developer for System z (sin contar los espacios de direcciones temporales no documentados).

$$(x + y + z) + 5*s$$

Figura 51. Número de procesos por cliente

Donde

- "x" depende de las opciones de configuración seleccionadas y se documenta para que la fórmula calcule el número máximo de procesos (Figura 50 en la página 219).

- "y" depende de las opciones de configuración seleccionadas y se documenta para que la fórmula calcule el número máximo de procesos (Figura 50 en la página 219).
- "z" es 0 de forma predeterminada, pero puede aumentar dependiendo de las acciones de usuario, ya que se documenta para que la fórmula calcule el número máximo de procesos (Figura 50 en la página 219).
- "s" es 1 cuando se utiliza SCLM Developer Toolkit; de lo contrario, el valor es 0.

Las definiciones de Tabla 43 pueden limitar el número real de procesos.

*Tabla 43. Límites de procesos*

Ubicación	Límite	Recursos afectados
BPXPRMxx	MAXPROCSYS	Limita el número total de procesos
BPXPRMxx	MAXPROCUSER	Limita el número de procesos por UID de z/OS UNIX

Nota:

- El daemon RSE y las agrupaciones de hebras RSE utilizan el mismo ID de usuario. Dado que el daemon RSE inicia una agrupación de hebras nueva cada vez que es necesaria, el número de procesos para este ID de usuario puede aumentar. Así, debe establecerse MAXPROCUSER para acomodar este aumento, que se puede formular como  $3 + 2 \cdot A$ .
- El límite MAXPROCUSER es por ID de usuario de z/OS UNIX exclusivo (UID). Multiplique el recuento estimado de procesos por usuario por el número de clientes activos simultáneamente, en caso de que los usuarios compartan el mismo UID.

## Recuento de hebras

Tabla 44 enumera el número de hebras utilizado por las funciones de Developer for System z seleccionadas. "u" de la columna "Hebras" indica que la cantidad debe multiplicarse por el número de usuarios activos simultáneamente que están utilizando la función. El recuento de hebras se enumera por proceso, puesto que los límites están establecidos a este nivel.

- RSEDx: Estas hebras se crean en la agrupación de hebras RSE, que pueden compartir varios clientes. Todas las hebras que finalizan en la misma agrupación de hebras pueden añadirse conjuntamente para obtener el recuento total.
- Activas: Estas hebras son parte del proceso que realiza en efecto la función solicitada. Cada proceso es una unidad autónoma, de manera que no es necesario sumar los recuentos de hebras, aunque estén asignados a un mismo ID de usuario; a menos que se especifique lo contrario.
- Programa de arranque: Los procesos del programa de arranque son necesarios para iniciar el proceso real. Cada uno tiene 1 hebra, y puede haber varios programas de arranque consecutivos. No es necesario sumar los recuentos de hebras.

*Tabla 44. Recuento de hebras*

Hebras			ID de usuario	Descripción
RSEDx	Activas	Programa de arranque		
-	$3 + 1u$	-	STCJMON	Supervisor de trabajos JES

Tabla 44. Recuento de hebras (continuación)

Hebras			ID de usuario	Descripción
-	10	2	STCLOCK	Daemon de bloqueo
-	11	2	STCRSE	Daemon RSE
10 (a) + 16u	-	1 (a)	STCRSE	Agrupación de hebras RSE
-	4u (b)	1u (b)	<userid>	Pasarela de cliente ISPF (Servicio de mandatos TSO y SCLMDT)
-	2u	-	<userid>	Servicio de mandatos TSO (APPC)
1u	2u	-	STCRSE y <userid>	CARMA (por lotes)
4u	2u	-	STCRSE y <userid>	CARMA (crastart)
5u	4u	3u	STCRSE y <userid>	CARMA (ispf)
-	1u (d)	2u	<userid>	Construcción de z/OS UNIX (mandatos de shell)
6u	1u	-	STCRSE y <userid>	Shell de z/OS UNIX
-	2u (c)	-	<userid>	Gestor de archivos
-	(5)	-	<userid>	SCLM Developer Toolkit
1u	-	-	STCRSE	Temporizador para tiempo de espera desocupado

**Nota:**

- (a) Hay, como mínimo, un espacio de direcciones de agrupaciones de hebras RSE activo. Consulte “Recuento de espacios de direcciones” en la página 215 para determinar el número real de espacios de direcciones de agrupaciones de hebras RSE.
- (b) En los casos normales, y cuando se utilizan las opciones de configuración predeterminadas, hay 1 Pasarela de cliente ISPF activa por usuario. El número real puede variar, tal como se describe en “Recuento de espacios de direcciones” en la página 215.
- (c) Hay un proceso específico del usuario (con el recuento de hebras enumerado) por interacción con el Gestor de archivos de IBM. Estos procesos finalizan cuando el objeto solicitado ya no es necesario.
- SCLMDT requiere un espacio de direcciones de Pasarela de cliente ISPF. SCLMDT comparte el espacio de direcciones con el servicio de mandatos TSO.

- Dependiendo de la acción seleccionada, SCLMDT puede utilizar varios procesos de una única hebra que finalizan cuando se completan las tareas. Tabla 44 en la página 221 enumera el número máximo de hebras SCLMDT simultáneos.
- La mayor parte de acciones relacionadas con conjuntos de datos de MVS utilizan el servicio de mandatos TSO, que puede estar activo en la Pasarela de cliente ISPF o en una transacción APPC, respectivamente.
- (d) Una construcción de z/OS UNIX invoca los distintos programas de utilidad de construcción, que pueden ser de varias hebras. Tabla 44 en la página 221 enumera el número mínimo de hebras de construcción de z/OS UNIX simultáneas.
- Todas las hebras enumeradas permanecen activas hasta que el proceso relacionado finaliza, a menos que se indique lo contrario.

Utilice la fórmula de Figura 52 para estimar el número máximo de hebras utilizado por una agrupación de hebras RSE. Utilice la fórmula de Figura 53 para estimar el número máximo de hebras utilizado por el Supervisor de trabajos JES.

$$9 + N*(16 + x + y + z) + (20 + N*0.1)$$

Figura 52. Número máximo de hebras de la agrupación de hebras RSE

$$3 + N$$

Figura 53. Número máximo de hebras del Supervisor de trabajos JES

Donde

- "N" representa el número máximo de usuarios simultáneos en esta agrupación de hebras o Supervisor de trabajos JES. El objetivo de los valores predeterminados es de 60 usuarios por agrupación de hebras.
- "x" es uno de los siguientes valores, dependiendo de las opciones de configuración seleccionadas.

X	SCLMDT	TSO a través de la Pasarela de cliente	TSO a través de APPC	Tiempo de espera
0	No	No	Sí	No
0	No	Sí	No	No
0	Sí	Sí	No	No
1	No	No	Sí	Sí
1	No	Sí	No	Sí
1	Sí	Sí	No	Sí

- "y" es uno de los siguientes valores, dependiendo de las opciones de configuración seleccionadas.

Y	
0	No CARMA
1	CARMA (por lotes)

Y	
4	CARMA (crastart)
5	CARMA (ispf)

- "z" es 0 de forma predeterminada, pero puede aumentar dependiendo de las acciones de usuario:
  - Añada 6 cuando se abra un shell de z/OS UNIX. Estas hebras permanecen activas hasta que el usuario finaliza la sesión.
- "20 + N\*0.1" añade un almacenamiento intermedio para las hebras temporales. El tamaño del almacenamiento intermedio necesario puede ser distinto en su sitio.

Las definiciones de Tabla 45 pueden limitar el número real de hebras en un proceso, que es de suma importancia para las agrupaciones de hebras RSE.

Tabla 45. Límites de hebras

Ubicación	Límite	Recursos afectados
BPXPRMxx	MAXTHREADS	Limita el número de hebras en un proceso
BPXPRMxx	MAXTHREADTASKS	Limita el número de tareas de MVS en un proceso.
BPXPRMxx	MAXASSIZE	Limita el tamaño de espacio de direcciones y, con ello, el almacenamiento disponible para los bloques de control relacionados con las hebras.
rsed.envvars	Xmx	Establece el tamaño del almacenamiento dinámico Java máximo. Este almacenamiento está reservado, por lo que no está disponible para los bloques de control relacionados con las hebras.
rsed.envvars	maximum.clients	Limita el número de clientes (y sus hebras) de una agrupación de hebras RSE.
rsed.envvars	maximum.threads	Limita el número de hebras de cliente en una agrupación de hebras RSE.
FEJJCNFG	MAX_THREADS	Limita el número de hebras en el Supervisor de trabajos JES.

**Nota:** El valor para maximum.threads de rsed.envvars debe ser inferior que el valor para MAXTHREADS y MAXTHREADTASKS de BPXPRMxx.

## Uso de almacenamiento

RSE es una aplicación Java, que implica que la planificación de uso de almacenamiento (memoria) para Developer for System z debe tener en cuenta dos límites de asignación de almacenamiento, el almacenamiento dinámico Java y el tamaño de Espacio de direcciones.

### Límite de tamaño de almacenamiento dinámico Java

Java ofrece varios servicios para facilitar los esfuerzos de codificación para las aplicaciones Java. Uno de estos servicios es la gestión de almacenamiento.

La gestión de almacenamiento de Java asigna bloques grandes de almacenamiento, y los utiliza para las solicitudes de almacenamiento de la aplicación. Este almacenamiento gestionado por Java se denomina almacenamiento dinámico Java.

La recogida de basura periódica (desfragmentación) reclama el espacio no utilizado del almacenamiento dinámico y reduce su tamaño.

El tamaño máximo de almacenamiento dinámico Java se define en `rsed.envvars` con la directiva `Xmx`. Si no se especifica esta directiva, Java utiliza un tamaño predeterminado de 64 MB.

Cada agrupación de hebras RSE (que proporciona servicio a las acciones de cliente) es una aplicación Java individual, por lo que tiene un almacenamiento dinámico Java personal. Tenga en cuenta que todas las agrupaciones de hebras utilizan el mismo archivo de configuración `rsed.envvars`, por lo que tienen el mismo límite de tamaño de almacenamiento dinámico Java.

El uso de la agrupación de hebras del almacenamiento intermedio Java depende sobre todo de las acciones realizadas por los clientes conectados. Es necesario supervisar regularmente el uso del almacenamiento dinámico para establecer el límite de tamaño de almacenamiento dinámico óptimo. Utilice el mandato de operador **modify display process** para supervisar el uso del almacenamiento dinámico Java por parte de las agrupaciones de hebras RSE.

## Límite de tamaño del espacio de direcciones

Todas las aplicaciones z/OS, incluidas las aplicaciones Java, están activas dentro de un espacio de direcciones, por lo que están limitadas por las limitaciones de tamaño del espacio de direcciones.

El tamaño de espacio de direcciones deseado se especifica durante el inicio, por ejemplo con el parámetro `REGION` en `JCL`. Sin embargo, los valores del sistema pueden limitar el tamaño de espacio de direcciones real. Consulte "Tamaño del espacio de direcciones" en la página 153 para obtener más información sobre estos límites.

- `MAXASSIZE` en `SYS1.PARMLIB(BPXPRMxx)`
- `ASSIZEMAX` en el segmento `OMVS` del `ID` de usuario asignado a la tarea iniciada
- salidas del sistema `IEFUSI` y `IEALIMIT`

Las agrupaciones de hebras RSE heredan los límites de tamaño del espacio de direcciones del daemon RSE. El tamaño del espacio de direcciones debe ser suficiente para albergar el almacenamiento dinámico Java, el propio Java, las áreas de almacenamiento comunes y todos los bloques de control que el sistema crea para soportar la actividad de las agrupaciones de hebras, por ejemplo, un `TCB` (bloque de control de tareas) por hebra. Tenga en cuenta que parte del uso de este almacenamiento está por debajo de la línea de 16 MB.

Debe supervisar el tamaño del espacio de direcciones real antes de cambiar ningún valor que tenga una influencia sobre el mismo, como cambiar el tamaño del almacenamiento dinámico Java o la cantidad de usuarios soportados por una única agrupación de hebras. Utilice el software de supervisión del sistema habitual para rastrear el uso del almacenamiento real por `Developer for system z`. Si no dispone de una herramienta de supervisión por ello, se puede reunir la información básica con herramientas como la vista `SDSF DA` o `TASID` (una herramienta de información del sistema tal cual disponible en el sitio Web de ISPF "Soporte y descargas").

## Directrices de estimación de tamaño

Tal como se ha indicado anteriormente, el uso real que Developer for system z hace del almacenamiento está muy condicionado por la actividad del usuario. Algunas acciones utilizan una cantidad fija de almacenamiento (por ejemplo, el inicio de sesión), mientras que otras son variables (por ejemplo, enumerar conjuntos de datos con un calificador de alto nivel especificado).

- Utilice un espacio de direcciones de 2 GB para que RSE tenga espacio para el almacenamiento dinámico Java y todos los bloques de control del sistema.
- El objetivo del `rsed.envvars` de ejemplo es de 60 usuarios por agrupación de hebras.
  - `maximum.clients=60`
  - `maximum.threads=1000` ( $10+16*60 = 970$ , de manera que 1000 permite a 61 clientes)
- La configuración de `rsed.envvars` de ejemplo permite al almacenamiento dinámico Java crecer hasta 256 MB. Ello permite, con 60 clientes, que cada cliente pueda utilizar un media de 4 MB ( $60*4 = 240$ ).

Tenga en cuenta que RSE muestra el almacenamiento dinámico Java actual y el límite de tamaño del espacio de direcciones durante el inicio en el mensaje de consola FEK004I.

Utilice cualquiera de los siguientes casos de ejemplo si la supervisión muestra que el almacenamiento dinámico Java actual es insuficiente para la carga de trabajo real:

- Aumente el tamaño máximo de almacenamiento dinámico Java con la directiva `Xmx` de `rsed.envvars`. Antes de hacerlo, asegúrese de que hay espacio en el espacio de direcciones para el aumento de tamaño.
- Reduzca el número máximo de clientes por agrupación de hebras con la directiva `maximum.clients` de `rsed.envvars`. RSE seguirá soportando el mismo número de clientes, pero los clientes se distribuirán entre más agrupaciones de hebras.

## Análisis del uso de almacenamiento de ejemplo

Las pantallas de las siguientes figuras muestran algunos números del uso de almacenamiento de ejemplo para una configuración de Developer for system z determinada con una modificación. El tamaño máximo de almacenamiento dinámico Java se establece en 10 MB, dado que un máximo pequeño resultará en un percentil de uso mayor y los límites de tamaño del almacenamiento dinámico se alcanzarán antes.

Tamaño máx. almacenamiento dinámico=10MB y Tamaño AS privado=1,959MB

inicio

BPXM023I (STCRSE)  
ProcessId(268 ) Uso de memoria(7%) Clientes(0)

Jobname	Tiempo Cpu	Almacenamiento	EXCP
JMON	0.01	2740	72
LOCKD	1.60	28.7M	14183
RSED	4.47	32.8M	15910
RSED8	1.15	27.4M	12612

inicio de sesión 1

BPXM023I (STCRSE)  
ProcessId(268 ) Uso de memoria(13%) Clientes(1)

Jobname	Tiempo Cpu	Almacenamiento	EXCP
JMON	0.01	2864	81
LOCKD	1.64	28.8M	14259
RSED	4.55	32.8M	15980
RSED8	3.72	55.9M	24128

inicio de sesión 2

BPXM023I (STCRSE)  
ProcessId(268 ) Uso de memoria(23%) Clientes(2)

Jobname	Tiempo Cpu	Almacenamiento	EXCP
JMON	0.02	2944	86
LOCKD	1.66	28.9M	14268
RSED	4.58	32.9M	16027
RSED8	4.20	57.8M	25205

inicio de sesión 3

BPXM023I (STCRSE)  
ProcessId(268 ) Uso de memoria(37%) Clientes(3)

Jobname	Tiempo Cpu	Almacenamiento	EXCP
JMON	0.02	3020	91
LOCKD	1.67	29.0M	14277
RSED	4.60	32.9M	16076
RSED8	4.51	59.6M	26327

inicio de sesión 4

BPXM023I (STCRSE)  
ProcessId(268 ) Uso de memoria(41%) Clientes(4)

Jobname	Tiempo Cpu	Almacenamiento	EXCP
JMON	0.02	3108	96
LOCKD	1.68	29.0M	14286
RSED	4.61	32.9M	16125
RSED8	4.77	62.3M	27404

Figura 54. Uso de recursos con 5 inicios de sesión

inició de sesión 5

BPXM023I (STCRSE)  
ProcessId(268 ) Uso de memoria(41%) Clientes(4)  
ProcessId(33554706) Uso de memoria(13%) Clientes(1)

Jobname	Tiempo Cpu	Almacenamiento	EXCP
JMON	0.03	3184	101
LOCKD	1.69	29.1M	14295
RSED	4.64	32.9M	16229
RSED8	4.78	62.4M	27413
RSED9	4.60	56.6M	24065

Figura 55. Uso de recursos con 5 inicios de sesión (continuación)

Figura 54 en la página 227 y Figura 55 muestran un caso de ejemplo en el que 5 clientes inician sesión en un daemon RSE con un almacenamiento dinámico Java de 10 MB.

- Una agrupación de hebras (RSED8) está en estado latente en el inicio y utiliza alrededor de 27 MB, de los cuales 0.7 MB están en el almacenamiento intermedio Java (7% de 10 MB).
- La agrupación de hebras pasa a estar activa cuando se conecta el primer cliente, y utiliza otros 27 MB más 2 MB por cada cliente que se conecta.
- Parte de estos 2MB por conexión estará en el almacenamiento intermedio Java, tal como muestra el aumento del uso de almacenamiento dinámico.
- Sin embargo, no hay ningún patrón real sobre el uso de almacenamiento dinámico, puesto que depende de mecanismos Java que estiman el almacenamiento necesario y asignan más del necesario. La recogida de basura intermitente libera almacenamiento, haciendo que las tendencias sean todavía más difíciles de detectar.
- Mecanismos internos que limitan el número de conexiones por agrupación de hebras para asegurar que haya el tamaño de almacenamiento dinámico suficiente para el resultado de hebras activas en la quinta conexión en una agrupación de hebras nueva (RSED9). Estas redes de seguridad interna no se suelen invocar cuando se utiliza una configuración adecuada, ya que antes se alcanzarían otros límites (probablemente el `maximum.clients` de `rsed.envvars`).

Tamaño máx. almacenamiento dinámico=10MB y Tamaño AS privado=1,959MB

inicio

BPXM023I (STCRSE)  
ProcessId(212 ) Uso de memoria(7%) Clientes(0)

Jobname	Tiempo Cpu	Almacenamiento	EXCP
JMON	0.01	2736	71
LOCKD	1.73	30.5M	14179
RSED	4.35	32.9M	15117
RSED8	1.43	27.4M	12609

inicio de sesión

BPXM023I (STCRSE)  
ProcessId(212 ) Uso de memoria(13%) Clientes(1)

Jobname	Tiempo Cpu	Almacenamiento	EXCP
JMON	0.01	2864	80
LOCKD	1.76	30.6M	14255
RSED	4.48	33.0M	15187
RSED8	3.53	53.9M	24125

ampliar árbol de MVS grande (195 conjuntos de datos)

BPXM023I (STCRSE)  
ProcessId(212 ) Uso de memoria(13%) Clientes(1)

Jobname	Tiempo Cpu	Almacenamiento	EXCP
JMON	0.01	2864	80
LOCKD	1.78	30.6M	14255
RSED	4.58	33.1M	16094
RSED8	4.28	56.1M	24740

ampliar PDS pequeño (21 miembros)

BPXM023I (STCRSE)  
ProcessId(212 ) Uso de memoria(13%) Clientes(1)

Jobname	Tiempo Cpu	Almacenamiento	EXCP
IBMUSER2	0.22	2644	870
JMON	0.01	2864	80
LOCKD	1.78	30.6M	14255
RSED	4.61	33.1M	16108
RSED8	4.40	56.2M	24937

abrir miembro de tamaño medio (86 líneas)

BPXM023I (STCRSE)  
ProcessId(212 ) Uso de memoria(13%) Clientes(1)

Jobname	Tiempo Cpu	Almacenamiento	EXCP
IBMUSER2	0.22	2644	870
JMON	0.01	2864	80
RSED	4.61	33.1M	16108
RSED8	8.12	62.7M	27044

Figura 56. Uso de recursos al editar un miembro PDS

Figura 56 muestra un caso de ejemplo en el que 1 cliente inicia sesión en un daemon RSE con un almacenamiento dinámico Java de 10 MB y edita un miembro PDS.

- La búsqueda de catálogos que resulta en 195 nombres de conjuntos de datos ha utilizado aproximadamente 2MB de almacenamiento, todo ello por la actividad del sistema, ya que el uso de almacenamiento dinámico Java no aumenta.
- Abrir un PDS de 21 miembros apenas utiliza memoria en la agrupación de hebras, pero la pantalla muestra que se ha invocado el Servicio de mandatos TSO. Hay un espacio de direcciones nuevo activo (IBMUUSER2), que utiliza el tamaño de región asignado a este ID de usuario en TSO. Este espacio de direcciones permanece activo durante un período de tiempo especificado, de manera que el servicio de mandatos TSO lo puede volver a utilizar para futuras peticiones.
- Abrir un miembro muestra números parecidos a ampliar un calificador de alto nivel. El uso de almacenamiento dinámico Java permanece igual, pero hay un aumento del almacenamiento de 6.5 MB como consecuencia de la actividad del sistema.

---

## Uso de espacio del sistema de archivos de z/OS UNIX

La mayoría de datos relacionados con Developer for System z que no se escriben una una sentencia DD terminan en un archivo z/OS UNIX. El programador del sistema controla qué datos se escriben y a dónde van. Sin embargo, no controla la cantidad de datos que se escriben.

Los datos pueden agruparse en estas categorías:

- Análisis de problemas (archivos de anotaciones y archivos de vuelco del sistema). Se documentan más detalles en Capítulo 9, “Resolución de problemas de configuración”, en la página 137
- Auditoría, tal como se documenta en “Anotaciones de auditoría” en la página 165
- Datos temporales

Tal como se documenta en Capítulo 9, “Resolución de problemas de configuración”, en la página 137, Developer for System z escribe las anotaciones del host relacionadas con RSE en los siguientes directorios de z/OS UNIX:

- /var/rdz/logs para las anotaciones de tareas iniciadas RSE
- /var/rdz/logs/\$LOGNAME para anotaciones de usuario

De forma predeterminada, en las anotaciones sólo se escriben los mensajes de error y de aviso. De manera que, si todo sale según se prevee, estos directorios deberían contener únicamente archivos vacíos o prácticamente vacíos (sin contar las anotaciones de auditoría).

Puede habilitar las anotaciones de mensajes informativos, preferiblemente bajo la dirección del centro de soporte de IBM, cosa que aumenta significativamente el tamaño de los archivos de anotaciones.

inicio

```
$ ls -l /var/rdz/logs
total 144
-rw-rw-rw- 1 STCRSE STCGRP 33642 10 Jul 12:10 rsedaemon.log
-rw-rw-rw- 1 STCRSE STCGRP 1442 10 Jul 12:10 rseserver.log
```

inicio de sesión

```
$ ls -l /var/rdz/logs
total 144
drwxrwxrwx 3 IBMUSER SYS1 8192 10 Jul 12:11 IBMUSER
-rw-rw-rw- 1 STCRSE STCGRP 36655 10 Jul 12:11 rsedaemon.log
-rw-rw-rw- 1 STCRSE STCGRP 1893 10 Jul 12:11 rseserver.log
$ ls -l /var/rdz/logs/IBMUSER
total 160
-rw-rw-rw- 1 IBMUSER SYS1 3459 10 Jul 12:11 ffs.log
-rw-rw-rw- 1 IBMUSER SYS1 0 10 Jul 12:11 ffsget.log
-rw-rw-rw- 1 IBMUSER SYS1 0 10 Jul 12:11 ffsput.log
-rw-rw-rw- 1 IBMUSER SYS1 303 10 Jul 12:11 lock.log
-rw-rw-rw- 1 IBMUSER SYS1 126 10 Jul 12:11 rmt_classloader_cache.jar
-rw-rw-rw- 1 IBMUSER SYS1 7266 10 Jul 12:11 rsecomm.log
-rw-rw-rw- 1 IBMUSER SYS1 0 10 Jul 12:11 stderr.log
-rw-rw-rw- 1 IBMUSER SYS1 0 10 Jul 12:11 stdout.log
```

fin de sesión

```
$ ls -l /var/rdz/logs
total 80
drwxrwxrwx 3 IBMUSER SYS1 8192 10 Jul 12:11 IBMUSER
-rw-rw-rw- 1 STCRSE STCGRP 36655 10 Jul 12:11 rsedaemon.log
-rw-rw-rw- 1 STCRSE STCGRP 2208 10 Jul 12:11 rseserver.log
$ ls -l /var/rdz/logs/IBMUSER
total 296
-rw-rw-rw- 1 IBMUSER SYS1 6393 10 Jul 12:11 ffs.log
-rw-rw-rw- 1 IBMUSER SYS1 0 10 Jul 12:11 ffsget.log
-rw-rw-rw- 1 IBMUSER SYS1 0 10 Jul 12:11 ffsput.log
-rw-rw-rw- 1 IBMUSER SYS1 609 10 Jul 12:11 lock.log
-rw-rw-rw- 1 IBMUSER SYS1 126 10 Jul 12:11 rmt_classloader_cache.jar
-rw-rw-rw- 1 IBMUSER SYS1 45157 10 Jul 12:11 rsecomm.log
-rw-rw-rw- 1 IBMUSER SYS1 0 10 Jul 12:11 stderr.log
-rw-rw-rw- 1 IBMUSER SYS1 176 10 Jul 12:11 stdout.log
```

detener

```
$ ls -l /var/rdz/logs
total 80
drwxrwxrwx 3 IBMUSER SYS1 8192 10 Jul 12:11 IBMUSER
-rw-rw-rw- 1 STCRSE STCGRP 36655 10 Jul 12:11 rsedaemon.log
-rw-rw-rw- 1 STCRSE STCGRP 2490 10 Jul 12:12 rseserver.log
```

*Figura 57. uso de espacio del sistema de archivos de z/OS UNIX*

Figura 57 muestra el uso de espacio mínimo del sistema de archivos de z/OS UNIX al utilizar el nivel de depuración 2 (mensajes informativos).

- Las anotaciones de tareas iniciadas utilizan 34 KB tras el inicio y aumentan paulatinamente a medida que los usuarios inician sesión, finalizan sesión, o bien se emiten mandatos de operador.
- Un directorio de anotaciones de cliente utiliza 11 KB tras el inicio de sesión y aumenta bastante cuando el usuario empieza a trabajar (no se muestra en el ejemplo).
- Finalizar la sesión añade otros 40 KB a las anotaciones del usuario, incrementando el número hasta 51 KB.

A excepción de las anotaciones de auditoría, los archivos de anotaciones se sobrescriben cada vez que se reinicia (para la tarea iniciada RSE) o se finaliza la sesión (para un cliente), manteniendo el tamaño total adecuado. La directiva `keep.last.log` de `rsed.envvars` cambia esto ligeramente, ya que puede hacer que RSE mantenga una copia de las anotaciones anteriores. Las copias antiguas se eliminan siempre.

Se envía un mensaje de aviso a la consola cuando el sistema de archivos que contiene los archivos de anotaciones de auditoría se está quedando sin espacio libre y la auditoría está activa. Este mensaje de consola (FEK103E) se repite regularmente hasta que se ha resuelto el problema de falta de espacio. Consulte “Mensajes de consola” en la página 132 para obtener una lista de los mensajes de consola generados por RSE.

Las definiciones de Tabla 46 controlan qué datos se graban en los directorios de anotaciones y dónde ubican los directorios.

*Tabla 46. Directivas de salidas de anotaciones*

Ubicación	Directiva	Función
<code>resecomm.properties</code>	<code>debug_level</code>	Establecer el nivel de detalle de anotaciones predeterminado
<code>rsed.envvars</code>	<code>keep.last.log</code>	Conserva una copia de las anotaciones previas antes del inicio/inicio de sesión.
<code>rsed.envvars</code>	<code>enable.audit.log</code>	Mantener un rastreo de auditoría de las acciones de clientes.
<code>rsed.envvars</code>	<code>enable.standard.log</code>	Escribir las secuencias <code>stdout</code> y <code>stderr</code> de la agrupación (o agrupaciones) de hebras en un archivo de anotaciones.
<code>rsed.envvars</code>	<code>DSTORE_TRACING_ON</code>	Habilitar anotaciones de acciones de DataStore.
<code>rsed.envvars</code>	<code>DSTORE_MEMLOGGING_ON</code>	Habilitar anotaciones de uso de memoria de DataStore.
Mandato de operador	<code>modify rsecommlog &lt;level&gt;</code>	Cambiar dinámicamente el nivel de detalle de anotaciones de <code>rsecomm.log</code>
Mandato de operador	<code>modify rsedaemonlog &lt;level&gt;</code>	Cambiar dinámicamente el nivel de detalle de anotaciones de <code>rsedaemon.log</code>
Mandato de operador	<code>modify rseserverlog &lt;level&gt;</code>	Cambiar dinámicamente el nivel de detalle de anotaciones de <code>rseserver.log</code>
Mandato de operador	<code>modify rsestandardlog {on   off}</code>	Cambiar dinámicamente la actualización de <code>std*.log</code>
<code>rsed.envvars</code>	<code>daemon.log</code>	Vía de acceso inicial para la tarea iniciada RSE y las anotaciones de auditoría.
<code>rsed.envvars</code>	<code>user.log</code>	Vía de acceso inicial de las anotaciones de usuario.

Developer for System z, junto con el software requisito, como la Pasarela de cliente ISPF, también escribe datos temporales en `/tmp` and `/var/rdz/WORKAREA`. La

cantidad de datos escritos aquí como resultado de las acciones de usuario no es predecible, de manera que debe tener mucho espacio libre en los sistemas de archivos que contienen estos directorios.

Developer for system z siempre intenta limpiar estos archivos temporales, pero se puede realizar la limpieza manual, tal como se documenta en “(Opcional) Limpieza de WORKAREA” en la página 108, en cualquier momento.

---

## Definiciones de recursos clave

### /etc/rdz/rsed.envvars

RSE, Java y z/OS UNIX utilizan las variables de entorno definidas en rsed.envvars. El archivo de ejemplo que viene con Developer for System z está destinado a instalaciones pequeñas-medias que no requieren los componentes adicionales de Developer for System z. “rsed.envvars, archivo de configuración de RSE” en la página 32 describe cada variable definida en el archivo de ejemplo, donde las siguientes variables requieren especial atención:

**\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Xms128m -Xmx256m"**

Establecer el tamaño inicial (Xms) y máximo (Xmx) de la memoria dinámica. Los valores predeterminados son 128M y 256M respectivamente. Cámbielo para aplicar los valores de tamaño de almacenamiento dinámico deseados. Si esta directiva tiene caracteres de comentario, se utilizarán los valores predeterminados de Java, que son 4M y 512M respectivamente (1M y 64M para Java 5.0).

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Dmaximum.clients=60"**

Número máximo de clientes a los que proporciona servicios una agrupación de hebras. El valor predeterminado es 60. Descomente y personalice este valor para limitar el número de clientes por agrupación de hebras. Tenga en cuenta que puede que otros límites impidan que RSE llegue a este límite.

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Dmaximum.threads=1000"**

Cantidad máxima de hebras activas en una agrupación de hebras para permitir clientes nuevos. El valor predeterminado es 1000. Descomente y personalice este valor para limitar el número de clientes por agrupación de hebras según el número de hebras que se estén utilizando. Tenga en cuenta que cada conexión de cliente utiliza varias hebras (16 o más) y que otros límites pueden impedir que RSE llegue a este límite.

**Nota:** Este valor debe ser inferior al valor de MAXTHREADS y MAXTHREADTASKS en SYS1.PARMLIB(BPXPRMxx).

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Dminimum.threadpool.process=10"**

Número mínimo de agrupaciones de hebras activas. El valor predeterminado es 1. Descomente y personalice este valor para iniciar como mínimo el número de procesos de agrupaciones de hebras indicado. Los procesos de agrupaciones de hebras se utilizan para el equilibrio de carga de las hebras del servidor RSE. Se inician más procesos nuevos cuando estos son necesarios. Iniciar procesos nuevos ayuda a evitar los retrasos de conexión pero utiliza más recursos durante momentos desocupados.

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Dmaximum.threadpool.process=100"**

Número máximo de agrupaciones de hebras activas. El valor predeterminado es 100. Descomente y personalice este valor para limitar el número de procesos de procesos de agrupaciones de hebras. Los procesos de agrupaciones de

hebras se utilizan para el equilibrio de carga de las hebras del servidor RSE, por lo que, al limitarlos, se limitará la cantidad de conexiones de cliente activas.

## **SYS1.PARMLIB(BPXPRMxx)**

RSE es una aplicación Java, lo que significa que está activo en el entorno z/OS UNIX. Ello hace que BPXPRMxx se convierta en un miembro parmlib crucial, ya que contiene los parámetros que controlan el entorno z/OS UNIX y los sistemas de archivos. BPXPRMxx se describe en el manual *MVS Initialization and Tuning Reference* (SA22-7592). Se conoce que las siguientes directivas afectan a Developer for System z:

### **MAXPROCSYS(nnnnn)**

Especifica el número máximo de procesos que el sistema permite.

Rango del valor: nnnnn es un valor decimal del 5 al 32767.

Valor predeterminado: 900

### **MAXPROCUSER(nnnnn)**

Especifica el número máximo de procesos que un solo ID de usuario de z/OS UNIX puede tener activos simultáneamente, independientemente de cómo se crearon los procesos.

Rango del valor: nnnnn es un valor decimal del 3 al 32767.

Valor predeterminado: 25

#### **Nota:**

- Todos los procesos RSE utilizan el mismo ID de usuario de z/OS UNIX (el del usuario asignado al daemon RSE), ya que todos los clientes se ejecutan como hebras dentro de los procesos RSE.
- Este valor también se puede establecer con la variable PROCUSERMAX en el segmento de perfil de seguridad OMVS del usuario asignado a la tarea iniciada RSED.

### **MAXTHREADS(nnnnnn)**

Especifica el número máximo de hebras pthread\_created, incluyendo las que están en ejecución, en cola y de las que se ha salido pero que no se han desconectado, que un único proceso puede tener activas simultáneamente. Especificar un valor de 0 impide que las aplicaciones puedan utilizar pthread\_create.

Rango del valor: nnnnnn es un valor decimal del 0 al 100000.

Valor predeterminado: 200

#### **Nota:**

- Cada cliente utiliza, como mínimo, 16 hebras dentro del proceso de agrupaciones de hebras RSE, y hay varios clientes activos dentro del proceso.
- Este valor también se puede establecer con la variable THREADSMAX en el segmento de perfil de seguridad OMVS del usuario asignado a la tarea iniciada RSED. Cuando se establece, el valor THREADSMAX se utiliza tanto para MAXTHREADS como para MAXTHREADTASKS.

### **MAXTHREADTASKS(nnnnn)**

Especifica el número máximo de tareas de MVS que un único proceso puede tener activas simultáneamente para las hebras pthread\_created.

Rango del valor: nnnnn es un valor decimal del 0 al 32768.  
Valor predeterminado: 1000

**Nota:**

- Cada hebra activa tiene una tarea de MVS (TCB, bloque de control de tareas).
- Cada tarea de MVS simultánea necesita almacenamiento adicional, y parte de este deberá estar por debajo de la línea de 16MB.
- Cada cliente utiliza, como mínimo, 16 hebras dentro del proceso de agrupaciones de hebras RSE, y hay varios clientes activos dentro del proceso.
- Este valor también se puede establecer con la variable THREADSMAX en el segmento de perfil de seguridad OMVS del usuario asignado a la tarea iniciada RSED. Cuando se establece, el valor THREADSMAX se utiliza tanto para MAXTHREADS como para MAXTHREADTASKS.

**MAXUIDS(nnnnn)**

Especifica el número máximo de ID de usuario de z/OS UNIX (UID) que pueden funcionar simultáneamente.

Rango del valor: nnnnn es un valor decimal del 1 al 32767.  
Valor predeterminado: 200

**MAXASSIZE(nnnnn)**

Especifica los valores de recurso RLIMIT\_AS que se establecerán como valores iniciales para los procesos nuevos. RLIMIT\_AS indica el tamaño de región del espacio de direcciones.

Rango del valor: nnnnn es un valor decimal de 10485760 (10 Megabytes) a 2147483647 (2147 Megabytes)  
Valor predeterminado: 209715200 (200 Megabytes)

**Nota:**

- Este valor se debe establecer como 2G.
- Este valor también se puede establecer con la variable ASSIZEMAX en el segmento de perfil de seguridad OMVS del usuario asignado a la tarea iniciada RSED.

**MAXFILEPROC(nnnnnn)**

Especifica el número máximo de descriptores para archivos, sockets, directorios, y cualquier otro objeto del sistema de archivos que un único proceso puede tener activos o asignados simultáneamente.

Rango del valor: nnnnnn es un valor decimal del 3 al 524287.  
Valor predeterminado: 64000

**Nota:**

- Una agrupación de hebras tiene todas las hebras de cliente en un único proceso.
- Este valor también se puede establecer con la variable FILEPROCMAX en el segmento de perfil de seguridad OMVS del usuario asignado a la tarea iniciada RSED.

**MAXMMAPAREA(nnnnn)**

Especifica la cantidad máxima de espacio de almacenamiento de espacios de

datos (en páginas) que se puede asignar para correlaciones de memoria de archivos de z/OS UNIX. El almacenamiento no se asigna hasta que la correlación de memoria no está activa.

Rango del valor: nnnnn es un valor decimal del 1 al 16777216.  
Valor predeterminado: 40960

**Nota:** Este valor también se puede establecer con la variable MMAPAREAMAX en el segmento de perfil de seguridad OMVS del usuario asignado a la tarea iniciada RSED.

Utilice el mandato de operador **SETOMVS** o **SET OMVS** para aumentar o disminuir dinámicamente (hasta la próxima IPL) el valor de cualquiera de las variables BPXPRMxx anteriores. Para realizar un cambio permanente, edite el miembro BPXPRMxx que se utilizará para las IPL. Consulte la publicación *MVS System Commands* (SA22-7627) para obtener más información sobre estos mandatos de operador.

Las definiciones siguientes son subparámetros de la sentencia NETWORK.

#### **MAXSOCKETS(nnnnnnnn)**

Especifica el número máximo de sockets soportados por este sistema de archivos para esta familia de direcciones. Este es un parámetro opcional.

Rango del valor: nnnnnnnn es un valor decimal del 0 al 16777215.  
Valor predeterminado: 100

#### **INADDRANYCOUNT(nnnn)**

Especifica el número de puertos que el sistema reserva para utilizar con el puerto PORT 0, enlaces INADDR\_ANY, empezando por el número de puerto especificado en el parámetro INADDRANYPORT. Este valor solo se necesita para CINET (varias pilas TCP/IP).

Rango del valor: nnnn es un valor decimal del 1 al 4000.  
Valor predeterminado: si no se especifica ni INADDRANYPORT ni INADDRANYCOUNT, el valor predeterminado para INADDRANYCOUNT es 1000.  
De lo contrario, no se reservará ningún puerto (0).

---

## **Definiciones de varios recursos**

### **Tarjeta EXEC del servidor JCL**

Se recomienda añadir las siguientes definiciones a la tarjeta EXEC del JCL de los servidores de Developer for System z.

#### **REGION=0M**

Se recomienda REGION=0M para las tareas iniciadas del daemon RSE y el Supervisor de trabajos de JES, RSED y JMON respectivamente. Con ello, el tamaño del espacio de direcciones está únicamente limitado por el almacenamiento privado disponible, o por las salidas del sistema IEFUSI o IEALIMIT. Tenga en cuenta que IBM recomienda encarecidamente no utilizar estas salidas para los espacios de direcciones de z/OS UNIX, como el daemon RSE.

#### **TIME=NOLIMIT**

Se recomienda utilizar TIME=NOLIMIT para todos los servidores de Developer for System z. Ello es debido a que el tiempo de la CPU de todos los clientes de Developer for System z se acumula en los espacios de direcciones del servidor.

## FEK.#CUST.PARMLIB(FEJJCNFG)

El Supervisor de trabajos JES utiliza las variables de entorno definidas en FEJJCNFG. El archivo de ejemplo que viene con Developer for System z está destinado a instalaciones pequeñas-medias. “FEJJCNFG, archivo de configuración del supervisor de trabajos JES” en la página 27 describe cada variable definida en el archivo de ejemplo, donde las siguientes variables requieren especial atención:

### MAX\_THREADS

Número máximo de usuarios que pueden utilizar un supervisor de trabajos JES en un momento dado. El valor predeterminado es 200. El valor máximo es 2147483647. Si aumenta este número, es posible que también deba aumentar el tamaño del espacio de direcciones del supervisor de trabajos JES.

## SYS1.PARMLIB(IEASYSxx)

IEASYSxx contiene los parámetros de sistema y se describe en el manual *MVS Initialization and Tuning Reference* (SA22-7592). Se conoce que las siguientes directivas afectan a Developer for System z:

### MAXUSER=nnnnn

Este parámetro especifica un valor que, en la mayoría de los casos, el sistema utiliza para limitar los trabajos y tareas iniciadas que se puede ejecutar simultáneamente durante una IPL determinada.

Rango del valor: nnnnn es un valor decimal de 0 a 32767. Tenga en cuenta que la suma de los valores especificados para los parámetros de sistema MAXUSER, y RSVNONR no puede superar 32767.

Valor predeterminado: 255

## SYS1.PARMLIB(IVTPRMxx)

IVTPRMxx establece los parámetros para el Communication Storage Manager (CSM) y se describe en el manual *MVS Initialization and Tuning Reference* (SA22-7592). Se conoce que las siguientes directivas afectan a Developer for System z:

### FIXED MAX(maxfix)

Define la cantidad máxima de almacenamiento dedicado a almacenamientos intermedios de CSM fijos.

Rango del valor: maxfix es un valor de 1024K a 2048M.

Valor predeterminado: 100M

### ECSA MAX(maxecsa)

Define la cantidad máxima de almacenamiento dedicado a almacenamientos intermedios de CSM ECSA.

Rango del valor: maxecsa es un valor de 1024K a 2048M.

Valor predeterminado: 100M

## SYS1.PARMLIB(ASCHPMxx)

El miembro parmlib de ASCHPMxx contiene información de planificación para el planificador de transacciones ASCH y se describe en el manual *MVS Initialization and Tuning Reference* (SA22-7592). Se conoce que las siguientes directivas afectan a Developer for System z:

### MAX(nnnnn)

Un parámetro opcional de la definición CLASSADD que especifica el número

máximo de iniciadores de transacciones APPC permitidos para una clase concreta de iniciadores de transacciones. Una vez se alcanza este límite, no se crean espacios de direcciones nuevos y las peticiones entrantes se dejan en cola hasta que los espacios de direcciones existentes del indicador queden disponibles. El valor no debe superar el número máximo de espacios de direcciones permitidos por la instalación, y debe tener también en cuenta los productos del sistema que también necesitarán espacios de direcciones.

Rango del valor: nnnnn es un valor decimal del 1 al 64000.

Valor predeterminado: 1

**Nota:** Si utiliza la APPC para iniciar el Servicio de mandatos TSO, la clase de transacción utilizada deberá tener suficientes iniciadores de transacción para permitir un iniciador para cada usuario simultáneo de Developer for System z.

## Supervisión

Dado que las cargas de trabajo pueden cambiar la necesidad de los recursos del sistema, es necesario supervisar el sistema regularmente para medir el uso de recursos, de manera que se pueda ajustar Rational Developer for System z y las configuraciones del sistema en respuesta a los requisitos de los usuarios. Los siguientes mandatos se pueden utilizar como ayuda en este proceso de supervisión.

### Supervisar RSE

Las agrupaciones de hebras RSE son el punto focal para la actividad de usuarios en Developer for System z y, por ello, requieren supervisión para que su uso sea el óptimo. Se puede consultar el daemon RSE para obtener información que no se puede reunir con las herramientas de supervisión de sistemas normales.

- Utilice las herramientas de supervisión del sistema normales, como RMF, para reunir datos específicos del espacio de direcciones, por ejemplo, el almacenamiento real utilizado y el tiempo de la CPU. Si no dispone de una herramienta de supervisión par ello, se puede reunir la información básica con herramientas como la vista SDSF DA o TASID (una herramienta de información del sistema tal cual disponible en el sitio Web de ISPF “Soporte y descargas”).
- Durante el inicio, el daemon RSE informa del tamaño del espacio de direcciones disponible y del tamaño del almacenamiento dinámico Java con el mensaje de consola FEK004I.

FEK004I Daemon Rse: Tamaño máx. almacenamiento dinámico=65MB y Tamaño AS privado=1,959MB

- El mandato de operador **MODIFY RSED,APPL=DISPLAY PROCESS** muestra los procesos de agrupaciones de hebras RSE. El campo “Uso de memoria” muestra qué cantidad del almacenamiento dinámico Java definido se está utilizando realmente. Consulte la sección Capítulo 8, “Mandatos de operador”, en la página 125 para obtener más información sobre este mandato.

```
f rsed,appl=d p
BPXM023I (STCRSE)
ProcessId(16777456) Uso de memoria(33%) Clientes(4) Oden(1)
```

Cuando se utiliza la opción **DETAIL** del mandato de modificación **DISPLAY PROCESS**, se proporciona más información:

```
f rsed,appl=d p,detail
BPXM023I (STCRSE)
ProcessId(33555087) ASId(002E) JobName(RSED8) Order(1)
PROCESS LIMITS:  CURRENT  HIGHWATER  LIMIT
JAVA HEAP USAGE(%)  10        56        100
```

CLIENTS	0	25	60
MAXFILEPROC	83	103	64000
MAXPROCUSER	97	99	200
MAXTHREADS	9	14	1500
MAXTHREADTASKS	9	14	1500

## Supervisar z/OS UNIX

La mayoría de límites de z/OS UNIX que afectan a Developer for System z se pueden visualizar con mandatos de operador. Algunos mandatos muestran incluso el uso simultáneo y la marca de límite superior para un límite concreto. Consulte la publicación *MVS System Commands* (SA22-7627) para obtener más información sobre estos mandatos.

- La directiva LIMMSG(ALL) de SYS1.PARMLIB(BPXPRMxx) hace que z/OS UNIX muestre los mensajes de consola (BPXI040I) cuando se está a punto de alcanzar alguno de los límites de parmlib. El valor predeterminado de LIMMSG es NONE, que inhabilita la función. Utilice el mandato de operador **SETOMVS LIMMSG=ALL** para activar dinámicamente esta función (hasta la próxima IPL). Consulte el manual *MVS Initialization and Tuning Reference* (SA22-7592) para obtener más información sobre esta directiva.
- El mandato de operador **DISPLAY OMVS,OPTIONS** muestra los valores actuales de las directivas z/OS UNIX que se pueden establecer dinámicamente.

```
d omvs,o
BPX0043I 13.10.16 DISPLAY OMVS 066
OMVS      000D ETC/INIT WAIT  OMVS=(M7)
CURRENT UNIX CONFIGURATION SETTINGS:
MAXPROCSYS      =      256    MAXPROCUSER      =      16
MAXFILEPROC     =      256    MAXFILESIZE       = NOLIMIT
MAXCPUPTIME     =      1000   MAXUIDS          =      200
MAXPTYS         =      256
MAXMAPAREA      =      256    MAXASSIZE         = 209715200
MAXTHREADS      =      200    MAXTHREADTASKS    =      1000
MAXCORESIZE     = 4194304    MAXSHAREPAGES     =      4096
IPCMSGQBYTES    = 2147483647 IPCMSGQMNUM      =     10000
IPCMSGNIDS      =      500    IPCSEMNIIDS      =      500
IPCSEMNIOPS     =      25     IPCSEMNISEMS    =     1000
IPCshmmpages    =     25600   IPCshmniids    =      500
IPCshmNSEGS     =      500    IPCshmSPAGES     =    262144
SUPERUSER       = BPXROOT    FORKCOPY         = COW
STEPLIBLIST     =
USERDALIASTABLE=
SERV_LINKLIB    = POSIX.DYNSERV.LOADLIB  BPXLK1
SERV_LPALIB     = POSIX.DYNSERV.LOADLIB  BPXLK1
PRIORITYPG VALUES: NONE
PRIORITYGOAL VALUES: NONE
MAXQUEUEDSIGS   =      1000   SHRLIBRGNSIZE = 67108864
SHRLIBMAXPAGES  =      4096   VERSION         = /
SYSCALL COUNTS  = NO         TTYGROUP          = TTY
SYSPLEX         = NO         BRML SERVER        = N/A
LIMMSG          = NONE       AUTOCVT           = OFF
RESOLVER PROC   = DEFAULT
AUTHPGMLIST     = NONE
SWA             = BELOW
```

- El mandato de operador **DISPLAY OMVS,LIMITS** muestra información sobre los límites de parmlib de los Servicios de sistemas z/OS UNIX actuales, las marcas de nivel superior y el uso del sistema actual.

```
d omvs,l
BPX0051I 14.05.52 DISPLAY OMVS 904
OMVS      0042 ACTIVE        OMVS=(69)
SYSTEM WIDE LIMITS:         LIMMSG=SYSTEM
                             CURRENT HIGHWATER SYSTEM
                             USAGE     USAGE     LIMIT
```

MAXPROCSYS	1	4	256
MAXUIDS	0	0	200
MAXPTYs	0	0	256
MAXMMAPAREA	0	0	256
MAXSHAREPAGES	0	10	4096
IPCMSGNIDS	0	0	500
IPCSEMNIDS	0	0	500
IPCSHMNIDS	0	0	500
IPCSHMSPAGES	0	0	262144 *
IPCMSGQBYTES	---	0	262144
IPCMSGQNUM	---	0	10000
IPCSHMPAGES	---	0	256
SHRLIBRGNSIZE	0	0	67108864
SHRLIBMAXPAGES	0	0	4096

El mandato muestra las marcas de nivel superior y el uso actual de un proceso individual cuando se especifica también la palabra clave PID=processid.

```
d,omvs,l,pid=16777456
BPX0051I 14.06.28 DISPLAY OMVS 645
OMVS 000E ACTIVE OMVS=(76)
USER JOBNAME ASID PID PPID STATE START CT_SECS
STCRSE RSED8 007E 16777456 67109106 HF---- 20.00.56 113.914
LATCHWAITPID= 0 CMD=java -Ddaemon.log=/var/rdz/logs -
PROCESS LIMITS: LIMMSG=NONE
CURRENT HIGHWATER PROCESS
USAGE USAGE LIMIT
MAXFILEPROC 83 103 256
MAXFILESIZE --- --- NOLIMIT
MAXPROCUSER 97 99 200
MAXQUEUEDSIGs 0 1 1000
MAXTHREADS 9 14 200
MAXTHREADTASKS 9 14 1000
IPCSHMNSEGS 0 0 500
MAXCORESIZE --- --- 4194304
MAXMEMLIMIT 0 0 16383P
```

- El mandato de operador **DISPLAY OMVS,PFS** muestra información sobre cada sistema de archivos físico que forma actualmente parte de la configuración de z/OS UNIX, que incluye pilas TCP/IP.

```
d omvs,p
BPX0046I 14.35.38 DISPLAY OMVS 092
OMVS 000E ACTIVE OMVS=(33)
PFS CONFIGURATION INFORMATION
PFS TYPE DESCRIPTION ENTRY MAXSOCK OPNSOCK HIGHUSED
TCP SOCKETS AF_INET EZBPFINI 50000 244 8146
UDS SOCKETS AF_UNIX BPXTUINT 64 6 10
ZFS LOCAL FILE SYSTEM IOEFSCM
14:32.00 RECYCLING
HFS LOCAL FILE SYSTEM GFUAINIT
BPXFTCLN CLEANUP DAEMON BPXFTCLN
BPXFTSYN SYNC DAEMON BPXFTSYN
BPXFPINT PIPE BPXFPINT
BPXFCSIN CHAR SPECIAL BPXFCSIN
NFS REMOTE FILE SYSTEM GFSCINIT
PFS NAME DESCRIPTION ENTRY STATUS FLAGS
TCP41 SOCKETS EZBPFINI ACT CD
TCP42 SOCKETS EZBPFINI ACT
TCP43 SOCKETS EZBPFINI INACT SD
TCP44 SOCKETS EZBPFINI INACT
PFS PARM INFORMATION
HFS SYNCDEFAULT(60) FIXED(50) VIRTUAL(100)
CURRENT VALUES: FIXED(55) VIRTUAL(100)
NFS biod(6)
```

- El mandato de operador **DISPLAY OMVS,PID=processid** muestra la información de hebras de un proceso específico.

```

d omvs,pid=16777456
BPX0040I 15.30.01 DISPLAY OMVS 637
OMVS      000E ACTIVE              OMVS=(76)
USER      JOBNAME  ASID            PID      PPID STATE   START    CT_SECS
STCRSE    RSED8    007E    16777456    67109106 HF---- 20.00.56 113.914
LATCHWAITPID= 0 CMD=java -Ddaemon.log=/var/rdz/logs -
THREAD_ID  TCB@    PRI_JOB  USERNAME  ACC_TIME SC  STATE
0E08A00000000000 005E6DF0 OMVS      .927 RCV  FU
0E08F00000000001 005E6C58          .001 PTX  JYNV
0E09300000000002 005E6AC0          7.368 PTX  JYNV
0E0CB00000000008 005C2CF0 OMVS      1.872 SEL  JFNV
0E1920000000003CE 005A0B70 OMVS      14.088 POL  JFNV
0E18D0000000003CF 005A1938          .581 SND  JYNV

```

## Supervisar la red

Cuando se admite que un número grande de clientes se conecten al host, no sólo Developer for System z debe ser capaz de manejar la carga de trabajo, sino que la infraestructura de red también debe serlo. La gestión de redes es un tema amplio y bien documentado que no está dentro del ámbito de la documentación de Developer for System z. Por ello, únicamente se facilitan los siguientes puntos:

- El mandato de operador **DISPLAY NET,CSM** le permite supervisar el uso del almacenamiento gestionado por el Communications Storage Manager (CSM). Puede utilizar este mandato para determinar qué cantidad de almacenamiento de CSM se está utilizando para ECSA y para agrupaciones de almacenamiento de espacios de datos, tal como se documenta en *Communications Server SNA Operations* (SC31-8779).

## Supervisar los sistemas de archivos de z/OS UNIX

Developer for System z utiliza sistemas de archivos de z/OS UNIX para almacenar varios tipos de datos, como archivos de anotaciones y temporales. Utilice el mandato de z/OS UNIX **df** para ver cuántos descriptores de archivos quedan disponibles, y cuánto espacio libre queda hasta que se crea la siguiente extensión del conjunto de datos HFS o zFS subyacente.

```

$ df
Montado en      Filesystem      Disp/Total      Archivos  Estado
/tmp            (OMVS.TMP)      1393432/1396800 4294967248 Disponible
/u/ibmuser      (OMVS.U.IBMUSER) 1248/1728      4294967281 Disponible
/usr/lpp/rdz    (OMVS.FEK.HHOP760) 3062/43200     4294967147 Disponible
/var            (OMVS.VAR)      27264/31680     4294967054 Disponible

```

## Configuración de ejemplo

La siguiente configuración de ejemplo muestra la configuración necesaria para soportar estos requisitos:

- 500 conexiones de cliente simultáneas
- 300 construcciones de MVS simultáneas (trabajo por lotes)
- 200 conexiones de CARMA simultáneas (utilizando el método de inicio CRASTART)
- 3 horas de tiempo de espera de inactividad
- no permitir el uso de z/OS UNIX
- No se utilizan SCLM Developer Toolkit ni File Manager Integration
- Previsión de un uso medio de almacenamiento dinámico Java de 5 MB
- Los usuarios tienen UID de z/OS UNIX exclusivos

## Recuento de agrupaciones de hebras

De forma predeterminada, Developer for system z intenta añadir 60 usuarios a una única agrupación de hebras. Sin embargo, nuestros requisitos indican que el tiempo de espera de inactividad estará activo. Tabla 44 en la página 221 muestra que esto añadirá una hebra por cada cliente conectado. Esta hebra es una hebra temporizadora, por lo que está activa constantemente. Esto impedirá que RSE coloque a 60 usuarios en una única agrupación de hebras, dado que  $60 \times (16+1) = 1020$ , y `maximum.threads` está establecido en 1000 de forma predeterminada.

Podríamos aumentar `maximum.threads`, pero dado que el requisito es que la media sea 5 MB de almacenamiento dinámico Java por cada usuario, decidimos bajar `maximum.clients` a 50. Con esto, sigue dentro del tamaño de almacenamiento dinámico Java máximo de 256 MB predeterminado ( $5 \times 50 = 250$ ).

Con 50 clientes por agrupación de hebras y la necesidad de admitir 500 conexiones, sabemos que necesitaremos 10 espacios de direcciones de agrupaciones de hebras.

## Determinar los límites mínimos

Utilizando las fórmulas mostradas anteriormente en este capítulo y los criterios indicados al principio de esta sección, podemos determinar el uso de los recursos que se deben acomodar.

- Recuento de espacios de direcciones - máximo
$$3 + A + N \times (x + y + z) + (2 + N \times 0.01)$$
$$3 + 10 + 500 \times 1 + 200 \times 1 + 300 \times 1 + (2 + 500 \times 0.01) = 1020$$
- Recuento de espacios de direcciones - por usuario
$$x + y + z$$
$$1 + 1 + 1 = 3$$
- Recuento de procesos - máximo
$$7 + 2 \times A + N \times (x + y + z) + (10 + N \times 0.05)$$
$$7 + 2 \times 10 + 500 \times 2 + 200 \times 1 + 300 \times 0 + (10 + 500 \times 0.05) = 1562$$
- Recuento de procesos - por usuario
$$(x + y + z) + 5 \times s$$
$$(2 + 1 + 0) + 5 \times 0 = 3$$
- Recuento de hebras - agrupación de hebras RSE
$$9 + N \times (16 + x + y + z) + (20 + N \times 0.1)$$
$$9 + 60 \times (16 + 1 + 4 + 0) + (20 + 60 \times 0.1) = 1295$$
- Recuento de hebras - Supervisor de trabajos JES
$$3 + N$$
$$3 + 500 = 503$$
- ID de usuario
$$500 + 3 = 503$$

Los 3 ID de usuario extras son para STCJMON, STCLOCK y STCRSE, los ID de usuario de las tareas iniciadas de Developer for System z.

## Definir límites

Ahora que conocemos los números del uso de recursos, podemos personalizar las directivas de limitación con los valores adecuados.

- /etc/rdz/rsed.envvars
  - Xmx256m
  - no cambia
  - Dmaximum.clients=50
  - Dmaximum.threads=1000
  - no cambia
  - Dminimum.threadpool.process=10
  - Este cambio es opcional; RSE iniciará nuevas agrupaciones de hebras según sea necesario
  - DHIDE\_ZOS\_UNIX=true
  - DDSTORE\_IDLE\_SHUTDOWN\_TIMEOUT=10800000
- FEK.#CUST.PARMLIB(FEJJCNFG)
  - MAX\_THREADS=503
- SYS1.PARMLIB(BPXPRMxx)
  - MAXPROCSYS(2500)
  - mínimo de 1562, almacenamiento intermedio extra añadido para otras tareas que las de for System z
  - MAXPROCUSER(25)
  - no cambia, mínimo 3
  - MAXTHREADS(1500)
  - debe ser, como mínimo, 503 (para el Supervisor de trabajos JES) si THREADSMAX del (mínimo 1295)
  - MAXTHREADTASKS(1500)
  - debe ser, como mínimo, 503 (para el Supervisor de trabajos JES) si THREADSMAX del (mínimo 1295)
  - MAXUIDS(700)
  - mínimo de 503, almacenamiento intermedio extra añadido para otras tareas que las de for System z
  - MAXASSIZE(209715200)
  - no cambia (valor predeterminado del sistema 200 MB), utilizamos ASSIZEMAX en el se
- SYS1.PARMLIB(IEASYSxx)
  - MAXUSER=2000
  - mínimo de 1020, almacenamiento intermedio extra añadido para otras tareas que las de for System z
- Segmento OMVS del ID de usuario STCRSE
  - ASSIZEMAX(2147483647)
  - 2 GB

## Utilización de recursos de supervisor

Después de activar los límites del sistema según se explica en “Definir límites” en la página 242, podemos empezar a supervisar la utilización de recursos por parte de Developer for System z para ver si es necesario ajustar varias variables.

Figura 58 muestra la utilización de recursos después de que 495 usuarios hayan iniciado la sesión. (El ejemplo de la figura muestra sólo el inicio de sesión. No se han indicado acciones de usuario en el ejemplo.)

BPXM023I (STCRSE)

ProcessId(16779764)	Uso de memoria(10%)	Clientes(50)	Orden(1)
ProcessId(67108892)	Uso de memoria(16%)	Clientes(50)	Orden(2)
ProcessId(67108908)	Uso de memoria(10%)	Clientes(50)	Orden(3)
ProcessId(67108898)	Uso de memoria(16%)	Clientes(50)	Orden(4)
ProcessId(67108916)	Uso de memoria(16%)	Clientes(50)	Orden(5)
ProcessId(67108897)	Uso de memoria(16%)	Clientes(50)	Orden(6)
ProcessId(67108921)	Uso de memoria(16%)	Clientes(50)	Orden(7)
ProcessId(83886146)	Uso de memoria(16%)	Clientes(50)	Orden(8)
ProcessId(67108920)	Uso de memoria(16%)	Clientes(50)	Orden(9)
ProcessId(3622)	Uso de memoria(8%)	Clientes(45)	Orden(10)

Jobname	Tiempo Cpu	Almacenamiento	EXCP
-----	-----	-----	-----
JMON	1.74	43.0M	2753
LOCKD	10.05	31.9M	24621
RSED	6.65	40.1M	41780
RSED1	8.17	187.0M	76566
RSED2	13.04	184.9M	78946
RSED3	17.77	181.1M	76347
RSED4	11.63	174.9M	74638
RSED5	15.27	172.9M	72883
RSED6	13.85	180.8M	75031
RSED7	9.79	174.3M	76636
RSED8	21.59	176.1M	70583
RSED8	18.88	184.7M	76953
RSED9	9.52	189.8M	80490

Figura 58. Utilización de recursos de configuración de ejemplo

---

## Capítulo 14. 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 for System z.

Consulte las publicaciones *MVS Initialization and Tuning Guide* (SA22-7591) y *UNIX System Services Planning* (GA22-7800) para obtener más información acerca del ajuste del sistema.

---

### Utilizar sistemas de archivos zFS

zFS (sistema de archivos de zSeries) y HFS (sistema de archivos jerárquico) son sistemas de archivos UNIX que pueden utilizarse en un entorno z/OS UNIX. 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 estratégico de z/OS UNIX. La funcionalidad del HFS se ha estabilizado y las mejoras realizadas en el sistema de archivos solo serán para zFS.

Consulte la publicación *UNIX System Services Planning* (GA22-7800) para obtener más información acerca de 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 ECSA (área de almacenamiento común ampliada). 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()).

Developer for System z tiene el valor predeterminado STEPLIB=NONE codificado en rsed.envvars, como se describe en la sección dedicada al archivo de configuración rsed.envvars. 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).

---

### Mejorar el acceso a las bibliotecas del sistema

Algunas bibliotecas del sistema y algunos módulos de carga se utilizan intensivamente 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.

CEE.SCEELPA

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. Debe 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 colocar estas bibliotecas en LINKLIST, debe configurar la sentencia STEPLIB pertinente en rsed.envvars, como se describe en la sección dedicada al archivo de configuración rsed.envvars. 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 principal es el desarrollo de aplicaciones, el rendimiento también podrá mejorar si coloca el editor de enlaces en la LPA dinámica, añadiendo las líneas siguientes a SYS1.PARMLIB(PROGxx):

```
LPA ADD MODNAME(CEEBINIT,CEEBLIBM,CEEEV003,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. Consulte la publicación *Language Environment Customization* (SA22-7564) para obtener más información acerca de la colocación de otros módulos de carga de LE en la LPA dinámica. Consulte la

publicación *UNIX System Services Planning* (GA22-7800) para obtener más información acerca de la colocación de 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. Consulte la publicación *UNIX System Services Planning* (GA22-7800) para obtener más información acerca de este perfil.

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

---

## 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 for System z se puede equilibrar estableciendo los objetivos correctos para sus procesos. A continuación figuran algunas directrices generales:

- Al utilizarlo, asigne la transacción APPC a un grupo de rendimiento TSO.
- Asigne un grupo de rendimiento de tareas iniciadas (SYSSTC) a los espacios de direcciones del servidor de Developer for System z: Supervisor de trabajos JES, (JMON), Daemon de bloqueo (LOCKD), Daemon RSE (RSED) y Agrupación de hebras RSE (RSEDx).

Consulte la publicación *MVS Planning Workload Management* (SA22-7602) para obtener más información sobre este tema.

---

## 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. Consulte la publicación *Java Diagnostics Guide* (SC34-6650) para obtener más información acerca de los tamaños del almacenamiento dinámico y la recogida de basura.

De forma predeterminada, el tamaño inicial del almacenamiento dinámico de una máquina virtual (JVM) en z/OS Java 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 for 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: “Definir parámetros de inicio Java adicionales con `_RSE_JAVA_OPTS`” en la página 42.

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

---

## Opción `-Xquickstart` de Java

**Nota:** La opción `-Xquickstart` de Java sólo resulta de utilidad si utiliza el método de inicio alternativo REXEC/SSH para el servidor RSE. Este método se describe en la sección “(Opcional) Utilizar REXEC (o SSH)” en la página 102.

La opción `-Xquickstart` puede utilizarse para mejorar el tiempo de inicio de algunas aplicaciones Java. `-Xquickstart` hace que el compilador JIT (Just In Time) se ejecute con un subconjunto de optimizaciones; es decir, una compilación rápida. Esta compilación rápida permite mejorar el tiempo de inicio.

`-Xquickstart` es adecuada para aplicaciones de corta ejecución, especialmente para aquellas en las que el tiempo de ejecución no está concentrado en un pequeño número de métodos. `-Xquickstart` puede degradar el rendimiento si se utiliza en aplicaciones de larga ejecución que contienen métodos dinámicos.

Para habilitar la opción `-Xquickstart` para el servidor RSE, añada la directiva siguiente al final de `rsed.envvars`:

```
_RSE_JAVA_OPTS="$_RSE_JAVA_OPTS -Xquickstart"
```

---

## Compartimiento de clases entre las JVM

La máquina virtual Java (JVM) de IBM versión 5 y posteriores permite compartir clases de aplicación y programa de arranque entre las JVM almacenándolas en una memoria 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.

Consulte la publicación *Java SDK and Runtime Environment User Guide* para obtener más información acerca del compartimiento de clases.

## Habilitar el compartimiento de clases

Para habilitar el compartimiento de clases para el servidor RSE, añada la directiva siguiente al final de `rsed.envvars`. 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). La tercera sentencia añade los parámetros de compartimiento de clases a las opciones de inicio de Java.

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

**Nota:** Como se ha mencionado en la sección “Seguridad de la memoria caché”, todos los usuarios que utilizan 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 memoria 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 memoria caché de clases compartidas y el almacenamiento dinámico de Java, el aumento del tamaño máximo del almacenamiento dinámico de Java reducirá el tamaño de la memoria caché de clases compartidas que puede crear.

## Seguridad de la memoria 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).

Consulte la publicación *Java SDK and Runtime Environment User Guide* para obtener más información acerca de las opciones de seguridad adicionales mediante un gestor de seguridad (SecurityManager) 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 acerca de las implicaciones sobre el rendimiento y la utilización de estos parámetros, consulte las publicaciones *MVS Initialization and Tuning Reference* (SA22-7592) y *UNIX System Services Planning* (GA22-7800). Los parámetros más significativos de `BPXPRMxx` que afectan a la operación de las clases compartidas son los siguientes:

- `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 tanto, establezca IPCSHMMPAGES en un valor superior a 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 IPCSHMMPAGES en su sistema.

- IPCSEMNIDS y 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 de 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). Consulte la publicación *Java SDK and Runtime Environment User Guide* para obtener 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:**

- Los programas de utilidad de memoria caché realizan la operación necesaria en la memoria caché especificada sin iniciar la JVM, por lo que el mensaje "No se ha podido crear la máquina virtual Java." es normal.
- Una memoria caché sólo puede destruirse si todas las JVM que la utilizan se han cerrado y el usuario que emite el mandato tiene autorización suficiente.



---

## Capítulo 15. consideraciones de CICSTS

Tradicionalmente, el papel de definir recursos en CICS ha sido competencia del administrador de CICS. Ha habido cierta renuencia en permitir que el desarrollador de aplicaciones definiera recursos CICS por diversas razones:

- La mayoría de las definiciones de recurso CICS tienen muchos parámetros que, debido a su complejidad, interrelación con otras definiciones de recurso y estándares comerciales, requieren conocimientos de administración de CICS para definirlos correctamente. Las definiciones incorrectas pueden provocar resultados inesperados que pueden influir sobre toda la región CICS.
- La mayoría de establecimientos comerciales ofrecen entornos de prueba y desarrollo CICS que deben estar disponibles para el uso compartido de varios grupos de aplicaciones y desarrolladores. Muchos establecimientos comerciales cuentan con el Acuerdo de nivel de servicio para estos entornos. El cumplimiento de estos acuerdos requiere un estricto control de los entornos.

Developer for System z soluciona estos problemas al permitir que los administradores de CICS controlen los valores predeterminados de las definiciones de recursos CICS y las propiedades de visualización de un parámetro de definición de recurso CICS por medio del servidor CRD (definición de recurso CICS), que es parte del Gestor de despliegue de aplicaciones.

Por ejemplo, el administrador de CICS puede suministrar determinados parámetros de definición de recurso CICS que el desarrollador de aplicaciones no puede actualizar. Otros parámetros de definición de recurso CICS pueden ser actualizables, con o sin valores predeterminados suministrados, o el parámetro de definición de recurso CICS puede ocultarse para evitar una complejidad innecesaria.

Una vez que el desarrollador de aplicaciones está satisfecho con las definiciones de recurso CICS, éstas pueden instalarse de inmediato en el entorno de prueba CICS en ejecución, o pueden exportarse las definiciones en un manifiesto para que un administrador de CICS las edite y apruebe. El administrador de CICS puede utilizar el programa de utilidad administrativo (programa de utilidad de proceso por lotes) o la herramienta de Proceso de manifiestos para implementar cambios de definición de recurso.

**Nota:** La herramienta de Proceso de manifiestos es un plugin de IBM CICS Explorer.

Consulte la sección Capítulo 4, “(Opcional) Gestor de despliegue de aplicaciones (ADM)”, en la página 73 para obtener más información acerca de las tareas necesarias para configurar el Gestor de despliegue de aplicaciones en el sistema host.

La personalización del Gestor de despliegue de aplicaciones añade los siguientes servicios a Developer for System z:

- (en el cliente) El Explorador CICS de IBM proporciona una infraestructura basada en Eclipse para poder visualizar y manipular recursos CICS, además de permitir una mayor integración entre las herramientas CICS
- (en el cliente) El editor CRD (definiciones de recursos CICS)

- (en el host) El servidor CDR (definiciones de recursos CICS), que se ejecuta como una aplicación CICS

El servidor CRD (definición de recurso CICS) del Gestor de despliegue de aplicaciones consta del propio servidor CRD, un repositorio CRD, las definiciones de recursos CICS asociadas y, al utilizar la interfaz de servicios Web, archivos de enlace de servicio Web y un manejador de mensajes de conducto de ejemplo. El servidor CRD debe ejecutarse en una WOR (Web Owning Region), a la que se hace referencia en la documentación de Developer for System z como región de conexión primaria CICS.

Consulte el Information Center de Developer for System z Information Center (<http://publib.boulder.ibm.com/infocenter/ratdevz/v7r6/index.jsp>) para obtener más información sobre los servicios del Gestor de despliegue de aplicaciones disponibles en el release actual de Developer for System z.

---

## RESTful versus Servicio Web

CICS Transaction Server proporciona en la versión 4.1 y superior soporte para una interfaz HTTP diseñada utilizando los principios de la transferencia de estado representativo (RESTful). Esta interfaz RESTful es ahora la interfaz CICSTS estratégica para las aplicaciones de cliente. La interfaz de servicio web anterior se ha estabilizado, y las mejoras serán únicamente para la interfaz RESTful.

El Gestor de despliegue de aplicaciones sigue esta sentencia de dirección y necesita el servidor CRD de RESTful para todos los servicios que son nuevos Developer for System versión 7.6 o superior.

Las interfaces RESTful y de servicio Web puede estar activas simultáneamente en una única región CICS, si lo desea. En este caso, habrá dos servidores CRD activos en la región. Ambos servidores compartirán el mismo repositorio CRD. Tenga en cuenta que CICS emitirá algunos avisos sobre definiciones duplicadas cuando se defina la segunda interfaz en la región.

---

## Comparación entre regiones de conexión primarias y no primarias

Un entorno de prueba CICS puede constar de varias regiones MRO (Opción multirregión) conectadas. Con el tiempo, se han utilizado denominaciones no oficiales para categorizar estas regiones. Son designaciones típicas TOR (región propietaria de terminal, WOR (región propietaria Web), AOR (región propietaria de aplicación) y DOR (región propietaria de datos).

Una WOR (región propietaria Web) se utiliza para implementar el soporte de servicios Web de CICS, y el servidor CRD (definición de recurso CICS) del Gestor de despliegue de aplicaciones debe ejecutarse en esta región. Esta región se conoce en el gestor de despliegue de aplicaciones como la región de conexión primaria CICS. El cliente CRD implementa una conexión de servicio Web con la región de conexión primaria CICS.

Las regiones de conexión no primaria CICS son todas las demás regiones a las que el servidor CRD puede dar servicio. Este servicio incluye la visualización de recursos mediante IBM CICS Explorer y la definición de recursos mediante el editor de definiciones de recurso CICS.

Si se utiliza BAS (Business Application Services) de CICSplex SM para gestionar las definiciones de recursos CICS de la región de conexión primaria CICS, el servidor CRD puede dar servicio a todas las demás regiones CICS gestionadas por BAS.

Las regiones CICS no gestionadas por BAS requieren cambios adicionales para que el servidor CRD pueda darles servicio.

---

## Anotaciones de instalación de recursos CICS

Las acciones realizadas por el servidor CRD en los recursos CICS se anotan en la cola TD CSDL de CICS, que generalmente señala hacia a la DD MSGUSR de la región CICS.

Si se utiliza Business Application Services (BAS) de CICSplex SM para gestionar las definiciones de recurso CICS, la directiva de CICSplex SM EYUPARM BASLOGMSG debe establecerse en (YES) para poder crear las anotaciones.

---

## Seguridad del Gestor de despliegue de aplicaciones

### Seguridad del repositorio de CRD

El conjunto de datos VSAM del repositorio del servidor CRD contiene todas las definiciones de recurso predeterminadas y, por tanto, debe protegerse contra actualizaciones, pero los desarrolladores deben poder leer los valores almacenados en él. Consulte “Definir perfiles de conjunto de datos” en la página 180 para obtener mandatos RACF de ejemplo para proteger el repositorio de CRD.

### seguridad de conducto

Cuando CICS recibe un mensaje SOAP a través de la interfaz de servicios Web, un conducto lo procesa. Un conducto es un conjunto de manejadores de mensajes que se ejecutan por orden. CICS lee el archivo de configuración del conducto para determinar qué manejadores de mensajes deben invocarse en el conducto. Un manejador de mensajes es un programa en el que pueden realizarse procesos especiales de solicitudes y respuestas de servicios Web.

El Gestor de despliegue de aplicaciones suministra un archivo de configuración de conducto de ejemplo que especifica la invocación de un manejador de mensajes y un programa de proceso de cabeceras SOAP.

El manejador de mensajes de conducto (ADNTMSGH) se utiliza para la seguridad, mediante el proceso del ID de usuario y la contraseña de la cabecera SOAP. El archivo de configuración de conducto de ejemplo hace referencia a ADNTMSGH, que, por lo tanto, se debe colocar en la concatenación RPL de CICS.

### Seguridad de transacciones

CPIH es el ID de transacción predeterminado bajo el que se ejecutará una aplicación invocada por un conducto. Generalmente, CPIH se establece para un nivel de autorización mínimo.

Developer for System z suministra varias transacciones que el servidor CRD utiliza al definir y consultar recursos CICS. Estos ID de transacción se establecen mediante el servidor CRD, dependiendo de la operación solicitada. Hallará más información sobre la personalización de ID de transacción en la sección Capítulo 4, “(Opcional) Gestor de despliegue de aplicaciones (ADM)”, en la página 73.

Transacción	Descripción
ADMS	Para las solicitudes de la herramienta de Proceso de manifiestos para cambiar recursos CICS. Normalmente, está destinado a los administradores de CICS. Esta transacción requiere un alto nivel de autorización.
ADMI	Para las peticiones que definen, instalan o desinstalan recursos CICS. Esta transacción puede requerir un nivel de autorización medio, dependiendo de las políticas establecidas en la ubicación.
ADMR	Para todas las demás peticiones que recuperan información de recursos o de entorno de CICS. Esta transacción puede requerir un nivel de autorización mínimo, dependiendo de las políticas establecidas en la ubicación.

Algunas de las solicitudes de definiciones de recurso realizadas por las transacciones del servidor CRD, o todas ellas, deben protegerse. Como mínimo, los mandatos de actualización (actualizar parámetros de servicio Web predeterminados, parámetros de descriptor predeterminados y enlace de nombre de archivo a nombre de conjunto de datos) deben protegerse para impedir nadie, excepto los administradores de CICS, emita estos mandatos utilizados para establecer valores predeterminados de recursos globales.

Cuando se conecta la transacción, la comprobación de la seguridad de recursos CICS, si está habilitada, se asegura de que el ID de usuario tiene autorización para ejecutar el ID de transacción.

La comprobación de recursos está controlada por la opción RESSEC en la transacción que se ejecuta, por el parámetro de inicialización del sistema RESSEC y, para el servidor CRD, por el parámetro de inicialización del sistema XPCT.

La comprobación de recursos sólo tiene lugar si el parámetro de inicialización del sistema XPCT tiene un valor que no es NO y la opción RESSEC de la definición de TRANSACTION es YES o el valor del parámetro de inicialización del sistema RESSEC es ALWAYS.

Los siguientes mandatos de RACF ofrecen un ejemplo de cómo pueden protegerse las transacciones del servidor CRD. Consulte la publicación *RACF Security Guide for CICSTS* para obtener más información acerca de la definición de la seguridad de CICS.

- RALTER GCICSTRN SYSADM UACC(NONE) ADDMEM(ADMS)
- PERMIT SYSADM CLASS(GCICSTRN) ID(#admindcics)
- RALTER GCICSTRN DEVELOPER UACC(NONE) ADDMEM(ADMI)
- PERMIT DEVELOPER CLASS(GCICSTRN) ID(#desarrolladorcics)
- RALTER GCICSTRN ALLUSER UACC(READ) ADDMEM(ADMR)
- SETROPTS RACLIST(TCICSTRN) REFRESH

## Comunicación cifrada con SSL

El cifrado SSL de la secuencia de datos está soportada cuando el cliente Gestor de despliegue de aplicaciones utiliza la interfaz de servicios Web para invocar el

servidor CRD. La utilización de SSL para esta comunicación está controlada por la palabra clave SSL(YES) en la definición de CICSTS TCPIP SERVICE, tal como se documenta en *RACF Security Guide for CICSTS*.

## Seguridad de recursos

CICSTS proporciona la capacidad de proteger recursos, además de los mandatos para manipularlos. Algunas acciones del Gestor de despliegue de aplicaciones pueden fallar si la seguridad está activada pero no está completamente configurada (por ejemplo, otorgar permisos para la manipulación de tipos de recursos nuevos).

Cuando falle una función en el Gestor de despliegue de aplicaciones, examine las anotaciones de CICS para buscar mensajes similares al siguiente, y realice la acción correctiva, tal como se documenta en *RACF Security Guide for CICSTS*.

DFHXS1111 %fecha %hora %id\_de\_aplicación %id\_de\_transición Violación de la seguridad por el usuario %id\_de\_usuario en la red %nombre\_de\_puerto del recurso %recurso de la clase %nombre\_de\_clase. Los códigos SAF son (X'safresp',X'safreas'). Los códigos ESM son (X'esmrsp',X'esmrreas').

---

## Programa de utilidad administrativo

Developer for System z suministra el programa de utilidad administrativo que permite a los administradores de CICS proporcionar los valores predeterminados para las definiciones de recursos CICS. Estos valores predeterminados pueden ser de sólo lectura o pueden ser editables para los desarrolladores de aplicaciones.

El programa de utilidad administrativo ofrece las funciones siguientes:

- Nombre CICSplex para los entornos de prueba gestionados por CICSplex
- Nombre del grupo intermedio de CICSplex SM
- Valor de la norma de exportación de manifiestos
- Permisos de visualización y valores predeterminados de atributo de recurso CICS
- Enlace lógico a físico CICS utilizado para las definiciones de conjunto de datos VSAM

El programa de utilidad administrativo se invoca mediante el trabajo de ejemplo ADNJSPAU del conjunto de datos FEK.#CUST.JCL. La utilización de este programa de utilidad requiere acceso de actualización (UPDATE) al repositorio de CRD.

ADNJSPAU se encuentra en FEK.#CUST.JCL, a menos que el programador de sistemas z/OS haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Encontrará más detalles en: "Configuración de la personalización" en la página 15.

**Nota:** Antes de ejecutar el trabajo ADNJSPAU, debe cerrarse el repositorio CRD de CICS. El repositorio puede abrirse de nuevo una vez finalizado el trabajo. Por ejemplo, después de iniciar la sesión en CICS, especifique los mandatos siguientes para cerrar y abrir el archivo, respectivamente:

- CEMT S FILE(ADNREPF0) CLOSED
- CEMT S FILE(ADNREPF0) OPEN

Se utilizan sentencias de control de entrada para actualizar el repositorio del CRD para un entorno de prueba CICS, para las que se aplican las siguientes normas generales de sintaxis:

- Un asterisco en la posición 1 indica una línea de comentario.
- Un mandato DEFINE debe empezar en la posición 1, seguido por un sólo espacio, seguido de una palabra clave válida, como por ejemplo TRANSACTION.
- Una palabra clave debe ir seguida inmediatamente de un valor de palabra clave. No se permiten espacios intercalados. La única excepción la constituyen las palabras clave de permiso de visualización UPDATE, PROTECT y HIDDEN, que no tienen valores.
- Los valores de palabra clave se especifican entre paréntesis.
- Una palabra clave y su valor deben especificarse en una sola línea.

Las siguientes definiciones de ejemplo siguen la estructura de los mandatos DFHCSDUP, tal como se define en *CICS Resource Definition Guide for CICSTS*. La única diferencia es la inserción de las siguientes palabras clave de permiso de visualización utilizadas para agrupar los valores de atributo en tres conjuntos de permisos:

UPDATE	Un desarrollador de aplicaciones podrá actualizar los atributos situados a continuación de esta palabra clave mediante Developer for System z. Este es también el valor predeterminado para los atributos omitidos.
PROTECT	Los atributos situados a continuación de esta palabra clave se visualizarán, pero el desarrollador de aplicaciones no podrá actualizarlos mediante Developer for System z.
HIDDEN	Los atributos situados a continuación de esta palabra clave no se visualizarán, y el desarrollador de aplicaciones no podrá actualizarlos mediante Developer for System z.

Consulte el siguiente ejemplo de código de ADNJSPAU.

```

//TRABAJO ADNJSPAU <PARÁMETROS TRABAJO>
//*
//ADNSPAU EXEC PGM=ADNSPAU,REGION=1M
//STEPLIB DD DISP=SHR,DSN=FEK.SFEKLOAD
//ADMREP DD DISP=OLD,DSN=FEK.#CUST.ADNREPF0
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
*
* Parámetros de CICSplex SM
*
DEFINE CPSMNAME( )
*DEFINE STAGINGGROUPNAME(ADMSTAGE)
*
* Regla de exportación de manifiestos
*
DEFINE MANIFESTEXPORTRULE(installOnly)
*
* Valores predeterminados de definición de recurso CICS
* Los atributos omitidos toman por omisión el valor UPDATE.
*
* Atributos predeterminados de DB2TRAN
*
DEFINE DB2TRAN()
    UPDATE DESCRIPTION()
    ENTRY()
    TRANSID()
*
* Atributos predeterminados de DOCTEMPLATE
*
DEFINE DOCTEMPLATE()
    UPDATE DESCRIPTION()
    TEMPLATENAME()
    FILE() TSQUEUE() TDQUEUE() PROGRAM() EXITPGM()
    DDNAME(DFHHTML) MEMBERNAME()
    HFSFILE()
    APPENDCRLF(YES) TYPE(EBCDIC)
*
* Atributos predeterminados de File
*
DEFINE FILE()
    UPDATE DESCRIPTION()
    RECORDSIZE() KEYLENGTH()
    RECORDFORMAT(V) ADD(NO)
    BROWSE(NO) DELETE(NO) READ(YES) UPDATE(NO)
    REMOTESYSTEM() REMOTENAME()
    PROTECT DSNNAME() RLSACCESS(NO) LSRPOOLID(1) STRINGS(1)
    STATUS(ENABLED) OPENTIME(FIRSTREF)
    DISPOSITION(SHARE) DATABUFFERS(2) INDEXBUFFERS(1)
    TABLE(NO) MAXNUMRECS(NOLIMIT)
    READINTEG(UNCOMMITTED) DSNSHARING(ALLREQS)
    UPDATEMODEL(LOCKING) LOAD(NO)
    JNLREAD(NONE) JOURNAL(NO)
    JNLSYNCREAD(NO) JNLUPDATE(NO)
    JNLADD(NONE) JNLSYNCSWRITE(YES)
    RECOVERY(NONE) FWDRECOVLOG(NO)
    BACKUPTYPE(STATIC)
    PASSWORD() NSRGROUP()
    CFDTPOOL() TABLENAME()

```

Figura 59. ADNJSPAU - programa de utilidad administrativo de CICSTS (Parte 1 de 3)

```

*
*   Atributos predeterminados de Mapset
*
DEFINE MAPSET()
    UPDATE  DESCRIPTION()
    PROTECT RESIDENT(NO) STATUS(ENABLED)
           USAGE(NORMAL) USELPACOPY(NO)
** Atributos predeterminados de Processtype
*
DEFINE PROCESSTYPE()
    UPDATE  DESCRIPTION()
           FILE(BTS)
    PROTECT STATUS(ENABLED)
           AUDITLOG() AUDITLEVEL(OFF)
*
*   Atributos predeterminados de Program
*
DEFINE PROGRAM()
    UPDATE  DESCRIPTION()
           CEDF(YES) LANGUAGE(LE370)
           REMOTESYSTEM() REMOTENAME() TRANSID()
    PROTECT API(CICSAPI) CONCURRENCY(QUASIRENT)
           DATALOCATION(ANY) DYNAMIC(NO)
           EXECCKEY(USER) EXECUTIONSET(FULLAPI)
           RELOAD(NO) RESIDENT(NO)
           STATUS(ENABLED) USAGE(NORMAL) USELPACOPY(NO)
    HIDDEN JVM(NO) JVMCLASS() JVMPROFILE(DFHJVMPR)
*
*   Atributos predeterminados de TDQueue
*
DEFINE TDQUEUE()
    UPDATE  DESCRIPTION()
           TYPE(INTRA)
* Parámetros extra-partición
    DDNAME() DSNAME()
    REMOTENAME() REMOTESYSTEM() REMOTELLENGTH(1)
    RECORDSIZE() BLOCKSIZE(0) RECORDFORMAT(UNDEFINED)
    BLOCKFORMAT() PRINTCONTROL() DISPOSITION(SHR)
* Parámetros intra-partición
    FACILITYID() TRANSID() TRIGERRLEVEL(1)
    USERID()
* Parámetros indirectos
    INDIRECTNAME()
    PROTECT WAIT(YES) WAITACTION(REJECT)
* Parámetros extra-partición
    DATABUFFERS(1)
    SYSOUTCLASS() ERROROPTION(IGNORE)
    OPENTIME(INITIAL) REWIND(LEAVE) TYPEFILE(INPUT)
* Parámetros intra-partición
    ATIFACILITY(TERMINAL) RECOVSTATUS(NO)

```

*Figura 59. ADNJSPAU - programa de utilidad administrativo de CICSTS (Parte 2 de 3)*

```

*
*   Atributos predeterminados de Transaction
*
DEFINE TRANSACTION()
    UPDATE  DESCRIPTION()
            PROGRAM()
            TWASIZE(0)
            REMOTESYSTEM() REMOTENAME() LOCALQ(NO)
    PROTECT PARTITIONSET() PROFILE(DFHCICST)
            DYNAMIC(NO) ROUTABLE(NO)
            ISOLATE(YES) STATUS(ENABLED)
            RUNAWAY(SYSTEM) STORAGECLEAR(NO)
            SHUTDOWN(DISABLED)
            TASKDATAKEY(USER) TASKDATALOC(ANY)
            BREXIT() PRIORITY(1) TRANCLASS(DFHTCL00)
            DTIMOUT(NO) RESTART(NO) SPURGE(NO) TPURGE(NO)
            DUMP(YES) TRACE(YES) CONFDATA(NO)
            OTSTIMEOUT(NO) WAIT(YES) WAITTIME(00,00,00)
            ACTION(BACKOUT) INDOUBT(BACKOUT)
            RESSEC(NO) CMDSEC(NO)
            TRPROF()
            ALIAS() TASKREQ()
            XTRANID() TPNAME() XTPNAME()

|
| *
| *   Atributos de URDIMAP
| *
| DEFINE URIMAP()
|     UPDATE  USAGE(CLIENT)
|             DESCRIPTION()
|             PATH(/required/path)
|             TCPIPSERVICE()
|             TRANSACTION()
|             PROGRAM()
|     PROTECT ANALYZER(NOANALYZER)
|             ATOMSERVICE()
|             CERTIFICATE()
|             CHARACTERSET()
|             CIPHERS()
|             CONVERTER()
|             HFSFILE()
|             HOST(host.mycompany.com)
|             HOSTCODEPAGE()
|             LOCATION()
|             MEDIATYPE()
|             PIPELINE()
|             PORT(NO)
|             REDIRECTTYPE(NONE)
|             SCHEME(HTTP)
|             STATUS(ENABLED)
|             TEMPLATENAME()
|             USERID()
|             WEBSERVICE()
|
| *
| *   Enlace opcionales de nombre de archivo con nombre de conjunto de datos VSAM
| *
| *DEFINE DSBINDING() DSNAME()
| /*

```

Figura 59. ADNJSPAU - programa de utilidad administrativo de CICSTS (Parte 3 de 3)

## Notas de migración del programa de utilidad administrativo

Developer for System z versión 7.6.1 añadió soporte de URIMAP al programa de utilidad administrativo. Para poder utilizar el soporte URIMAP, el conjunto de

datos de repositorio CRD VSAM debe estar asignado con un tamaño de registro máximo de 3000. Hasta Developer for System z Versión 7.6.1, el trabajo de asignación de repositorio CRD de ejemplo utiliza un tamaño de registro máximo de 2000.

Siga estos pasos para habilitar el soporte de URIMAP si está utilizando un repositorio CRD más antiguo:

1. Cree una copia de seguridad del repositorio CRD existente, FEK.#CUST.ADNREPF0.
2. Suprima el repositorio CRD existente.
3. Personalice y someta el trabajo FEK.SFEKSAMP(ADNVCRD) para asignar e inicializar un repositorio CRD nuevo. Consulte la documentación del miembro para obtener instrucciones de personalización.
4. Personalice y someta el trabajo FEK.SFEKSAMP(ADNJSPAU) para utilizar el programa de utilidad administrativo para llenar el repositorio CRD nuevo.

**Nota:**

- No es necesario migrar el repositorio CRD existente porque el programa de utilidad administrativo sustituye todo el contenido del repositorio CRD cada vez que se ejecuta.
- No hay problemas de compatibilidad de versión con el repositorio CRD. Todo el código de cliente y de host de Developer for System z soportado funcionará con el tamaño de registro máximo para cada caso. Pero el soporte de URIMAP estará inhabilitado si el tamaño de registro máximo no es 3000.

## Mensajes del programa de utilidad administrativo

El programa de utilidad administrativo emite los mensajes siguientes para la DD SYSPRINT. Los mensajes CRAZ1803E, CRAZ1891E, CRAZ1892E y CRAZ1893E contienen códigos de estado de archivo, retorno VSAM, función VSAM e información de retorno VSAM. Los códigos de retorno, función e información de retorno de VSAM se describen en la publicación *DFSMS Macro Instructions for Data Sets* (SC26-7408). Los códigos de estado de archivo se describen en la publicación *Enterprise COBOL for z/OS Language Reference* (SC27-1408).

### **CRAZ1800I**

**completado satisfactoriamente en la línea <número de línea de última sentencia de control>**

**Descripción:** El programa de utilidad administrativo del programador del sistema se ha completado satisfactoriamente.

**Respuesta del usuario:** Ninguna.

### **CRAZ1801W**

**completado con avisos en la línea <número de línea de última sentencia de control>**

**Descripción:** El programa de utilidad administrativo del programador del sistema se ha completado con uno o varios avisos encontrados al procesar las sentencias de control.

**Respuesta del usuario:** Compruebe si hay otros mensajes de aviso.

### **CRAZ1802E**

**se ha encontrado un error en la línea <número de línea>**

**Descripción:** El programa de utilidad administrativo del programador del sistema ha encontrado un error grave.

**Respuesta del usuario:** Compruebe si hay otros mensajes de aviso.

**CRAZ1803E**

**Error de apertura de repositorio, estado=<código de estado de archivo>  
RC=<código de retorno de VSAM> FC=<código de función de VSAM>  
FB=<código de información de VSAM>**

**Descripción:** El programa de utilidad administrativo del programador del sistema ha encontrado un error grave al abrir el repositorio de CRD.

**Respuesta del usuario:** Compruebe los códigos de estado, de retorno, de función y de información de retorno de VSAM.

**CRAZ1804E**

**Registro de entrada no reconocido en la línea <número de línea>**

**Descripción:** El programa de utilidad administrativo del programador del sistema ha encontrado una sentencia de control de entrada no reconocida.

**Respuesta del usuario:** Compruebe que un mandato **DEFINE** vaya seguido de un solo espacio y de la palabra clave **CPSMNAME**, **STAGINGGROUPNAME**, **MANIFESTEXPORTRULE**, **DSBINDING**, **DB2TRAN**, **DOCTEMPLATE**, **FILE**, **MAPSET**, **PROCESSTYPE**, **PROGRAM**, **TDQUEUE** o **TRANSACTION**.

**CRAZ1805E**

**Procesando palabra clave <palabra clave> en la línea <número de línea>**

**Descripción:** El programa de utilidad administrativo del programador del sistema está procesando la sentencia de control de entrada de la palabra clave **DEFINE**.

**Respuesta del usuario:** Ninguna.

**CRAZ1806E**

**Regla de exportación de manifiestos no válida en la línea <número de línea>**

**Descripción:** El programa de utilidad administrativo del programador del sistema ha encontrado una regla de exportación de manifiestos no válida.

**Respuesta del usuario:** Compruebe que el valor de la palabra clave **MANIFESTEXPORTRULE** sea "installOnly", "exportOnly" o "both".

**CRAZ1807E**

**Falta la palabra clave **DSNAME** en la línea <número de línea>**

**Descripción:** El programa de utilidad administrativo del programador del sistema estaba procesando una sentencia de control **DEFINE DSBINDING** a la que le falta la palabra clave **DSNAME**.

**Respuesta del usuario:** Compruebe que la sentencia de control **DEFINE DSBINDING** contenga la palabra clave **DSNAME**.

**CRAZ1808E**

**Valor de palabra clave no válido para la palabra clave <palabra clave> en la línea <número de línea>**

**Descripción:** El programa de utilidad administrativo del programador del sistema estaba procesando una sentencia de control **DEFINE** y ha encontrado un valor no válido para la palabra clave indicada.

**Respuesta del usuario:** Compruebe que la longitud y el valor de la palabra clave indicada sean correctos.

**CRAZ1890W**

**Error de sintaxis de palabra clave en la línea <número de línea>**

**Descripción:** El programa de utilidad administrativo del programador del sistema estaba procesando una sentencia de control DEFINE y ha encontrado un error de sintaxis en una palabra clave o en un valor de palabra clave.

**Respuesta del usuario:** Compruebe que el valor de palabra clave se haya especificado entre paréntesis y se encuentre inmediatamente a continuación de la palabra clave. La palabra clave y su valor deben encontrarse en la misma línea.

**CRAZ1891W**

**Error de grabación de clave duplicada en el repositorio, estado=<código de estado de archivo> RC=<código de retorno de VSAM> FC=<código de función de VSAM> FB=<código de información de retorno de VSAM>**

**Descripción:** El programa de utilidad administrativo del programador del sistema ha encontrado un error de clave duplicada al grabar en el repositorio de CRD.

**Respuesta del usuario:** Compruebe los códigos de estado, de retorno, de función y de información de retorno de VSAM.

**CRAZ1892W**

**Error de grabación en el repositorio, estado=<código de estado de archivo> RC=<código de retorno de VSAM> FC=<código de función de VSAM> FB=<código de información de VSAM>**

**Descripción:** El programa de utilidad administrativo del programador del sistema ha encontrado un error grave al grabar en el repositorio de CRD.

**Respuesta del usuario:** Compruebe los códigos de estado, de retorno, de función y de información de retorno de VSAM.

**CRAZ1893W**

**Error de lectura del repositorio, estado=<código de estado de archivo> RC=<código de retorno de VSAM> FC=<código de función de VSAM> FB=<código de información de VSAM>**

**Descripción:** El programa de utilidad administrativo del programador del sistema ha encontrado un error grave al leer el repositorio de CRD.

**Respuesta del usuario:** Compruebe los códigos de estado, de retorno, de función y de información de retorno de VSAM.

---

## Capítulo 16. Personalizar el entorno TSO

Este apéndice está destinado a servir de ayuda para emular un procedimiento de inicio de sesión TSO añadiendo sentencias DD y conjuntos de datos al entorno TSO en Developer for System z.

---

### El servicio de mandatos TSO

El servicio de mandatos TSO es el componente de Developer for System z que ejecuta mandatos TSO e ISPF (por lotes) y devuelve el resultado al cliente solicitante. Estos mandatos puede solicitarlos implícitamente el producto o explícitamente el usuario.

Los miembros de ejemplo suministrados con Developer for System z crean un entorno TSO/ISPF mínimo. Si los desarrolladores de su establecimiento necesitan acceso a bibliotecas personalizadas o de terceros, el programador del sistema z/OS debe añadir las sentencias DD y bibliotecas necesarias al entorno de servicio de mandatos TSO. Aunque la implementación es diferente en Developer for System z, la lógica subyacente es idéntica al procedimiento de inicio de sesión de TSO.

**Nota:** El servicio de mandatos TSO es una herramienta de línea de mandatos no interactiva, por lo que los mandatos o procedimientos que solicitan datos o visualizan paneles de ISPF no funcionarán. Para ejecutarlos, será necesario un emulador 3270, como el Emulador de conexión de host que forma parte del cliente Developer for System z.

### Métodos de acceso

Desde la versión 7.1, Developer for System z ofrece una opción relativa a cómo acceder al servicio de mandatos TSO.

- Pasarela de cliente TSO/ISPF de ISPF, que requiere un nivel de servicio mínimo de ISPF. Este es el método predeterminado utilizado en los ejemplos suministrados.
- Una transacción APPC (como en los releases anteriores a la versión 7.1).

**Nota:** El servicio de Pasarela de cliente TSO/ISPF de ISPF sustituye a la función de SCLM Developer Toolkit utilizada en la versión 7.1.

Compruebe `rsed.envvars` para determinar el método de acceso utilizado para hosts de la versión 7.1 y posteriores. Si se han utilizado los valores predeterminados durante el proceso de configuración, `rsed.envvars` reside en `/etc/rdz/`.

- Si la sentencia `_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DTSO_SERVER=APPC"` no está presente (o está comentada, que es el valor predeterminado), se utiliza el servicio de Pasarela de cliente TSO/ISPF de ISPF.
- Si la sentencia `_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DTSO_SERVER=APPC"` está presente (y no está comentada), se utiliza APPC.

---

## Utilizar el método de acceso de Pasarela de cliente TSO/ISPF

### Personalización básica – ISPF.conf

El archivo de configuración ISPF.conf (por omisión ubicado en /etc/rdz/) define el entorno TSO/ISPF utilizado por Developer for System z. Sólo existe un archivo de configuración ISPF.conf activo, que utilizan todos los usuarios de Developer for System z.

La sección principal del archivo de configuración define los nombres de DD y las concatenaciones de conjuntos de datos relacionados, como en el ejemplo siguiente:

```
sysproc=ISP.SISPCLIB,FEK.SFEKPROC
ispmlib=ISP.SISPMENU
isptlib=ISP.SISPTENU
ispplib=ISP.SISPPENU
ispslib=ISP.SISPSLIB
ispllib=ISP.SISpload
myDD=HLQ1.LLQ1,HLQ2.LLQ2
```

- Cada definición de DD utiliza exactamente una línea (la multilínea no está soportada) y no hay límites de longitud de línea.
- Las definiciones no son sensibles a las mayúsculas y minúsculas.
- Las líneas de comentarios empiezan por un asterisco (\*).
- Los nombres de DD van seguidos de un signo igual (=), que a su vez va seguido de la concatenación de conjuntos de datos. Los diversos nombres de conjuntos de datos están separados por una coma (,).
- Las búsquedas en las concatenaciones de conjuntos de datos se realizan en el orden de la lista.
- Los conjuntos de datos deben estar totalmente calificados, sin entrecomillarse (') y sin utilizar variables.
- Todos los conjuntos de datos se asignan con DISP=SHR.
- Pueden añadirse nombres de DD nuevos a voluntad, pero deben ajustarse a las normas (JCL) para los nombres de DD y no pueden entrar en conflicto con otros parámetros de configuración de ISPF.conf. Asimismo, ISPPROF se asigna dinámicamente (DISP=NEW,DELETE) por medio del servicio de Pasarela de cliente TSO/ISPF.

### Avanzado – Utilizar perfiles ISPF existentes

Por omisión, La Pasarela de cliente TSO/ISPF crea un perfil ISPF temporal para el servicio de mandatos TSO. Sin embargo, puede indicar a la Pasarela de cliente z TSO/ISPF que utilice una copia de un perfil ISPF existente. La clave es aquí la sentencia `_RSE_CMDSERV_OPTS` de `rsed.envvars`.

```
#_RSE_CMDSERV_OPTS="$_RSE_CMDSERV_OPTS&ISPPROF=&SYSUID..ISPPROF"
```

Descomente la sentencia (elimine el carácter de almohadilla (#) inicial) y especifique el nombre totalmente calificado del conjunto de datos del perfil ISPF existente para utilizar este recurso.

Pueden utilizarse las variables siguientes en el nombre del conjunto de datos:

- `&SYSUID`. para sustituir el ID de usuario del desarrollador
- `&SYSPREF`. para sustituir el prefijo TSO del desarrollador

**Nota:**

- Si el nombre del conjunto de datos pasado en "ISPPROF" no es válido, se utiliza en su lugar un perfil ISPF temporal vacío.
- El perfil ISPF (tanto el temporal como el copiado) se suprime al final de la sesión. Los cambios efectuados en el perfil no se fusionan con el perfil ISPF existente.

## Avanzado – Utilizar un exec de asignación

La sentencia `allocjob` de `ISPF.conf` (que está comentada por omisión) señala a un exec que puede utilizarse para suministrar más asignaciones de conjunto de datos por ID de usuario.

```
*allocjob = FEK.#CUST.CNTL(CRAISPRX)
```

Descomente la sentencia (elimine el carácter de asterisco (\*) inicial) y especifique la referencia totalmente calificada al exec de asignación para utilizar este recurso.

- El exec se ejecuta después de la asignación de `ISPPROF` y de los DD definidos en `ISPF.conf`, pero antes de inicializar ISPF. Asegúrese del que el exec de asignación no deshace estas definiciones.
- Se pasa 1 parámetro al exec: el ID de usuario del llamante.
- En la biblioteca de ejemplo `FEK.#CUST.CNTL` se proporciona un exec de ejemplo `CRAISPRX`, a menos que haya especificado otra ubicación al personalizar y someter el trabajo `FEK.SFEKSAMP(FEKSETUP)`. Encontrará más detalles en: "Configuración de la personalización" en la página 15.

**Nota:** Dado que el exec se llama antes de inicializar ISPF, no puede utilizarse `VPUT` ni `VGET`. Sin embargo, puede crear su propia implementación de estas funciones utilizando un archivo `PDS(E)` o `VSAM`.

## Avanzado – Utilizar varios execs de asignación

Aunque `ISPF.conf` sólo da soporte a la llamada a un exec de asignación, no hay límite para que ese exec llame a otro exec. Y el ID de usuario del cliente que se pasa como parámetro abre la posibilidad de llamar a execs de asignación personalizados. Por ejemplo, puede comprobar si el miembro `USERID'.EXEC(ALLOC)'` existe y ejecutarlo.

Una variante elaborada de este tema permite utilizar los procedimientos de inicio de sesión TSO existentes, del siguiente modo:

- Lea un archivo de configuración específico del usuario, como `USERID'.FEKPROF'`.
- Observe qué procedimiento de inicio de sesión se menciona en el archivo.
- Lea el procedimiento mencionado en `SYS1.PROCLIB` y analícelo para encontrar las sentencias DD y las asignaciones de conjunto de datos que contiene.
- Asigne el conjunto de datos de modo similar al procedimiento de inicio de sesión real.

## Avanzado – Varios archivos ISPF.conf con varias configuraciones de Developer for System z

Si los escenarios de exec de asignación descritos anteriormente no pueden manejar sus necesidades específicas, puede crear instancias diferentes del servidor de comunicaciones RSE de Developer for System z, cada una de ellas con su propio archivo `ISPF.conf`. El principal inconveniente del método descrito más abajo es que los usuarios de Developer for System z deben conectarse a servidores diferentes del mismo host para obtener el entorno TSO deseado.

**Nota:** Para crear una segunda instancia del servidor RSE sólo es necesario duplicar y actualizar los archivos de configuración, el JCL de inicio y las definiciones de tareas iniciadas. No es necesario realizar una nueva instalación del producto, ni tampoco duplicar ningún código.

```
$ cd /etc/rdz
$ mkdir /etc/rdz/tso2
$ cp rsed.envvars /etc/rdz/tso2
$ cp ISPF.conf /etc/rdz/tso2
$ ls /etc/rdz/tso2
ISPF.conf          rsed.envvars
$ oedit /etc/rdz/tso2/rsed.envvars
-> cambiar: _CMDSERV_CONF_HOME=/etc/rdz/tso2
-> descomentar y cambiar: -Ddaemon.log=/var/rdz/logs/tso2
-> añadir al final (END):
# -- NECESARIO PARA ENCONTRAR LOS ARCHIVOS DE CONFIGURACIÓN RESTANTES
CFG_BASE=/etc/rdz
CLASSPATH=.:$CFG_BASE:$CLASSPATH
# --
$ oedit /etc/rdz/tso2/ISPF.conf
-> cambiar: cambiar según sea necesario
```

Los mandatos del ejemplo anterior copian los archivos de configuración de Developer for System z que necesitan cambios en un directorio tso2 de reciente creación. La variable `_CMDSERV_CONF_HOME` de `rsed.envvars` debe actualizarse para definir el nuevo directorio inicial de `ISPF.conf` y `daemon.log` debe actualizarse para definir una ubicación de anotaciones nueva (que se crea automáticamente en caso de que no exista). La actualización de `CLASSPATH` garantiza que RSE pueda encontrar los archivos de configuración que no se han copiado en tso2. El propio archivo `ISPF.conf` puede actualizarse para ajustarlo a las necesidades. Tenga en cuenta que el área de trabajo de ISPF (variable `_CMDSERV_WORK_HOME` de `rsed.envvars`) puede compartirse entre ambas instancias.

La tarea restante consiste en crear una tarea iniciada para RSE que utilice un número de puerto nuevo y los nuevos archivos de configuración `/etc/rdz/tso2`.

Consulte las secciones relacionadas de esta publicación para obtener más información acerca de las acciones mostradas anteriormente.

---

## Utilizar el método de acceso APPC

### Personalización básica – JCL de transacción APPC

La definición de una transacción APPC consta de parámetros APPC y un JCL de transacción. El JCL de ejemplo destinado a crear una transacción APPC de Developer for System z, `FEK.#CUST.JCL(FEKAPPCC)`, contiene dos opciones para definir el JCL de transacción, con y sin soporte para ISPF.

```
//SYSIN DD DDNAME=SYSINISP * utilizar SYSINTSO o SYSINISP
```

El cliente obtiene el entorno TSO/ISPF definido en el JCL de transacción, por lo que personalizando esta sección y siguiendo las normas habituales de DD puede personalizar el entorno para el cliente.

```
...
//CMDSERV EXEC PGM=IKJEFT01,DYNAMNBR=50,
// PARM='ISPSTART CMD(%FEKFRSRV TIMEOUT=60) NEWAPPL(ISR) NESTMACS'
//SYSPROC DD DISP=SHR,DSN=FEK.SFEKPROC
//ISPLIB DD DISP=SHR,DSN=ISP.SISPPENU
//ISPLIB DD DISP=SHR,DSN=ISP.SISPMENU
//ISPTLIB DD DISP=SHR,DSN=ISP.SISPTENU
//ISPSLIB DD DISP=SHR,DSN=ISP.SISPSENU
```

```
//ISPPROF DD DISP=(NEW,DELETE,DELETE),UNIT=SYSALLDA,
//        SPACE=(TRK,(1,1,5)),LRECL=80,RECFM=FB
//SYSTSPRT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSTSIN DD DUMMY
//MYDD DD DISP=SHR,DSN=HLQ1.LLQ1
//        DISP=SHR,DSN=HLQ2.LLQ2
```

**Nota:** Una transacción APPC existente puede modificarse mediante los paneles ISPF de APPC.

## Avanzado – Utilizar perfiles ISPF existentes

Si se selecciona el soporte ISPF, Developer for System z crea por omisión un perfil ISPF temporal para el servicio de mandatos TSO. Sin embargo, puede indicar a Developer for System z que utilice una copia de un perfil ISPF existente. Como se describe en el trabajo de ejemplo FEK.SFEKSAMP(FEKAPPC), debe hacer lo siguiente:

- Descomentar el paso COPY del JCL de transacción (EXEC y las tarjetas DD relacionadas).
- Cambiar &SYSUID..ISPPROF para que coincida con el nombre del conjunto de datos del perfil ISPF del usuario.
- Comentar la primera sentencia DD ISPPROF del paso CMDSERV y descomentar la segunda.

```
...
//COPY EXEC PGM=IEBCOPY * (opcional) clonar el perfil ISPF existente
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DISP=SHR,DSN=&SYSUID..ISPPROF
//SYSUT2 DD DISP=(MOD,PASS),DSN=&&PROF,
//        UNIT=SYSALLDA,
//        LIKE=&SYSUID..ISPPROF
//*
//CMDSERV EXEC PGM=IKJEFT01,DYNAMNBR=50,
// PARM='ISPSTART CMD(%FEKFRSRV TIMEOUT=60) NEWAPPL(ISR) NESTMACS'
//*ISPPROF DD DISP=(NEW,DELETE,DELETE),UNIT=SYSALLDA,
//*        SPACE=(TRK,(1,1,5)),LRECL=80,RECFM=FB
//ISPPROF DD DISP=(OLD,DELETE,DELETE),DSN=&&PROF
```

**Nota:** Si se utiliza un nombre de conjunto de datos no válido, el inicio de la transacción APPC (y por tanto del servicio de mandatos TSO) fallará.

## Avanzado – Utilizar un exec de asignación

El JCL de transacción de ejemplo llama directamente al servicio de mandatos TSO pasando su nombre (FEKFRSRV) como parámetro al programa IKJEFT01. Puede cambiar este procedimiento para que llame a otro exec. Este exec puede realizar asignaciones basadas en el ID de usuario actual y luego llamar al servicio de mandatos TSO.

Al contrario que en el método de acceso de la Pasarela de cliente TSO/ISPF, este exec puede utilizar las variables almacenadas en el perfil ISPF del usuario para facilitar la personalización del entorno. Sin embargo, recuerde que las actualizaciones del perfil se perderán al final de la sesión, ya que está utilizando una copia temporal, no el perfil real.

Sin embargo, tenga en cuenta que la utilización de un exec de asignación en la transacción APPC no está soportada y la descripción anterior es tal cual.

## Avanzado – Varias transacciones APPC con varias configuraciones de Developer for System z

Si necesita varios entornos TSO exclusivos, puede crear instancias diferentes del servidor de comunicaciones RSE de Developer for System z, cada una de ellas con su propia transacción APPC. El principal inconveniente del método descrito más abajo es que los usuarios de Developer for System z deben conectarse a servidores diferentes del mismo host para obtener el entorno TSO deseado.

**Nota:** Para crear una segunda instancia del servidor RSE sólo es necesario duplicar y actualizar los archivos de configuración, el JCL de inicio y las definiciones de tareas iniciadas. No es necesario realizar una nueva instalación del producto, ni tampoco duplicar ningún código.

```
$ cd /etc/rdz
$ mkdir /etc/rdz/tso2
$ cp rsed.envvars /etc/rdz/tso2
$ ls /etc/rdz/tso2/
rsed.envvars
$ oedit /etc/rdz/tso2/rsed.envvars
-> descomentar y cambiar: _FEKFSCMD_TP_NAME_=FEKFTSO2
-> descomentar y cambiar: -Ddaemon.log=/var/rdz/logs/tso2
-> añadir al final (END):
# -- NECESARIO PARA ENCONTRAR LOS ARCHIVOS DE CONFIGURACIÓN RESTANTES
CFG_BASE=/etc/rdz
CLASSPATH=.:$CFG_BASE:$CLASSPATH
# --
```

Los mandatos anteriores crean un directorio tso2 y copian los archivos de configuración de Developer for System z que necesitan cambios en la nueva ubicación. La variable `_FEKFSCMD_TP_NAME_` de `rsed.envvars` debe actualizarse para definir el nombre de la transacción APPC nuevo y `daemon.log` deben actualizarse para definir una ubicación de anotaciones de daemon nueva (que se crea automáticamente en caso de que no exista). La actualización de `CLASSPATH` garantiza que RSE pueda encontrar los archivos de configuración que no se han copiado en tso2.

```

//FEKAPPCC JOB CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1),NOTIFY=&SYSUID
//*
//TPADD EXEC PGM=ATBSDFMU
//SYSSDLIB DD DISP=SHR,DSN=SYS1.APPCTP
//SYSSDOUT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSIN DD DDNAME=SYSINISP * utilizar SYSINTSO o SYSINISP
//SYSINISP DD DATA,DLM='QT'
TPADD
TPNAME(FEKFTS02)
ACTIVE(YES)
TPSCHED_DELIMITER(DLM1)
KEEP_MESSAGE_LOG(ERROR)
MESSAGE_DATA_SET(&SYSUID..FEKFTS02.&TPDATE..&TPTIME..LOG)
DATASET_STATUS(MOD)
CLASS(A)
JCL_DELIMITER(DLM2)
//FEKFTS02 JOB CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1)
//*
//CMDSERV EXEC PGM=IKJEFT01,DYNAMNBR=50,
// PARM='ISPSTART CMD(%FEKFRSRV TIMEOUT=60) NEWAPPL(ISR) NESTMACS'
//SYSPROC DD DISP=SHR,DSN=FEK.SFEKPROC
//ISPPLIB DD DISP=SHR,DSN=ISP.SISPPENU
//ISPMLIB DD DISP=SHR,DSN=ISP.SISPMENU
//ISPTLIB DD DISP=SHR,DSN=ISP.SISPTENU
//ISPSLIB DD DISP=SHR,DSN=ISP.SISPSENU
//ISPPROF DD DISP=(NEW,DELETE,DELETE),UNIT=SYSALLDA,
// SPACE=(TRK,(1,1,5)),LRECL=80,RECFM=FB
//SYSTSPRT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSTSIN DD DUMMY
//MYDD DD DISP=SHR,DSN=HLQ1.LLQ1
// DISP=SHR,DSN=HLQ2.LLQ2
DLM2
DLM1
QT

```

Figura 60. FEKAPPCC - crear una segunda transacción APPC

A continuación, cree una transacción APPC personalizando y sometiendo el trabajo de ejemplo FEK.#CUST.JCL(FEKAPPCC), tal como se muestra en el ejemplo anterior. Encima de la personalización normal (descrita en el JCL), debe cambiar también TPNAME por TPNAME(FEKFTS02) para que coincida con la definición de \_FEKFSCMD\_TP\_NAME\_ del nuevo archivo rsed.envvars. Debe cambiar también el nombre en la variable MESSAGE\_DATA\_SET y el nombre del trabajo (JOB) del JCL de transacción.

La tarea restante consiste en crear una tarea iniciada para RSE que utilice un número de puerto nuevo y los nuevos archivos de configuración /etc/rdz/tso2.

Consulte las secciones relacionadas de esta publicación para obtener más información acerca de las acciones mostradas anteriormente.



---

## Capítulo 17. Ejecutar varias instancias

En algunas ocasiones le interesará tener múltiples instancias de Developer for 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 for System z, y después podrá usar esta guía de configuración para personalizarlas.

Aunque es posible compartir algunos componentes de Developer for System z entre dos (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 for System z deja suficiente margen de personalización para crear múltiples instancias que no se solapen, y le aconsejamos encarecidamente que utilice estas características.

### Nota:

- FEK y /usr/lpp/rdz son el calificador de alto nivel y la vía de acceso que se emplean durante la instalación del producto. FEK.#CUST, /etc/rdz y /var/rdz son las ubicaciones predeterminadas utilizadas durante la personalización del producto (consulte la sección “Configuración de la personalización” en la página 15 para obtener más información).
- Debe instalar Developer for System z en un sistema de archivos privado (HFS o zFS) para desplegar fácilmente los componentes z/OS UNIX del producto.
- Si no puede utilizar un sistema de archivos privado, debe utilizar una herramienta de archivado como el mandato tar de z/OS UNIX para transportar los directorios de z/OS UNIX de un sistema a otro. Eso permite conservar los atributos (como por ejemplo el control de programa) de los archivos y directorios de Developer for System z. Consulte la publicación *UNIX System Services Command Reference* (SA22-7802) para obtener más información acerca de los siguientes mandatos de ejemplo para archivar y restaurar el directorio de instalación de Developer for System z.
  - Archivar: `cd /SYS1/usr/lpp/rdz; tar -cSf /u/userid/rdz.tar`
  - Restaurar: `cd /SYS2/usr/lpp/rdz; tar -xSf /u/userid/rdz.tar`

---

## Configuración idéntica en todo un sysplex

Los archivos de configuración de Developer for System z (y el código) se pueden compartir entre los distintos sistemas de un sysplex (cada sistema ejecuta su propia copia idéntica de Developer for System z), si se siguen algunas directrices.

- Los archivos de anotaciones deben terminar en ubicaciones exclusivas para evitar que un sistema sobrescriba la información de otro. Direccionando las anotaciones de z/OS UNIX a ubicaciones concretas con las directivas `daemon.log` y `user.log` de `rsed.envvars`, puede compartir los archivos de configuración si monta un sistema de archivos de z/OS UNIX específico del sistema en la vía de acceso especificada. De esta manera, todas las anotaciones se escriben en el mismo sitio lógico, pero terminan en ubicaciones físicas distintas.

- Los directorios del tipo configuración, como /etc/rdz/ y /var/rdz/projects/, se pueden compartir en sysplex, ya que Developer for System z los utiliza en modalidad de sólo lectura.
- Los directorios de datos temporales, como /tmp/ y /var/rdz/WORKAREA/, deberían ser exclusivos para cada sistema, ya que los nombres de archivos temporales no tienen en cuenta sysplex.
- Si comparte el código, debe compartir también los archivos de configuración para asegurar que no tienen algunos sistemas fuera de la sincronización una vez aplicado el mantenimiento.

## 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 utilizar de manera segura para probar el mantenimiento, un avance tecnológico o un nuevo release.

Para configurar otra instancia de una instalación activa de Developer for System z, rehaga los pasos de personalización para los componentes que sean distintos, utilizando diferentes conjuntos de datos, directorios y puertos para evitar que se solape con la instalación actual.

En el ejemplo SSL mencionado más arriba, la configuración del daemon RSE actual se puede clonar y, después, la configuración clonada se puede actualizar. A continuación, el JCL de inicio del daemon RSE puede clonarse y personalizarse con un nuevo puerto TCP/IP y la ubicación de los archivos de configuración actualizados. 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:

```
$ cd /etc/rdz
$ mkdir /etc/rdz/ssl
$ cp rsed.envvars /etc/rdz/ssl
$ cp ssl.properties /etc/rdz/ssl
$ ls /etc/rdz/ssl/
rsed.envvars    ssl.properties
$ oedit /etc/rdz/ssl/rsed.envvars
-> descomentar y cambiar: -Ddaemon.log=/var/rdz/logs/ssl
-> añadir al final (END):
# -- NECESARIO PARA ENCONTRAR LOS ARCHIVOS DE CONFIGURACIÓN RESTANTES
CFG_BASE=/etc/rdz
CLASSPATH=.:$CFG_BASE:$CLASSPATH
# --
$ oedit /etc/rdz/ssl/ssl.properties
-> cambiar: cambiar según sea necesario
```

Los mandatos anteriores copian los archivos de configuración de Developer for System z que necesitan cambios en un directorio ssl de reciente creación. La variable daemon.log de rsed.envvars debe actualizarse para definir una ubicación de anotaciones nueva (que se crea automáticamente en caso de que no exista). La actualización de CLASSPATH garantiza que RSE pueda encontrar los archivos de configuración que no se han copiado en ssl. El propio archivo ssl.properties puede actualizarse para ajustarlo a las necesidades.

La tarea restante consiste en crear una tarea iniciada para RSE que utilice un número de puerto nuevo y los nuevos archivos de configuración `/etc/rdz/ssl`.

Consulte las secciones relacionadas de esta publicación para obtener más información acerca de las acciones mostradas anteriormente.

---

## Todas las demás situaciones

Cuando hay cambios de código implicados (de mantenimiento, avances tecnológicos, nuevo release), o cuando los cambios son bastante complejos, debe hacer otra instalación de Developer for 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 for System z:

- SMP/E CSI
- Bibliotecas de instalación
- Puerto TCP/IP del supervisor de trabajos JSE, así como su archivo de configuración FEJJCNFG
- JCL de inicio del supervisor de trabajos JES
- Nombre de la transacción APPC
- Archivos de configuración de RSE, `rsed.envvars`, `*.properties` y `.valores *`
- Puerto TCP/IP de RSE
- JCL de arranque de RSE

A continuación se proporciona una visión general más detallada:

- SMP/E CSI
  1. Instale cada instancia de Developer for System z en un CSI separado. SMP/E impedirá una segunda instalación del mismo FMID en un CSI, pero aceptará la instalación de otro FMID. Si el segundo FMID es una versión más reciente, suprimirá la versión existente del producto. Si el segundo FMID es una versión más antigua, la instalación fallará debido a los nombres de componentes duplicados.
- Bibliotecas de instalación
  1. Instale cada instancia de Developer for 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/rdz`. Un ejemplo válido podría ser `/service/usr/lpp/rdz`.
  2. El trabajo de configuración de personalización FEK.SFEKSAMP(FEKSETUP) crea los conjuntos de datos y directorios utilizados para almacenar los archivos de configuración. Dado que los archivos de configuración deben ser exclusivos, y para evitar la sobrescritura de las personalizaciones existentes, debe utilizar nombres exclusivos de conjunto de datos y directorio al someter este trabajo.
- Componentes obligatorios
  1. El archivo de configuración del supervisor de trabajos JES, `FEK.#CUST.PARMLIB(FEJJCNFG)`, 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, FEK.#CUST.PROCLIB(JMON), hace referencia a FEJCNFG y, por lo tanto, tampoco se puede compartir. Tras redenominar el miembro (y la tarjeta JOB si la ha iniciado como trabajo de usuario), puede colocar todos los JCL en el mismo conjunto de datos.
  3. El archivo de configuración de RSE, /etc/rdz/rsed.envvars, contiene referencias a la vía de instalación y, opcionalmente, a la ubicación de las anotaciones del servidor, que tiene que ser exclusivo. El nombre del archivo es obligatorio, por lo tanto, no podrá conservar distintas copias en el mismo directorio.
  4. El archivo de configuración ISPF.conf contiene una referencia a FEK.SFEKPROC(FEKFRRSV) , el servidor de mandatos TSO. Es específico del nivel de software, por lo que debe crear un archivo ISPF.conf por cada instancia.
  5. Todos los demás archivos de configuración basados en z/OS UNIX (como \*.properties) deben residir en el mismo directorio que rsed.envvars y, por tanto no se pueden compartir, ya que rsed.envvars debe estar en una ubicación no compartida.
  6. El JCL de inicio de RSE FEK.#CUST.PROCLIB(RSED) no puede compartirse, ya que define el número de puerto TCP/IP y contiene una referencia a los directorios de instalación y configuración, que deben ser exclusivos. Tras redenominar el miembro (y la tarjeta JOB si la ha iniciado como trabajo de usuario), puede colocar todos los JCL en el mismo conjunto de datos.
  7. El JCL de inicio del daemon de bloqueo FEK.#CUST.PROCLIB(LOCKD) no puede compartirse, ya que contiene una referencia a los directorios de instalación y configuración, que deben ser exclusivos. Tras redenominar el miembro (y la tarjeta JOB si la ha iniciado como trabajo de usuario), puede colocar todos los JCL en el mismo conjunto de datos.
- Componentes opcionales
    1. Los puertos TCP/IP de REXEC y SSH se pueden compartir sin restricciones.
    2. La transacción APPC tiene una referencia a FEK.SFEKPROC(FEKFRRSV), el servidor de mandatos TSO. Este es específico del nivel de software, por lo que debe crear una transacción APPC por cada instancia. Tenga presente que, dado que el nombre de la transacción APPC cambia, hay que definir la variable \_FEKFSCMD\_TP\_NAME\_ en rsed.envvars.
    3. Algunos procedimientos ELAXF\* tienen una referencia a hlq.SFEKLOAD, la biblioteca de carga de Developer for System z. Para encontrar una posible solución de cómo hacer que distintos conjuntos estén disponibles para los usuarios, vea la nota sobre JCLLIB en: "Procedimientos de construcción remota ELAXF\*" en la página 25.
    4. 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               EXTERNAL NAME  ELAXMRXX
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) Procedimiento almacenado DB2” en la página 89.
  - e. El nombre nuevo del entorno WLM (por ejemplo, ELAXMWDZ) se debe utilizar en el asistente de procedimientos almacenados DB2 en el cliente.
5. El soporte bidireccional en las regiones CICS 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.
  6. Los módulos de carga del Gestor de despliegue de aplicaciones incluidos en las regiones CICS son compatibles hacia atrás y, por lo tanto, la versión más reciente se puede compartir en los distintos releases.
  7. El VSAM del CRD del Gestor de despliegue de aplicación es compatible hacia atrás y, por lo tanto, la versión más reciente se puede compartir en los distintos releases.
  8. Las definiciones de recursos CICS del Gestor de despliegue de aplicación son compatibles hacia atrás y, por lo tanto, la versión más reciente se puede compartir en los distintos releases.
  9. Los VSAM de CARMA podrían cambiar entre niveles de software, por lo que no conviene compartirlos.



---

## Capítulo 18. Guía de migración

---

### Consideraciones acerca de la migración

En este apartado se resaltan los cambios de instalación y configuración comparados con los releases anteriores del producto. También se ofrecen algunas directrices generales para la migración a este release. Para obtener más información, consulte las secciones relacionadas de este manual.

- Si es usted un usuario anterior de IBM Rational Developer for System z, IBM WebSphere Developer for System z, IBM WebSphere Developer for zSeries o IBM WebSphere Studio Enterprise Developer, le recomendamos que guarde los archivos personalizados relacionados ANTES de instalar la actualización en esta versión de IBM Rational Developer for System z Versión 7.6.1.
- Si piensa ejecutar múltiples instancias de Developer for System z, lea: Capítulo 17, “Ejecutar varias instancias”, en la página 273.

**Nota:** La información de migración de esta lista es para las versiones de Developer for System z que siguen estando soportadas en el momento de publicar este documento.

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

Si es usted un usuario anterior de Developer for System z, le recomendamos que guarde los archivos personalizados relacionados antes de instalar esta versión de IBM Developer for System z.

Puede encontrar archivos personalizables de Developer for system z en las ubicaciones siguientes:

- Versión 7.5
  - FEK.SFEKSAMP, el trabajo de ejemplo FEKSETUP copia algunos miembros en FEK.#CUST.\*, donde \* iguala PARMLIB, PROCLIB, JCL, CNTL, ASM y COBOL
  - FEK.SFEKSAMV
  - /usr/lpp/rdz/samples/, el trabajo de ejemplo FEKSETUP copia algunos archivos en /etc/rdz/ y /etc/rdz/sclmdt/\*, donde \* iguala CONFIG/, CONFIG/PROJECT/ y CONFIG/script/
- Versión 7.1
  - FEK.SFEKSAMP
  - CRA.SCRASAMP
  - /usr/lpp/wd4z/rse/lib/, se recomienda copiar los archivos personalizables en /etc/wd4z/
- Versión 7.0
  - FEK.SFEKSAMP
  - CRA.SCRASAMP
  - /usr/lpp/wd4z/rse/lib/, se recomienda copiar los archivos personalizables en /etc/wd4z/

Las configuraciones anteriores de Developer for system z también describen los cambios en los archivos de configuración propiedad de otros productos.

- Versión 7.5
  - SYS1.PARMLIB(ASCHPMxx)  
definir una clase de transacción APPC para el servicio de mandatos TSO
  - SYS1.PARMLIB(BPXPRMxx)  
establecer los valores predeterminados del sistema z/OS UNIX
  - SYS1.PARMLIB(COMMNDxx)  
iniciar los servidores en tiempo de IPL
  - SYS1.PARMLIB(LPALSTxx)  
añadir FEK.SFEKLPA a LPA
  - SYS1.PARMLIB(PROGxx)  
FEK.SFEKAUTH autorizada por APF  
añadir FEK.SFEKAUTH y FEK.SFEKLOAD a LINKLIST
  - (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 un procedimiento almacenado de DB2
- Versión 7.1
  - SYS1.PARMLIB(ASCHPMxx)  
definir una clase de transacción APPC para el servicio de mandatos TSO
  - SYS1.PARMLIB(BPXPRMxx)  
establecer los valores predeterminados del sistema z/OS UNIX
  - SYS1.PARMLIB(PROGxx)  
FEK.SFEKLOAD autorizada por APF
  - /etc/services  
definir el puerto del daemon RSE
  - /etc/inetd.conf  
definir el servicio del daemon RSE
  - /etc/SCLMDT/CONFIG/ISPF.conf  
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 un procedimiento almacenado de DB2
- Versión 7.0
  - SYS1.PARMLIB(ASCHPMxx)  
definir una clase de transacción APPC para el servicio de mandatos TSO
  - SYS1.PARMLIB(BPXPRMxx)  
establecer los valores predeterminados del sistema z/OS UNIX
  - SYS1.PARMLIB(PROGxx)

- FEK.SFEKLOAD autorizada por APF
- /etc/services  
definir el puerto del daemon RSE
- /etc/inetd.conf  
definir el servicio del daemon RSE
- (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 un procedimiento almacenado de DB2

---

## Notas de migración de la versión 7.6.1

Las notas de migración siguientes son específicas de la versión 7.6.1. Son válidas para migrar de la versión 7.6 o son añadidos a las notas de migración de la versión 7.6 existentes.

- Gestor de despliegue de aplicaciones: es necesario actualizar los módulos de ADN\* existentes en la concatenación RPL de CICS.
- Gestor de despliegue de aplicaciones: los miembros de ejemplo siguientes se han actualizado para añadir soporte URIMAP en el programa de utilidad administrativo:
  - ADNJSPAU
  - ADNVCRD
- Gestor de despliegue de aplicaciones: un VSAM de repositorio CRD existente debe sustituirse para habilitar el soporte URIMAP.
- CARMA: se ha añadido soporte para un diseño de longitud variable para el conjunto de datos VSAM de información personalizada de CARMA, CRASTRS
- CARMA: se han añadido miembros de ejemplo nuevos
  - CRA#VS2: migrar CRASTRS a formato de longitud variable
- Supervisor de trabajos JES: utilización de \_CEE\_ENVFILE\_S en el JCL de tarea iniciada.
- Supervisor de trabajos JES: las directivas FEJJCNGF siguientes se han convertido en opcionales
  - HOST\_CODEPAGE
- PROCLIB - Se han añadido miembros PROCLIB nuevos
  - ELAXFDCL
- RSE: la utilización de Java de 64 bits está ahora soportada.
- RSE: se han añadido mandatos de operador nuevos (desde la versión 7.6.1.0):
  - MODIFY DISPLAY PROCESS,DETAIL
- RSE: las directivas no personalizables han cambiado o son nuevas en rsed.envvars (desde la versión 7.6.0.0):
  - (\_RSE\_JAVAOPTS) -DDSTORE\_KEEPALIVE\_RESPONSE\_TIMEOUT
  - (\_RSE\_JAVAOPTS) -DDSTORE\_IO\_SOCKET\_READ\_TIMEOUT
  - (\_RSE\_JAVAOPTS) -DRSECOMM\_LOGFILE\_MAX
- RSE: Se han añadido directivas opcionales nuevas a rsed.envvars (desde la versión 7.6.0.0 y 7.6.0.1):

- (\_RSE\_JAVAOPTS) -Denable.automount
- (\_RSE\_JAVAOPTS) -Ddeny.nozero.port
- (\_RSE\_JAVAOPTS) -Dsingle.logon
- (\_RSE\_JAVAOPTS) -Dprocess.cleanup.interval
- RSE: los mensajes de consola siguientes se han cambiado o son nuevos (desde la versión 7.6.0.1 y 7.6.1.0):
  - FEK001I
  - FEK210I

---

## Migrar desde la versión 7.5 a la versión 7.6

### IBM Rational Developer for System z, FMID HHOP760

- La ubicación de instalación de SMP/E predeterminada para los componentes de MVS y z/OS UNIXno ha cambiado y, por ello, sigue siendo FEK.\* y /usr/lpp/rdz/\*.
- Gestor de despliegue de aplicaciones: es necesario actualizar los módulos de ADN\* existentes en la concatenación RPL de CICS.
- Gestor de despliegue de aplicaciones - Los módulos de carga nuevos, que deben ser parte de la concatenación de CICS TS, se han añadido para soportar la interfaz de RESTful de CICS.
  - ADNANAL
  - ADNCRD41
  - ADNREST
- Gestor de despliegue de aplicaciones - Se han añadido miembros de ejemplo nuevos para soportar la interfaz RESTful de CICS.
  - ADNCSDRS
  - ADNCSDTX
  - ADNTXNC
- Gestor de despliegue de aplicaciones - Se han renombrado los miembros de ejemplo existentes.
  - ADNARCSD -> ADNCSDAR
  - ADNCMSGH -> ADNMSGHC
  - ADNMFEST -> ADNVMFST
  - ADNPCCSD -> ADNCSDWS
  - ADNSMSGH -> ADNMSGHS
  - ADNVSAM -> ADNVCRD
- Se proporciona un RAM nuevo, tipo de producción, para acceder al CA Endevor® SCM.
  - CRARNDVR
- CARMA - Se han facilitado miembros de ejemplo nuevos para soportar el CA Endevor® SCM RAM.
  - FEK.#CUST.JCL(CRA#VCAD)
  - FEK.#CUST.JCL(CRA#VCAS)
  - FEK.#CUST.CNTL(CRASUBCA)
  - FEK.#CUST.PARMLIB(CRASHOW)
  - FEK.#CUST.PARMLIB(CRATMAP)
  - FEK.SFEKPROC(CRANDVRA)

- /etc/rdz/crstart.endevor.conf
- CARMA - Se han facilitado miembros de ejemplo nuevos para soportar la fusión de definiciones del RAM.
  - CRA#UADD
  - CRA#UQRY
- Integración del gestor de archivos - La interfaz por lotes para acceder al Gestor de archivos ya no está soportada.
- Integración del gestor de archivos - El archivo de configuración FMEXT.properties ha cambiado completamente y es necesario sustituirlo.
- Supervisor de trabajos JES - Las opciones de LE están incluidas en el módulo de carga FEJMON (desde la versión 7.5.0.1), que es posible que requiera cambios en su definición de tarea iniciada. Consulte el JLC de ejemplo FEK.SFEKSAMP(FEJJJCL) para obtener más detalles.
- Supervisor de trabajos JES - Se han añadido directivas opcionales nuevas a FEJCNFG (en las versiones 7.5.0.1 y 7.5.1.0).
  - APPLID
  - CONSOLE\_NAME
  - GEN\_CONSOLE\_NAME
- Supervisor de trabajos JES - Hay un mandato nuevo, Mostrar JCL, soportado (desde la versión 7.5.1.0), que puede requerir actualizaciones en su software de seguridad.
- Daemon de bloqueo – El daemon de bloqueo (LOCKD) es una tarea iniciada nueva (desde la versión 7.5.0.1). Se puede consultar esta tarea iniciada para identificar qué cliente de Developer for z está manteniendo el bloqueo de un conjunto de datos. (Los mandatos del sistema se detienen a nivel de espacio de dirección, que es la agrupación de hebras RSE.)
- SCLMDT - la ubicación predeterminada del archivo de configuración del proyecto SCLMDT ha cambiado.
  - /var/rdz/sclmdt
- RSE - Se han añadido mandatos de operador nuevos.
  - MODIFY RSESTANDARDLOG
- RSE - Se han añadido directivas obligatorias nuevas a rsed.envvars (en las versiones 7.5.0.1 y 7.6.0.0).
  - \_RSE\_LOCKD\_PORT
  - (\_RSE\_JAVAOPTS) -Dlock.daemon.port
  - (\_RSE\_JAVAOPTS) -Dlock.daemon.cleanup.interval
  - \_RSE\_LOCKD\_CLASS
  - \_RSE\_HOST\_CODEPAGE
  - (\_RSE\_JAVAOPTS) -Dfile.encoding
  - (\_RSE\_JAVAOPTS) -Dconsole.encoding
- RSE - Se han añadido directivas opcionales nuevas a rsed.envvars (desde las versiones 7.5.0.1, 7.5.1.0 y 7.6.0.0).
  - (\_RSE\_JAVAOPTS) -Duser.log
  - (\_RSE\_JAVAOPTS) -Dkeep.last.log
  - (\_RSE\_JAVAOPTS) -Denable.standard.log
  - (\_RSE\_JAVAOPTS) -DDSTORE\_LOG\_DIRECTORY
  - (\_RSE\_JAVAOPTS) -DHIDE\_ZOS\_UNIX
  - (\_RSE\_JAVAOPTS) -Denable.certificate.mapping

- GSK\_CRL\_SECURITY\_LEVEL
- GSK\_LDAP\_SERVER
- GSK\_LDAP\_PORT
- GSK\_LDAP\_USER
- GSK\_LDAP\_PASSWORD
- RSE - Algunas directivas opcionales han cambiado en `rsed.envvars`.
  - `(_RSE_JAVAOPTS) -Ddaemon.log`
  - `(_RSE_JAVAOPTS) -Xmx`
  - `SCLMDT_CONF_HOME`
- RSE - Se han añadido directivas opcionales nuevas a `ssl.properties` (desde las versiones 7.5.1.0 y 7.6.0.0).
  - `server_keystore_label`
  - `server_keystore_type`
- RSE - El daemon RSE soporta la autenticación de certificados de cliente X.509 (desde la versión 7.5.1.0), que requiere actualizaciones en la configuración de seguridad y de certificados actual en el momento en que la utilice.
- RSE - Se ha reforzado la seguridad, han fallado peticiones de conexión a causa de errores de PassTicket y FEKAPPL.
- RSE - Ha cambiado la ubicación predeterminada para todos los archivos de anotaciones (anotaciones de daemon y de usuarios).
  - `/var/rdz/logs`
  - `/var/rdz/logs/$LOGNAME`
- RSE - Se ha facilitado un JCL de ejemplo nuevo para reunir la información sobre anotaciones y configuración de Developer for System z.
  - FEKLOGS

## Archivos configurables

Tabla 47 ofrece una visión general de los archivos personalizados en la versión 7.6. Tenga en cuenta que las bibliotecas de ejemplo de Developer for System z, FEK.SFEKSAMP, FEK.SFEKSAMV y `/usr/lpp/rdz/samples/`, se suministran con más miembros personalizables que los indicados aquí, como el código fuente de ejemplo de CARMA y los trabajos para compilarlo.

**Nota:** El trabajo de ejemplo FEKSETUP copia todos los miembros de la lista en conjuntos de datos y directorios diferentes, por omisión en `FEK.#CUST.*` y `/etc/rdz/*`.

Tabla 47. Personalizaciones de la versión 7.6

Miembro/Archivo	Ubicación predeterminada	Finalidad	Notas de migración
FEKSETUP	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para crear conjuntos de datos y directorios y llenarlos con archivos personalizables	Actualizado para incluir nuevos miembros personalizables
JMON	FEK.SFEKSAMP(FEJJJCL) [FEK.#CUST.PROCLIB]	JCL del supervisor de trabajos JES	Opción añadida para cambiar las opciones de LE
FEJJJCL	FEK.SFEKSAMP [FEK.#CUST.PROCLIB(JMON)]	Nombre suministrado para el miembro JMON	Consulte el miembro JMON

Tabla 47. Personalizaciones de la versión 7.6 (continuación)

Miembro/Archivo	Ubicación predeterminada	Finalidad	Notas de migración
RSED	FEK.SFEKSAMP(FEKRSED) [FEK.#CUST.PROCLIB]	JCL para el daemon RSE	none
FEKRSED	FEK.SFEKSAMP [FEK.#CUST.PROCLIB(RSED)]	Nombre suministrado para el miembro RSED	Consulte el miembro RSED
LOCKD	FEK.SFEKSAMP(FEKLCKD) [FEK.#CUST.PROCLIB]	JCL para el daemon de bloqueo	NUEVO, es necesaria personalización
FEKLCKD	FEK.SFEKSAMP [FEK.#CUST.PROCLIB(LOCKD)]	Nombre suministrado para el miembro LOCKD	Consulte el miembro LOCKD
ELAXF*	FEK.SFEKSAMP [FEK.#CUST.PROCLIB]	JCL para construcciones de proyectos remotos, etc.	none
FEKRACF	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para definiciones de seguridad	Actualizaciones menores
FEJCNFG	FEK.SFEKSAMP [FEK.#CUST.PARMLIB]	Archivo de configuración del supervisor de trabajos JES	Se han añadido nuevas directivas opcionales
FEJTSO	FEK.SFEKSAMP [FEK.#CUST.CNTL]	JCL de sometimientos TSO	none
CRA\$VMSG	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para crear el VSAM de mensajes de CARMA	none
CRA\$VDEF	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para crear el VSAM de configuraciones de CARMA	none
CRA\$VSTR	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para crear el VSAM de información personalizada de CARMA	none
CRASUBMT	FEK.SFEKSAMP [FEK.#CUST.CNTL]	CLIST de inicio de proceso por lotes de CARMA	none
CRASUBCA	FEK.SFEKSAMP [FEK.#CUST.CNTL]	CLIST de inicio por lotes de CARMA para CA Endevor® SCM RAM	NUEVO, la personalización es opcional
CRASHOW	FEK.SFEKSAMP [FEK.#CUST.PARMLIB]	Configuración de CARMA para CA Endevor® SCM RAM	NUEVO, la personalización es opcional
CRATMAP	FEK.SFEKSAMP [FEK.#CUST.PARMLIB]	Configuración de CARMA para CA Endevor® SCM RAM	NUEVO, la personalización es opcional
CRANDVRA	FEK.SFEKPROC	REXX de asignación de CARMA para CA Endevor® SCM RAM	NUEVO, la personalización es opcional
CRAISPRX	FEK.SFEKSAMP [FEK.#CUST.CNTL]	Exec de asignación de DD de ejemplo para CARMA utilizando la Pasarela de cliente TSO/ISPF	none
CRA#VSLM	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para crear el VSAM de mensajes de RAM de SCLM	none

Tabla 47. Personalizaciones de la versión 7.6 (continuación)

Miembro/Archivo	Ubicación predeterminada	Finalidad	Notas de migración
CRA#ASLM	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para crear los conjuntos de datos de RAM SCLM	none
CRA#VPDS	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para crear el VSAM de mensajes de RAM de PDS	none
CRA#CRAM	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para compilar el RAM de esqueleto	none
CRA#VCAD	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para crear el VSAM de configuración de CARMA para CA Endevor® SCM RAM	NUEVO, la personalización es opcional
CRA#VCAS	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para crear el VSAM de información personalizada de CARMA para CA Endevor® SCM RAM	NUEVO, la personalización es opcional
CRA#UADD	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para fusionar definiciones del RAM	NUEVO, la personalización es opcional
CRA#UQRY	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para extraer definiciones del RAM	NUEVO, la personalización es opcional
CRAXJCL	FEK.SFEKSAMP [FEK.#CUST.ASM]	Código fuente de ejemplo para sustitución de IRXJCL	none
CRA#CIRX	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para compilar CRAXJCL	none
ADNCSDRS	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para definir el servidor CRD de RESTful en la región primaria CICS	NUEVO, la personalización es opcional
ADNCSDTX	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para definir los ID de transacción alternativos en la región CICS	NUEVO, la personalización es opcional
ADNTXNC	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para crear ID de transacción alternativos	NUEVO, la personalización es opcional
ADNMSGHC	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para compilar ADNMSGHS	Renombrado, era ADNCMSGH
ADNMSGHS	FEK.SFEKSAMP [FEK.#CUST.COBOL]	Código fuente de ejemplo para el manejador de mensajes de conducto	Renombrado, era ADNSMSGH
ADNVCRD	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para crear el repositorio del CRD	Renombrado, era ADNVSAM
ADNCSDWS	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para definir el servidor CRD del Servicio Web en la región primaria CICS	Renombrado, era ADNPCCSD
ADNCSDAR	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para definir el servidor CRD en regiones no primarias CICS	Renombrado, era ADNARCSD

Tabla 47. Personalizaciones de la versión 7.6 (continuación)

Miembro/Archivo	Ubicación predeterminada	Finalidad	Notas de migración
ADNJSPAU	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para actualizar los valores predeterminados del CRD	Se añaden definiciones para el servicio RESTful. Es necesario rehacer las personalizaciones
ADNVMFST	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para crear y definir el repositorio de manifiestos	Renombrado, era ADNMFEST
ELAXMSAM	FEK.SFEKSAMP [FEK.#CUST.PROCLIB]	Procedimiento JCL del espacio de direcciones WLM para el constructor de procedimientos almacenados PL/I y COBOL	none
ELAXMJCL	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para definir el constructor de procedimientos almacenados PL/I y COBOL en DB2	none
FEKAPPCC	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para crear una transacción APPC	none
FEKAPPCL	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para visualizar una transacción APPC	none
FEKAPPX	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para suprimir una transacción APPC	none
FEKLOGS	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para recoger los archivos de anotaciones	NUEVO, la personalización es opcional
rsed.envvars	/usr/lpp/rdz/samples/ [/etc/rdz/]	Variables de entorno RSE	Las copias más antiguas deben sustituirse por esta (deben volver a realizarse personalizaciones).
ISPF.conf	/usr/lpp/rdz/samples/ [/etc/rdz/]	archivo de configuración de la Pasarela de cliente TSO/ISPF	ISP.SISPCLIB añadido a SYSPROC para SCLMDT
CRASRV.properties	/usr/lpp/rdz/samples/ [/etc/rdz/]	Archivo de configuración de CARMA	none
crastart.conf	/usr/lpp/rdz/samples/ [/etc/rdz/]	Archivo de configuración de CARMA para la utilización de CRASTART	none
crastart.endevor.conf	/usr/lpp/rdz/samples/ [/etc/rdz/]	Archivo de configuración de CARMA para utilización de CRASTART para CA Endevor® SCM RAM	NUEVO, la personalización es opcional
ssl.properties	/usr/lpp/rdz/samples/ [/etc/rdz/]	Archivo de configuración SSL de RSE	Se han añadido nuevas directivas opcionales

Tabla 47. Personalizaciones de la versión 7.6 (continuación)

Miembro/Archivo	Ubicación predeterminada	Finalidad	Notas de migración
rsecomm.properties	/usr/lpp/rdz/samples/ [/etc/rdz/]	Archivo de configuración de rastreo de RSE	none
propertiescfg.properties	/usr/lpp/rdz/samples/ [/etc/rdz/]	Archivo de configuración de grupos de propiedades basadas en host	none
projectcfg.properties	/usr/lpp/rdz/samples/ [/etc/rdz/]	Archivo de configuración de proyectos basados en host	none
FMIEXT.properties	/usr/lpp/rdz/samples/ [/etc/rdz/]	Archivo de configuración de Integración del gestor de archivos	Las copias más antiguas deben sustituirse por esta (deben volver a realizarse personalizaciones).
uchars.settings	/usr/lpp/rdz/samples/ [/etc/rdz/]	Archivo de configuración de caracteres no editables	none

## Migrar desde la versión 7.1 a la versión 7.5

### IBM Rational Developer for System z, FMID HHOP750

- La ubicación de instalación de SMP/E predeterminada para los componentes de MVS no ha cambiado y, por ello, sigue siendo FEK.\*.
- La ubicación de instalación de SMP/E predeterminada para los componentes de z/OS UNIX ha cambiado a /usr/lpp/rdz/\*.
- Common Access Repository Manager (CARMA) se ha fusionado con Developer for System z Versión 7.5, eliminando la necesidad de instalarlo como producto por separado.
- SCLM Developer Toolkit se ha fusionado con Developer for System z Versión 7.5, eliminando la necesidad de instalarlo como producto por separado.
- En la versión 7.5, el servicio de Pasarela de cliente TSO/ISPF de ISPF sustituye a la función de SCLM Developer Toolkit utilizada en la versión 7.1 para conectarse al servicio de mandatos TSO. El método de conexión APPC sigue estando soportado.
- En la versión 7.5, el servidor RSE ya no es un proceso gestionado por INETD, sino una tarea iniciada. El servidor RSE utiliza ahora un modelo de servidor único, mientras que, en las versiones anteriores, cada conexión cliente-host tenía un servidor RSE privado.
- Todos los módulos que requieren autorización APF (Supervisor de trabajos JES y SCLM Developer Toolkit) se han trasladado a FEK.SFEKAUTH en la versión 7.5, por lo que requieren una actualización para las definiciones APF existentes.
- El módulo de carga del Supervisor de trabajos JES se ha trasladado a FEK.SFEKAUTH en la versión 7.5, lo cual requiere una actualización del procedimiento de tarea iniciada existente.
- Los módulos de carga de CARMA se han trasladado a bibliotecas nuevas, por lo que requieren una actualización para el script de inicio del servidor CRASUBMT existente.

- Los módulos de carga de SCLM Developer Toolkit se han trasladado a bibliotecas nuevas, por lo que requieren una actualización para las definiciones LINKLIST existentes.
- ELAXFTS0 es un nuevo procedimiento de construcción de ejemplo desde la versión 7.1.1, ELAXFCP1 y ELAXFPP1 son nuevos en la versión 7.5.
- uchars.settings es un archivo de configuración nuevo para caracteres no editables.
- propertiescfg.properties es un archivo de configuración nuevo para grupos de propiedades predeterminadas.
- FEJJCNFG, CRASRV.properties y FMIEXT.properties contienen nuevas directivas opcionales.
- rsed.envvars ha cambiado en la versión 7.5 y debe sustituirse.
- El archivo ISPF.conf de ejemplo suministrado con la versión 7.5 es similar al utilizado por SCLM Developer Toolkit en la versión 7.1.
- Algunas de las personalizaciones existentes del Gestor de despliegue de aplicaciones deben volver a realizarse.
- El gestor de despliegue de aplicaciones ofrece funciones nuevas que requieren personalización.
- Los valores de seguridad para el servidor RSE han cambiado drásticamente en la versión 7.5.
- El perfil de seguridad MVS.MCSOPER.JMON es nuevo para el supervisor de trabajos JES en la versión 7.5.
- El script de inicio de CARMA ha cambiado de nombre y se ha movido a una nueva ubicación, por lo que es necesario actualizar el archivo de configuración CRASRV.properties existente.
- El script de inicio de FMI ha cambiado de nombre y se ha movido a una nueva ubicación, por lo que es necesario actualizar el archivo de configuración FMIEXT.properties existente.
- Se ha añadido un nuevo módulo de carga para el soporte bidireccional y es necesario actualizar la concatenación DFHRPL de CICS existente en caso que no esté utilizando la biblioteca de FEK.SFEKLOAD.
- Ahora, también se describen los cambios realizados en el parámetro MAXPROCUSER de SYS1.PARMLIB(BPXPRMxx).

## Archivos configurables

Tabla 48 ofrece una visión general de los archivos personalizados en la versión 7.5. Tenga en cuenta que las bibliotecas de ejemplo de Developer for System z, FEK.SFEKSAMP, FEK.SFEKSAMV y /usr/lpp/rdz/samples/, se suministran con más miembros personalizables que los indicados aquí, como el código fuente de ejemplo de CARMA y los trabajos para compilarlo.

**Nota:** El trabajo de ejemplo FEKSETUP copia todos los miembros de la lista en conjuntos de datos y directorios diferentes, por omisión en FEK.#CUST.\* y /etc/rdz/\*.

Tabla 48. Personalizaciones de la versión 7.5

Miembro/Archivo	Ubicación predeterminada	Finalidad	Notas de migración
FEKSETUP	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para crear conjuntos de datos y directorios y llenarlos con archivos personalizables	NUEVO, es necesaria personalización

Tabla 48. Personalizaciones de la versión 7.5 (continuación)

Miembro/Archivo	Ubicación predeterminada	Finalidad	Notas de migración
JMON	FEK.SFEKSAMP (FEJJJCL) [FEK.#CUST.PROCLIB]	JCL del supervisor de trabajos JES	STEPLIB cambiado por SFEKAUTH
RSED	FEK.SFEKSAMP (FEKRSED) [FEK.#CUST.PROCLIB]	JCL para el daemon RSE	NUEVO, es necesaria personalización
ELAXF*	FEK.SFEKSAMP [FEK.#CUST.PROCLIB]	JCL para construcciones de proyectos remotos, etc.	ELAXFTSO, ELAXFCP1 y ELAXFPP1 son nuevos
FEKRACF	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para definiciones de seguridad	NUEVO, personalización obligatoria
FEJJCNFG	FEK.SFEKSAMP [FEK.#CUST.PARMLIB]	Archivo de configuración del supervisor de trabajos JES	<ul style="list-style-type: none"> <li>• Algunas directivas han pasado a ser opcionales</li> <li>• Se han añadido nuevas directivas opcionales</li> </ul>
FEJTSO	FEK.SFEKSAMP [FEK.#CUST.CNTL]	JCL de sometimientos TSO	El nombre de trabajo puede ser ahora una variable
CRAISPRX	FEK.SFEKSAMP [FEK.#CUST.CNTL]	Exec de asignación de DD de ejemplo para CARMA utilizando la Pasarela de cliente TSO/ISPF	NUEVO, la personalización es opcional
CRAXJCL	FEK.SFEKSAMP [FEK.#CUST.ASM]	Código fuente de ejemplo para sustitución de IRXJCL	NUEVO, la personalización es opcional
CRA#CIRX	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para compilar CRAXJCL	NUEVO, la personalización es opcional
ADNSMSGH	FEK.SFEKSAMP [FEK.#CUST.COBOL]	Código fuente de ejemplo para el manejador de mensajes de conducto	Las copias más antiguas deben sustituirse por esta (deben volver a realizarse personalizaciones)
ADNPCCSD	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para definir el servidor CRD en la región primaria CICS	Las copias más antiguas deben sustituirse por esta (deben volver a realizarse personalizaciones)
ADNJSPAU	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para actualizar los valores predeterminados del CRD	NUEVO, la personalización es opcional

Tabla 48. Personalizaciones de la versión 7.5 (continuación)

Miembro/Archivo	Ubicación predeterminada	Finalidad	Notas de migración
ADNMFEST	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para crear y definir el repositorio de manifiestos	NUEVO, la personalización es opcional
rsed.envvars	/usr/lpp/rdz/samples/ [/etc/rdz/]	Variables de entorno RSE	Las copias más antiguas deben sustituirse por esta (deben volver a realizarse personalizaciones)
ISPF.conf	/usr/lpp/rdz/samples/ [/etc/rdz/]	archivo de configuración de la Pasarela de cliente TSO/ISPF	Idéntico al archivo ISPF.conf suministrado con SCLMDT en v7.1
CRASRV.properties	/usr/lpp/rdz/samples/ [/etc/rdz/]	Archivo de configuración de CARMA	<ul style="list-style-type: none"> <li>El script de inicio ha cambiado de ubicación y de nombre</li> <li>Se han añadido nuevas directivas opcionales</li> </ul>
crastart.conf	/usr/lpp/rdz/samples/ [/etc/rdz/]	Archivo de configuración de CARMA para la utilización de CRASTART	NUEVO, la personalización es opcional
FMEXT.properties	/usr/lpp/rdz/samples/ [/etc/rdz/]	Archivo de configuración de Integración del gestor de archivos	<ul style="list-style-type: none"> <li>El script de inicio ha cambiado de ubicación y de nombre</li> <li>Se han añadido nuevas directivas opcionales</li> </ul>
uchars.settings	/usr/lpp/rdz/samples/ [/etc/rdz/]	Archivo de configuración de caracteres no editables	NUEVO, la personalización es opcional

## Migrar desde la versión 7.0 a la versión 7.1

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

- La ubicación de instalación de SMP/E predeterminada para los componentes de MVS y z/OS UNIX no ha cambiado y, por ello, sigue siendo FEK.\* y /usr/lpp/wd4z/\*.
- Adición:** opción de configuración - 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 ADNCMSGH
- /usr/lpp/wd4z/rse/lib/FMIEXT.properties
- **Cambio:** se han trasladado los miembros siguientes:
  - SFEKDLL(FEJBDTRX) -> SFEKLOAD(FEJBDTRX)
- **Cambio:** los siguientes miembros personalizables han cambiado:
  - 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 CARMA (Common Access Repository Manager), FMID HCMA710

- La ubicación de instalación de SMP/E predeterminada para los componentes de MVS no ha cambiado y, por ello, sigue siendo CRA.\*.
- **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 siguientes miembros personalizables han cambiado de nombre:
  - samplib CRAREPR -> CRA\$VDEF
  - samplib CRAMREPR -> CRA\$VMMSG
  - 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 siguientes miembros personalizables han cambiado:
  - clist CRASUBMT

## Archivos configurables

La tabla 23 ofrece una visión general de los archivos personalizados en la versión 7.1. Tenga en cuenta que las bibliotecas de ejemplo de CARMA y Developer for System z, CRA.SCRASAMP, FEK.SFEKSAMP y /usr/lpp/wd4z/rse/lib/, se suministran con más miembros personalizables que los indicados aquí, como el código fuente de ejemplo de CARMA y los trabajos para compilarlo.

Tabla 49. Personalizaciones de la versión 7.1

Miembro/Archivo	Ubicación predeterminada	Finalidad	Notas de migración
ELAXF*	FEK.SFEKSAMP	JCL para construcciones de proyectos remotos y otros trabajos	ELAXFADT es nuevo
CRA\$VMSG	CRA.SCRASAMP	JCL para crear el VSAM de mensajes de CARMA	<ul style="list-style-type: none"> <li>• renombrado, era CRAMREPR</li> <li>• se ha actualizado el VSAM creado por este trabajo</li> </ul>
CRA\$VDEF	CRA.SCRASAMP	JCL para crear el VSAM de configuraciones de CARMA	<ul style="list-style-type: none"> <li>• renombrado, era CRAREPR</li> <li>• se ha actualizado el VSAM creado por este trabajo</li> </ul>
CRA\$VSTR	CRA.SCRASAMP	JCL para crear el VSAM de información personalizada de CARMA	renombrado, era CRASREPR
CRASUBMT	CRA.SCRASAMP	CLIST de inicio de proceso por lotes de CARMA	añadir DD CARMALOG
CRA#VSLM	CRA.SCRASAMP	JCL para crear el VSAM de mensajes de RAM de SCLM	renombrado, era CRALREPR
CRA#ASLM	CRA.SCRASAMP	JCL para crear los conjuntos de datos de RAM SCLM	renombrado, era CRASALX
CRA#VPDS	CRA.SCRASAMP	JCL para crear el VSAM de mensajes de RAM de PDS	renombrado, era CRATREPR
CRA#CRAM	CRA.SCRASAMP	JCL para compilar el RAM de esqueleto	renombrado, era CRARAMCM
FEKAPPCC	FEK.SFEKSAMP	JCL para crear una transacción APPC	aprovechar soporte NEST de ISPF
rsed.envvars	/usr/lpp/wd4z/rse/lib/ [/etc/wd4z/]	Variables de entorno RSE	Las copias más antiguas deben sustituirse por esta (deben volver a realizarse personalizaciones)
FMIEXT.properties	/usr/lpp/wd4z/rse/lib/ [/etc/wd4z/]	Archivo de configuración de Integración del gestor de archivos	NUEVO, es necesaria personalización cuando se utiliza



---

## Apéndice A. Configurar SSL y autenticación de X.509

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. Este apéndice también facilita una configuración de ejemplo para admitir que los usuarios se autenticuen con un certificado X.509.

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. 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. Consulte la publicación *Security Server RACF Security Administrator's Guide* (SA22-7683) para obtener más información acerca de los certificados digitales y el protocolo de claves públicas utilizado por SSL.

Las acciones necesarias para configurar las comunicaciones SSL para Developer for 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 del daemon RSE que utilice SSL. También crearemos nuestros propios certificados de seguridad para que los utilicen diferentes componentes de la conexión RSE.

- “Decida dónde desea almacenar los certificados y claves privadas” en la página 296
- “Crear un anillo de claves con RACF” en la página 297
- “Clonar la configuración RSE existente” en la página 299
- “Actualizar rsed.envvars para habilitar la coexistencia” en la página 299
- “Actualizar ssl.properties para habilitar la SSL” en la página 299
- “Activar la SSL creando un daemon RSE nuevo” en la página 300
- “Probar la conexión” en la página 300
- “(Opcional) Añadir soporte de autorización al cliente de X.509” en la página 303
- “(Opcional) Crear una base de datos de claves con gskkyman” en la página 303
- “(Opcional) Crear un almacén de claves con keytool” en la página 306

A lo largo de este apéndice se utiliza un convenio de denominación uniforme:

- Certificado: rdzrse
- Almacenamiento de claves y certificados: rdzssl.\*
- Contraseña: rsessl
- ID de usuario del daemon : stcrse

Algunas 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.

## Decida dónde desea almacenar los certificados y claves privadas

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.

sin embargo, todas las implementaciones siguen el mismo principio. Un mandato genera un par de claves (una clave pública y una clave privada asociada). Este envuelve luego la clave pública en un certificado X.509 autofirmado, que se almacena como cadena de certificados de un solo elemento. Esta cadena de certificados y la clave privada se almacenan como una entrada (identificado por un alias) en un archivo de claves.

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 compatible con SAF (por ejemplo, por RACF). El servidor RSE (que se inicia mediante el daemon) es una aplicación SSL Java y utiliza un archivo de almacén de claves creado por keytool o bien un anillo de claves gestionado por su software de seguridad.

*Tabla 50. Mecanismos de almacenamiento de certificados de SSL*

Almacenamiento de certificados	Creado y gestionado por	Daemon RSE	servidor RSE
anillo de claves	producto de seguridad compatible con SAF	soportado	soportado
base de datos de claves	gskkyman de z/OS UNIX	soportado	/
almacén de claves	Keytool de Java	/	soportado

Para establecer la conexión por medio de SSL, necesitamos tanto el almacén de claves como la base de datos de claves, ya sea como un archivo de z/OS UNIX o como un anillo de claves compatible con SAF:

- almacén de claves (RACF o keytool)
- base de datos de claves (RACF o gskkyman)

### Nota:

- Los anillos de claves compatibles con SAF son el método preferido para gestionar certificados.
- Se puede utilizar un certificado compartido si el daemon RSE y el servidor RSE utilizan el mismo método de gestión de certificados.
- El daemon RSE debe ejecutarse controlado por programa. Utilizar el SSL del sistema implica que SYS1.SIEALNKE debe estar controlado por programa por el software de seguridad.
- 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.

```
STEPLIB=$STEPLIB:SYS1.SIEALNKE
```

Sin embargo, tenga en cuenta lo siguiente:

- El hecho de utilizar STEPLIB en z/OS UNIX afecta negativamente al rendimiento.
- Si una biblioteca de STEPLIB tiene autorización APF, todas deben tener autorización. Las bibliotecas pierden su autorización APF si se mezclan con bibliotecas sin autorización en STEPLIB.
- 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. Consulte la publicación *System SSL Programming* (SC24-5901) para obtener más información.

Consulte la publicación *Security Server RACF Security Administrator's Guide* (SA22-7683) para obtener más información acerca de RACF y los certificados digitales. La documentación relativa a gskkyman se encuentra en la publicación *System SSL Programming* (SC24-5901) y la documentación de keytool está disponible en el sitio <http://java.sun.com/j2se/1.5.0/docs/tooldocs/solaris/keytool.html>.

---

## Crear un anillo de claves con RACF

No realice este paso si utiliza gskkyman para crear la base de datos de claves del daemon RSE y keytool para crear el almacén de claves del servidor RSE.

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.

Consulte la publicación *Security Server RACF Command Language Reference* (SA22-7687) para obtener detalles acerca 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(stcrse)
PERMIT IRR.DIGTCERT.LISTRING CLASS(FACILITY) ACCESS(READ) ID(stcrse)
SETROPTS RACLIST(FACILITY) REFRESH
```

```
RACDCERT ID(stcrse) GENCERT SUBJECTSDN(CN('rdz rse ssl') +
OU('rdz') O('IBM') L('Raleigh') SP('NC') C('US')) +
NOTAFTER(2017-05-21)) WITHLABEL('rdzrse') KEYUSAGE(HANDSHAKE)
```

```
RACDCERT ID(stcrse) ADDRING(rdzssl.racf)
RACDCERT ID(stcrse) CONNECT(LABEL('rdzrse') RING(rdzssl.racf) +
DEFAULT USAGE(PERSONAL))
```

En el ejemplo anterior se empieza por crear los perfiles necesarios y permitir que el ID de usuario STCRSE tenga acceso a los anillos de claves y a los certificados propiedad de ese ID de usuario. El ID de usuario que se utilice debe coincidir con el ID de usuario utilizado para ejecutar para el daemon RSE SSL. El próximo paso consiste en crear un certificado autofirmado con la etiqueta rdzrse . No se necesita ninguna contraseña. Luego, este certificado se añade a un anillo de claves recién creado (rdzssl.racf). Igual que con el certificado, tampoco se necesita una contraseña para el anillo de claves.

El resultado se puede verificar con la opción list siguiente:

```
RACDCERT ID(stcrse) LIST
Información de certificado digital para el usuario STCRSE:

Etiqueta: rdzrse
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=rdz rse ssl.OU=rdz.O=IBM.L=Raleigh.SP=NC.C=US<
Nombre del sujeto:
    >CN=rdz rse ssl.OU=rdz.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: STCRSE
Anillo:
    >rdzssl.racf<

```

## (Opcional) Uso de un certificado firmado

Los certificados pueden ser autofirmados o estar firmados por una autoridad certificadora (CA). Un certificado firmado por una CA implica que la CA garantiza que el propietario del certificado es quién dice ser. El proceso de firma añade las credenciales de la CA (otro certificado) al certificado, constituyendo una cadena de certificados de varios elementos.

Al utilizar un certificado firmado por una CA puede evitar las preguntas de validación de la confianza realizadas por el cliente de Developer for System z, si para el cliente la CA ya es de confianza.

Siga estos pasos para crear y utilizar un certificado firmado por una CA:

1. Cree un certificado autofirmado.  
RACDCERT ID(stcrse) GENCERT WITHLABEL('rdzrse') . . .
  2. Cree una solicitud firmada para este certificado.  
RACDCERT ID(stcrse) GENREQ (LABEL('rdzrse')) DSN(dsn)
  3. Envíe la solicitud firmada a la CA que haya elegido.
  4. Compruebe si ya se conocen las credenciales de la CA (otro certificado).  
RACDCERT CERTAUTH LIST
  5. Marque el certificado de CA como De confianza.  
RACDCERT CERTAUTH ALTER(LABEL('CA cert')) TRUST  
O añada el certificado de CA a la base de datos.  
RACDCERT CERTAUTH ADD(dsn) WITHLABEL('CA cert') TRUST
  6. Añada el certificado firmado a la base de datos; este sustituirá el certificado autofirmado.  
RACDCERT ID(stcrse) ADD(dsn) WITHLABEL('rdzrse') TRUST
- Nota:** NO suprima el certificado autofirmado antes de sustituirlo. Si lo hace, perderá la clave privada que va con el certificado, cosa que anula toda utilidad que tuviera el certificado.
7. Crear un anillo de claves.  
RACDCERT ID(stcrse) ADDRING(rdzssl.racf)
  8. Añada el certificado firmado al anillo de claves.  
RACDCERT ID(stcrse) CONNECT(ID(stcrse) LABEL('rdzrse'))  
RING(rdzssl.racf))
  9. Añada el certificado de CA al anillo de claves.  
RACDCERT ID(stcrse) CONNECT(CERTAUTH LABEL('CA cert'))  
RING(rdzssl.racf))

---

## Clonar la configuración RSE existente

En este paso se crea una nueva instancia de los archivos de configuración de RSE para que la configuración de SSL pueda ejecutarse en paralelo con las existentes. En los mandatos que siguen se presupone que los archivos de configuración se encuentran en /etc/rdz/, que es la ubicación predeterminada utilizada en “Configuración de la personalización” en la página 15.

```
$ cd /etc/rdz
$ mkdir ssl
$ cp rsed.envvars ssl
$ cp ssl.properties ssl
$ ls ssl
rsed.envvars    ssl.properties
```

Los mandatos z/OS UNIX que figuran más arriba crean un subdirectorio llamado `ssl` y lo llenan con los archivos de configuración que requieren cambios. Podemos compartir el resto de archivos de configuración, el directorio de instalación y los componentes de MVS, puesto que no son específicos de la SSL.

Reutilizando la mayor parte de los archivos de configuración existente, podemos centrarnos en los cambios que son realmente necesarios para la configuración de la SSL y evitar tener que volver a realizar la configuración de RSE completa. (Por ejemplo, podemos ahorrarnos definir una nueva ubicación para `ISPF.conf`.)

---

## Actualizar `rsed.envvars` para habilitar la coexistencia

Hasta el momento, las definiciones son una copia exacta de la configuración actual, lo que implica que las anotaciones del daemon RSE nuevo se superponen a los archivos de anotaciones del servidor actual. RSE también necesita saber dónde encontrar los archivos de configuración que no se han copiado en el directorio `ssl`. Ambas emisiones pueden direccionarse por cambios sin importancia a `rsed.envvars`.

```
$ oedit /etc/rdz/ssl/rsed.envvars
-> descomentar y cambiar: -Ddaemon.log=/var/rdz/logs/ssl
-> añadir al final (END):
# -- NECESARIO PARA ENCONTRAR LOS ARCHIVOS DE CONFIGURACIÓN RESTANTES
CFG_BASE=/etc/rdz
CLASSPATH=.:$CFG_BASE:$CLASSPATH
# --
```

Los cambios anteriores definen una nueva ubicación de anotaciones (que el daemon RSE creará en caso que la ubicación de anotaciones no exista). Los cambios también actualizan la `CLASSPATH` de manera que los procesos RSE de la SSL buscarán los archivos de configuración primero en el directorio actual (/etc/rdz/ssl) y luego en el directorio original (/etc/rdz).

---

## Actualizar `ssl.properties` para habilitar la SSL

Actualizando `ssl.properties` el RSE sigue instrucciones para empezar a utilizar la comunicación cifrada de la SSL.

```
$ oedit /etc/rdz/ssl/ssl.properties
-> cambiar: enable_ssl=true
-> descomentar y cambiar: daemon_keydb_file=rdzssl.racf
-> descomentar y cambiar: daemon_key_label=rdzrse
-> descomentar y cambiar: server_keystore_file=rdzssl.racf
-> descomentar y cambiar: server_keystore_label=rdzrse
-> descomentar y cambiar: server_keystore_type=JCERACFKS
```

Los cambios anteriores habilitan la SSL e indican al daemon RSE y al servidor RSE que el certificado (compartido) está almacenado bajo la etiqueta rdzrse en el anillo de claves rdzssl.racf. La palabra clave JCERACFKS indica al servidor RSE que se está utilizando un anillo de claves compatible con SAF como almacén de claves.

---

## Activar la SSL creando un daemon RSE nuevo

Como se ha indicado anteriormente, vamos a crear una segunda conexión que utilizará SSL, lo que implica crear un daemon RSE. El daemon RSE puede ser una tarea iniciada o un trabajo de usuario. Utilizaremos el método del trabajo de usuario para la configuración inicial (prueba). En las instrucciones que siguen se presupone que el JCL de ejemplo se encuentra en FEK.#CUST.PROCLIB(RSED), que es la ubicación predeterminada utilizada en “Configuración de la personalización” en la página 15:

1. Cree un miembro FEK.#CUST.PROCLIB(RSEDSSL) y cópielo en el JCL de ejemplo FEK.#CUST.PROCLIB(RSED).
2. Personalice RSEDSSL añadiendo una tarjeta de trabajo al principio y una sentencia exec al final. Especifique también un número de puerto nuevo (4047) y la ubicación de los archivos de configuración relacionados con SSL (/etc/rdz/ssl), como se muestra en ejemplo de código siguiente. Observe que aplicamos la utilización del ID de usuario STCRSE, ya que a este ID de usuario se le ha otorgado autorización de acceso a los certificados y los anillos de claves en el paso anterior.

```
//RSEDSSL JOB CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1),USER=STCRSE
//*
//* RSE DAEMON - SSL
//*
//RSED      PROC IVP=' ',                                * 'IVP' para realizar una prueba IVP
//          PORT=4047,
//          HOME='/usr/lpp/rdz',
//          CNFG='/etc/rdz/ssl'
//*
//RSE       EXEC PGM=BPXBATS,REGION=0M,TIME=NOLIMIT,
//          PARM='PGM &HOME./bin/rsed.sh &IVP &PORT &CNFG'
//STDOUT    DD SYSOUT=*
//STDERR    DD SYSOUT=*
//PEND
//*
//RSED      EXEC RSED
//*
```

*Figura 61. RSEDSSL - Trabajos de usuario del daemon RSE para SSL*

**Nota:** El ID de usuario asignado al trabajo RSEDSSL debe tener las mismas autorizaciones que el daemon RSE original. El perfil de FACILITY BPX.SERVER y el perfil de PTKTDATA IRRPTAUTH.FEKAPPL.\* son aquí elementos de claves.

---

## Probar la conexión

La configuración del host de SSL está completa y el daemon RSE para SSL puede iniciarse sometiéndolo al trabajo FEK.#CUST.PROCLIB(RSEDSSL) creado anteriormente.

Ahora, puede probarse la configuración nueva conectándose con el cliente de Developer for System z. Dado que hemos creado una nueva configuración (clonando la existente) para que SSL la utilice, hay que definir una nueva conexión en el cliente utilizando el puerto 4047 para el daemon RSE.

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 for System z, si no reconoce el certificado del host o la CA que lo ha firmado, el cliente de Developer for System z preguntará al usuario si el certificado es de confianza.



Figura 62. Diálogo 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:** El daemon RSE y el servidor RSE pueden utilizar dos ubicaciones de certificado diferentes, generando dos certificados distintos y, por consiguiente, dos 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:

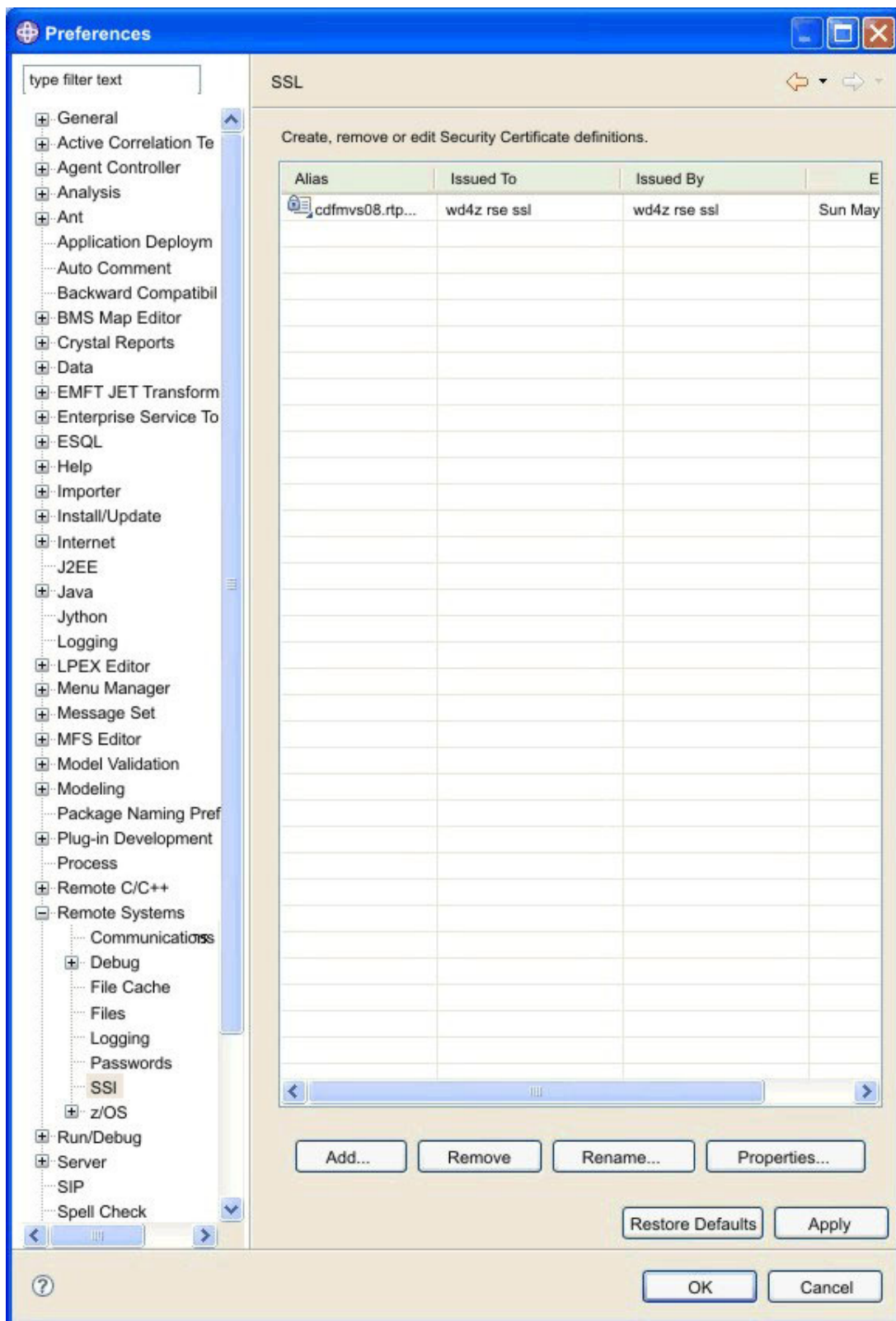


Figura 63. Diálogo Preferencias - SSL

Si la comunicación SSL falla, el cliente devolverá un mensaje de error. Hay más información en los distintos archivos de anotaciones del usuario y del servidor,

como se describe en: “Daemon RSE y anotaciones de la agrupaciones de hebras” en la página 139 y “Anotaciones de usuario de RSE” en la página 140.

---

## (Opcional) Añadir soporte de autorización al cliente de X.509

El daemon RSE admite que los usuarios se autenticuen con un certificado X.509. El uso de una comunicación cifrada con SSL es un requisito previo para utilizar esta función, dado que es una extensión de la autenticación de host con un certificado utilizado en SSL.

Hay varias formas de realizar la autorización de certificados para un usuario, tal como se describe en “Autenticación de cliente mediante certificados X.509” en la página 171. Los siguientes pasos describen la configuración necesaria para soportar el método por el cual su software de seguridad autentica el certificado mediante la ampliación de certificado HostIdMappings.

1. Cambie el certificado que identifica a la autoridad certificadora (CA) utilizada para la firma del certificado del cliente por un certificado de CA de confianza total. Aunque el estado TRUST basta para la validación de un certificado, se realiza el cambio a HIGHTRUST porque se utiliza para la parte de autenticación de certificados del proceso de inicio de sesión.  
`RACDCERT CERTAUTH ALTER(LABEL('HighTrust CA')) HIGHTRUST`
2. Añada el certificado de CA al anillo de claves, `rdzssl.racf`, de manera que esté disponible para validar los certificados de clientes.  
`RACDCERT ID(stcrse) CONNECT(CERTAUTH LABEL('HighTrust CA') +  
RING(rdzssl.racf))`

Esta acción concluye la configuración del software de seguridad para el certificado de CA.

3. Defina un recurso (formato `IRR.HOST.hostname`) en la clase `SERVAUTH` para el nombre de host, `CDFMVS08.RALEIGH.IBM.COM`, definido en la ampliación de `HostIdMappings` de su certificado de cliente.  
`RDEFINE SERVAUTH IRR.HOST.CDFMVS08.RALEIGH.IBM.COM UACC(NONE)`
4. Otorgue al ID de usuario de la tarea iniciada RSE, `STCRSE`, acceso a este recurso con autorización de `LECTURA`.  
`PERMIT IRR.HOST.CDFMVS08.RALEIGH.IBM.COM CLASS(SERVAUTH) +  
ACCESS(READ) ID(stcrse)`
5. Active los cambios en la clase `SERVAUTH`. Utilice el primer mandato si la clase `SERVAUTH` no está todavía activa. Utilice el segundo para renovar una configuración activa.  
`SETROPTS CLASSACT(SERVAUTH) RACLIST(SERVAUTH)  
o bien  
SETROPTS RACLIST(SERVAUTH) REFRESH`

Esta acción concluye la configuración del software de seguridad para ampliación de `HostIdMappings`.

6. Reinicie la tarea iniciada RSE para empezar a aceptar los inicios de sesión de clientes mediante certificados X.509.

---

## (Opcional) Crear una base de datos de claves con gskkyman

No realice este paso si utiliza un anillo de claves compatible con SAF para la base de datos de claves del daemon RSE.

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. Consulte la publicación *System SSL Programming* (SC24-5901) para obtener más información acerca de este tema.

```
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
```

```
$ cd /etc/rdz/ssl
$ gskkyman          Menú de base de datos
```

1 - Crear base de datos nueva

Entre el número de opción: **1**  
Especifique el nombre de la base de datos de claves (pulse Intro para volver al menú): **rdzssl.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/rdz/ssl/rdzssl.kdb.

Pulse Intro para continuar.

Menú de gestión de claves

6 - Crear un certificado autofirmado

Entre el número de opción (pulse Intro para volver al menú anterior): **6**

Tipo de certificado

5 - Certificado de usuario o servidor con clave RSA de 1024 bites

Seleccione el tipo de certificado (pulse Intro para volver al menú): **5**  
Entre la etiqueta (pulse Intro para volver al menú): **rdzrse**  
Entre el nombre de sujeto del certificado  
Nombre común (necesario): **rdz rse ssl**  
Unidad de organización (OU, opcional): **rdz**  
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

0 - Salir del programa

Entre el número de opción (pulse Intro para volver al menú anterior): **0**

```
$ ls -l rdzssl.*
```

```
total 152
```

```
-rw----- 1 IBMUSER SYS1      35080 May 24 14:24 rdzssl.kdb
```

```
-rw----- 1 IBMUSER SYS1       80 May 24 14:24 rdzssl.rdb
```

```
$ chmod 644 rdzssl.*
$ ls -l rdzssl.*
-rw-r--r-- 1 IBMUSER SYS1      35080 May 24 14:24 rdzssl.kdb
-rw-r--r-- 1 IBMUSER SYS1       80 May 24 14:24 rdzssl.rdb
```

En el ejemplo anterior se empieza por crear una base de datos de claves llamada `rdzssl.kdb` con la contraseña `rsessl`. Una vez creada la base de datos, ésta se llena creando un certificado autofirmado, válido durante 10 años (sin contar los días bisiestos). El certificado se almacena bajo la etiqueta `rdzrse` 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. 644 (el propietario tiene acceso de lectura/escritura y los demás tienen acceso de lectura) es una máscara que se puede usar para el mandato **`chmod`**.

El resultado se puede verificar seleccionando la opción **Mostrar información de certificado**, en el submenú **Gestionar claves y certificados**, del siguiente modo:

```
$ gskkyman
```

```
Menú de base de datos
```

```
2 - Abrir base de datos
```

```
Entre el número de opción: 2
```

```
Especifique el nombre de la base de datos de claves (pulse Intro para volver al menú): rdzssl.kdb
```

```
Entre la contraseña de la base de datos (pulse Intro para volver al menú): rsessl
```

```
Menú de gestión de claves
```

```
1 - Gestionar claves y certificados
```

```
Entre el número de opción (pulse Intro para volver al menú anterior): 1
```

```
Lista de claves y certificados
```

```
1 - rdzrse
```

```
Entre el número de etiqueta (Intro para volver al menú de selección, p para lista anterior): 1
```

```
Menú de claves y certificados
```

```
1 - Mostrar información de certificado
```

```
Entre el número de opción (pulse Intro para volver al menú anterior): 1
```

```
Información de certificado
```

```
Etiqueta: rdzrse
ID de registro: 14
ID de registro del emisor: 14
De confianza: Sí
Versión: 3
Número de serie: 45356379000ac997
Nombre del emisor: rdz rse ssl
rdz
IBM
Raleigh
NC
US
```

```

Nombre de sujeto: rdz rse ssl
rdz
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

0 - Salir del programa

Entre el número de opción (pulse Intro para volver al menú anterior): 0

El siguiente ejemplo de `ssl.properties` muestra que las directivas de `daemon_*` difieren del ejemplo de anillo de claves de SAF anterior.

```

$ oedit /etc/rdz/ssl/ssl.properties
-> cambiar: enable_ssl=true
-> descomentar y cambiar: daemon_keydb_file=rdzssl.kdb
-> descomentar y cambiar: daemon_keydb_password=rsessl
-> descomentar y cambiar: daemon_key_label=rdzrse
-> descomentar y cambiar: server_keystore_file=rdzssl.racf
-> descomentar y cambiar: server_keystore_label=rdzrse
-> descomentar y cambiar: server_keystore_type=JCERACFKS

```

Los cambios anteriores habilitan la SSL e indican al daemon RSE que el certificado está almacenado bajo la etiqueta `rdzrse` en la base de datos de claves `rdzssl.kdb` con la contraseña `rsessl`. El servidor RSE sigue utilizando un anillo de claves compatible con SAF.

---

## (Opcional) Crear un almacén de claves con keytool

No realice este paso si utiliza un anillo de claves compatible con SAF para el almacén de claves del servidor RSE.

"`keytool -genkey`" genera un par de claves privadas y un certificado autofirmado coincidente, que está almacenado 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, donde `/usr/lpp/java/J5.0` es el directorio en el que está instalado Java:

```
PATH=$PATH:/usr/lpp/java/J5.0/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, del siguiente modo:

```

$ cd /etc/rdz/ssl
$ keytool -genkey -alias rdzrse -validity 3650 -keystore rdzssl.jks -storepass
rsessl -keypass rsessl
¿Cuál es su nombre y su apellido?
[Desconocido]: rdz rse ssl

```

```

¿Cuál es el nombre de su unidad de organización (OU)?
[Desconocido]: rdz
¿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=rdz rse ssl, OU=rdz, O=IBM, L=Raleigh, ST=NC, C=US? (escriba "yes"
o "no")
[no]: yes
$ ls -l rdzssl.*
-rw-r--r--  1 IBMUSER  SYS1          1224 May 24 14:17 rdzssl.jks

```

El certificado autofirmado creado más arriba es válido durante 10 años (sin contar los días bisiestos). Se almacena en /etc/rdz/ssl/rdzssl.jks utilizando el alias rdzrse. 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, del siguiente modo:

```

$ keytool -list -alias rdzrse -keystore rdzssl.jks -storepass rsessl -v
Nombre de alias: rdzrse
Fecha de creación: May 24, 2007
Tipo de entrada: keyEntry
Longitud de la cadena de certificados: 1
Certificado 1:
Propietario: CN=rdz rse ssl, OU=rdz, O=IBM, L=Raleigh, ST=NC, C=US
Emisor: CN=rdz rse ssl, OU=rdz, 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

```

El siguiente ejemplo de ssl.properties muestra que las directivas de server\_\* difieren del ejemplo de anillo de claves de SAF anterior.

```

$ oedit /etc/rdz/ssl/ssl.properties
-> cambiar: enable_ssl=true
-> descomentar y cambiar: daemon_keydb_file=rdzssl.racf
-> descomentar y cambiar: daemon_key_label=rdzrse
-> descomentar y cambiar: server_keystore_file=rdzssl.jks
-> descomentar y cambiar: server_keystore_password=rsessl
-> descomentar y cambiar: server_keystore_label=rdzrse
-> descomentar y cambiar (opcional): server_keystore_type=JKS

```

Los cambios anteriores habilitan la SSL e indican al servidor RSE que el certificado está almacenado bajo la etiqueta rdzrse en el almacén de claves rdzssl.jks con la contraseña rsessl. El daemon RSE sigue utilizando un anillo de claves compatible con SAF.



---

## Apéndice B. 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 o modificar una configuración existente.

Consulte las publicaciones *Communications Server: IP Configuration Guide* (SC31-8775) y *Communications Server: IP Configuration Reference* (SC31-8776) para obtener más información acerca de la configuración de TCP/IP.

---

### Dependencia del nombre de host

Al utilizar APPC para el servicio de mandatos TSO, Developer for System z depende de que TCP/IP tenga el nombre de host correcto cuando se inicializa. 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 IVP (programa de verificación de la instalación) fekfivpt. El mandato debe devolver datos de salida parecidos a los de este ejemplo (\$ es el indicador de z/OS UNIX):

```
$ fekfivpt
```

```
Wed Jul  2 13:11:54 EDT 2008
uid=1(USERID) gid=0(GROUP)
using /etc/rdz/rsed.envvars
```

```
-----
configuración del resolvente TCP/IP (orden de búsqueda de z/OS UNIX):
-----
```

```
Inicialización de rastreo de resolvente completada -> 2008/07/02 13:11:54.745964
```

```
Valores de resolvente res_init:
Conjunto de datos Tcp/Ip global      = Ninguno
Conjunto de datos Tcp/Ip predeterminado = Ninguno
Conjunto de datos Tcp/Ip local       = /etc/resolv.conf
Tabla de conversión                  = Predeterminada
IDusuario/NombreTrabajo              = USERID
API llamante                         = LE C Sockets
Modalidad llamante                   = EBCDIC
(L) DataSetPrefix = TCPIP
(L) HostName      = CDFMVS08
(L) TcpIpJobName  = TCPIP
(L) DomainOrigin  = RALEIGH.IBM.COM
(L) NameServer    = 9.42.206.2
                  9.42.206.3
(L) NsPortAddr    = 53          (L) ResolverTimeout    = 10
(L) ResolveVia    = UDP         (L) ResolverUdpRetries = 1
(*) Options NDots = 1
(*) SockNoTestStor
(*) AlwaysWto     = NO          (L) MessageCase       = MIXED
(*) LookUp        = DNS LOCAL
```

```
res_init Satisfactoria
```

```
res_init Iniciada: 2008/07/02 13:11:54.755363
```

```
res_init Finalizada: 2008/07/02 13:11:54.755371
```

```
*****
```

```
MVS TCP/IP NETSTAT CS V1R9      Nombre TCPIP: TCPIP      13:11:54
```

```
Tcpip iniciado a las 01:28:36 el 06/23/2008 con IPv6 habilitado
```

```
-----
```

dirección IP del host:

```
-----  
hostName=CDFMVS08  
hostAddr=9.42.112.75  
bindAddr=9.42.112.75  
localAddr=9.42.112.75
```

Satisfactorio, coincidencia de direcciones

---

## 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`.

---

## 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 la publicación *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 la publicación *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 resolventes 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 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 310). 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, tarea o 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 310).

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 la publicación *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 esta información de configuración a Developer for System z

Tal como se ha indicado antes, Developer for System z depende de que TCP/IP tenga el nombre de host correcto en el momento de la inicialización cuando se está utilizando APPC. 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. Tenga en cuenta 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 310, 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 51 en la página 315 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 311, 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 51 en la página 315 también nos indica que si no tenemos nada que hacer, se utiliza 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, debe mantener los archivos de configuración de TCP/IP y del resolvente sincronizados entre sí.

Tabla 51. 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>
Tablas de hosts locales	endhostent endnetent getaddrinfo gethostbyaddr gethostbyname gethostent GetHostNumber GetHostResol GetHostString getnameinfo getnetbyaddr getnetbyname getnetent IsLocalHost Resolve sethostent setnetent	IPv4 <ol style="list-style-type: none"> <li>1. Variable de entorno X_SITE</li> <li>2. Variable de entorno X_ADDR</li> <li>3. /etc/hosts</li> <li>4. userid.HOSTS.xxxxINFO</li> <li>5. jobname.HOSTS.xxxxINFO</li> <li>6. hlq.HOSTS.xxxxINFO</li> </ol> IPv6 <ol style="list-style-type: none"> <li>1. GLOBALIPNODES</li> <li>2. Variable de entorno RESOLVER_IPNODES</li> <li>3. userid.ETC.IPNODES</li> <li>4. jobname.ETC.IPNODES</li> <li>5. hlq.ETC.IPNODES</li> <li>6. DEFAULTIPNODES</li> <li>7. /etc/ipnodes</li> </ol>

**Nota:** Tabla 51 es una copia parcial de una tabla de la publicación *Communications Server: IP Configuration Guide* (SC31-8775). En ese manual encontrará la tabla completa.

## La dirección del host no se resuelve correctamente

Cuando encuentre problemas relacionados con que el Resolvente de TCP/IP no pueda resolver la dirección de host correctamente, probablemente se deba a un archivo de configuración resolvente faltante o incompleto. Una indicación clara de este problema es el mensaje siguiente en `lock.log`:

```
clientip(0.0.0.0) <> callerip(<dirección IP host>)
```

Para verificarlo, ejecute el IVP de TCP/IP fekfivpt, como se describe en Capítulo 7, “Verificación de la instalación”, en la página 109. La sección de configuración del resolvente de la salida será como la del ejemplo siguiente:

Inicialización de rastreo de resolvente completada -> 2008/07/02 13:11:54.745964

```
Valores de resolvente res_init:
Conjunto de datos Tcp/Ip global      = Ninguno
Conjunto de datos Tcp/Ip predeterminado = Ninguno
Conjunto de datos Tcp/Ip local      = /etc/resolv.conf
Tabla de conversión                  = Predeterminada
IDusuario/NombreTrabajo              = USERID
API llamante                        = LE C Sockets
Modalidad llamante                   = EBCDIC
```

Asegúrese de que las definiciones del archivo (o del conjunto de datos) a las que hace referencia “Conjunto de datos Tcp/Ip local” sean correctas.

Este campo estará en blanco si no utiliza un nombre predeterminado para el archivo resolvente de IP (utilizando el orden de búsqueda de z/OS UNIX). Si es así, añada la sentencia siguiente al archivo rsed.envvars, donde <archivo resolvente> o <datos resolventes> representa el nombre del archivo resolvente de IP:

```
RESOLVER_CONFIG=<archivo resolvente>
```

o bien

```
RESOLVER_CONFIG='<conjunto de datos resolventes>'
```

---

## Apéndice C. Configurar INETD

Este apéndice se propone ayudarle a resolver algunos problemas comunes que pueden surgir al configurar INETD, o durante la tarea de comprobar o modificar una configuración existente. Developer for System z utiliza INETD para la función REXEC/SSH.

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 pueden ocupar varias 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
server_program_arguments
```

#### [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 cuatro 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, debe 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. Consulte la publicación *Communications Server IP Configuration Guide* (SC31-8775) para obtener más información acerca de 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

Debe 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. Consulte la publicación *UNIX System Services Planning* (GA22-7800) para obtener más información acerca de inittab.

## BPXBATCH

El método de inicio BPXBATCH funciona para las tareas iniciadas y trabajos 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 el ejemplo de código siguiente 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,REGION=0M,
//          PARM='SH ps -e | grep inetd | cut -c 1-10 | xargs -n 1 kill'
//*
//INETD    EXEC PGM=BPXBATCH,REGION=0M,
//          PARM='PGM /usr/sbin/inetd &PRM'
//STDERR   DD SYSOUT=*
//* STDIN, STDOUT y STDENV toman por defecto el valor /dev/null
//*
```

Figura 64. 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. Desde z/OS 1.7, se puede asignar SYSOUT a STDOUT y STDERR. Consulte la publicación *UNIX System Services Command Reference* (SA22-7802) para obtener más información acerca de BPXBATCH.
- 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 utilizar (si el usuario que los emite tiene autorización suficiente) para detener e iniciar INETD (# es el indicador de z/OS UNIX):

```
# ps -e | grep inetd
7 ?          0:00 /usr/sbin/inetd
# kill 7
# _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 “humanos”, 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)

Consulte la publicación *UNIX System Services Command Reference* (SA22-7802) para obtener más información acerca de los mandatos **extattr** y **su**. Consulte la publicación *UNIX System Services Planning* (GA22-7800) para obtener más información acerca de la clase UNIXPRIV y los perfiles BPX.\* de la clase FACILITY. Consulte la publicación *Security Server RACF Security Administrator's Guide* (SA22-7683) para obtener más información acerca de las definiciones de segmento OMVS y la clase PROGRAM.

---

## Requisitos de Developer for System z

Developer for System z depende de INETD para la gestión de REXEC y/o SSH. También puede imponer requisitos adicionales sobre la configuración de INETD descrita anteriormente.

REXEC (o SSH) se utiliza para las dos finalidades siguientes, descritas en la sección “(Opcional) Utilizar REXEC (o SSH)” en la página 102.

- acciones remotas (basadas en host) en subproyectos z/OS UNIX
- método de inicio alternativo del servidor RSE

Las acciones remotas de subproyectos z/OS UNIX no requieren valores especiales. Sin embargo, el método de inicio de RSE alternativo no requiere valores especiales.

### 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.

### REXEC (o SSH)

El daemon REXEC (o SSH) que se inicia mediante INETD cuando un cliente se conecta al puerto 512 (o 22, respectivamente) 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 REXEC (o SSH) (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 for System z, cuyo valor predeterminado es /usr/lpp/rdz/\*.
- Acceso de lectura y ejecución (READ y EXECUTE) a los directorios de configuración de Developer for System z, cuyo valor predeterminado es /etc/rdz/\*.

---

## Apéndice D. 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 o modificar una configuración existente.

Consulte las publicaciones *MVS Planning: APPC/MVS Management* (SA22-7599) y *MVS Initialization and Tuning Reference* (SA22-7592) para obtener información adicional acerca de la gestión de APPC y los miembros parmlib descritos más adelante.

Tenga en cuenta 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 los siguientes 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 for System z activa un nuevo TP que se llama FEKFRSRV, y al que nos referimos como servicio de mandatos TSO.

El trabajo siguiente 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 65. 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 tarea iniciada de

VTAM). Utilice el mandato de consola **d net,appls** 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 66. SYS1.SAMPLIB(ATBAPPL)

Consulte la publicación *Communications Server IP SNA Network Implementation Guide* (SC31-8777) para obtener más información acerca de la configuración de 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.

El ejemplo de código que sigue es un miembro SYS1.PARMLIB(APPCPMxx) que puede utilizarse para el servicio de mandatos TSO.

```

LUADD
  ACBNAME(MVSLU01)
  BASE
  SCHED(ASCH)
  TPDATA(SYS1.APPCTP)
  SIDEINFO DATASET(SYS1.APPCSI)

```

Figura 67. 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 lo siguiente:

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).

**Nota:** El mandato de operador **D APPC,LU,ALL** mostrará todas las definiciones de LU activa y marcará la LU base.

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: “rsed.envvars, archivo de configuración de RSE” en la página 32.

---

## 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 for System z.

Para el servicio de mandatos TSO, también hay que especificar las especificaciones predeterminadas en las secciones OPTIONS y TPDEFAULT.

El ejemplo de código que sigue es un miembro SYS1.PARMLIB(ASCHPMxx) que puede utilizarse 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 68. *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.
- El mandato de operador **D ASCH,ALL** mostrará todas las clases del planificador de transacciones APPC activas.

---

## 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 aún no está activo. Para iniciar APPC/MVS, especifique los siguientes mandatos de consola (siendo xx los dos 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 el siguiente 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**. Consulte la publicación *MVS System Commands* (GC28-1781) para obtener más información acerca de 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 for System z, como se describe en: “(Opcional) Transacción APPC para el servicio de mandatos TSO” en la página 104.

La manera documentada de definir la transacción APPC consiste en personalizar y someter FEK.#CUST.JCL(FEKAPPC).

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 documento *APPC and WebSphere Developer for System z* (SC23-5885-00) está disponible en la biblioteca internet de Developer for System z, en el sitio <http://www-306.ibm.com/software/awdtools/rdz/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 for 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'
```

---

## (Opcional) Opciones de configuración alternativas

Developer for System z da soporte a opciones de configuración alternativas de APPC y VTAM, algunas de las cuales se describen en esta sección.

### Nombre de transacción alternativo

El nombre de transacción alternativo para el servicios de mandatos TSO es FEKFRSRV, como se describe en la sección “(Opcional) Transacción APPC para el servicio de mandatos TSO” en la página 104. Como se describe en la misma sección, este nombre puede cambiarse al definir la transacción en APPC.

Tenga en cuenta que cambiar el nombre de transacción en APPC implica asignar el nuevo nombre a `_FEKFSCMD_TP_NAME_` en `rsed.envvars`, como se describe en la sección “`rsed.envvars`, archivo de configuración de RSE” en la página 32.

### Varias LU

APPC es un protocolo de comunicaciones que permite que un programa (el nodo asociado) interactúe con un programa del host (el nodo local). Con Developer for System z, tanto el nodo asociado (servidor de mandatos TSO) como el nodo local (servidor RSE) están activos en el mismo sistema z/OS. Y, por omisión, ambos utilizan la misma definición de LU (BASE) para comunicarse entre sí.

Puede especificar un nombre de LU asociada alternativo para el servicio de mandatos TSO en la directiva `_FEKFSCMD_PARTNER_LU_` de `rsed.envvars`, como se describe en la sección “`rsed.envvars`, archivo de configuración de RSE” en la página 32. Tenga en cuenta que no puede cambiar la LU local, que debe ser siempre una LU BASE válida (tener las palabras clave BASE y SCHED).

### Seguridad de LU

VTAM da soporte a un configuración de APPC segura, en la que la comunicación entre la LU asociada y la local debe definirse para el software de seguridad.

Estos e activa añadiendo `VERIFY=REQUIRED` a la definición VTAM de la LU local (BASE). Las definiciones de seguridad deben realizarse en la clase APPCLU, como se describe en la publicación *MVS Planning: APPC/MVS Management* (SA22-7599).

Tenga en cuenta que, si esta configuración está activa en VTAM y la configuración del software de seguridad no está completa, la comunicación con el servicio de mandatos TSO no podrá inicializarse sin mensajes en las anotaciones del sistema

que indiquen que VTAM ha rehusado configurar la conexión. La prueba de IVP de APPC (fekfi vpa) fallará con el mensaje “Código de retorno 1 - Anomalía de asignación sin reintento”.



---

## Apéndice E. Requisitos

Este apéndice enumera los prerrequisitos y correquisitos del host para esta versión de Developer for System z.

Consulte el manual *Rational Developer for System z Prerequisites* (SC23-7659) en la biblioteca en línea del Developer for System z en <http://www-01.ibm.com/software/awdtools/rdz/library/> si desea obtener una lista actualizada de los requisitos necesarios y opcionales.

Los productos enumerados en esta sección están todos disponibles en el momento de publicación de este manual. Consulte el sitio web IBM Software Support Lifecycle <http://www.ibm.com/software/support/lifecycle/>, para ver si un producto seleccionado está todavía disponible cuando desee utilizar la función de Developer for System z relacionada.

---

### Prerrequisitos del host z/OS

Para utilizar Developer for System z debe tener el entorno siguiente con los prerrequisitos adecuados:

#### z/OS

Uno de los niveles siguientes debe estar instalado en el host:

Número de programa	Nombre de producto	Arreglos PTF o niveles de servicio necesarios
5694-A01	z/OS v 1.11	ISPF: <ul style="list-style-type: none"><li>• APAR OA29489 (Pasarela de cliente TSO/ISPF) PTF UA51713</li></ul> TCP/IP: <ul style="list-style-type: none"><li>• No se necesitan arreglos PTF ni niveles de servicio</li></ul>
5694-A01	z/OS v 1.10	ISPF: <ul style="list-style-type: none"><li>• APAR OA29489 (Pasarela de cliente TSO/ISPF) PTF UA51712</li></ul> TCP/IP: <ul style="list-style-type: none"><li>• APAR PK74282 (aumento de almacenamiento fijo de CSM) PTF UK41810</li></ul>
5694-A01	z/OS v 1.9	ISPF: <ul style="list-style-type: none"><li>• APAR OA29489 (Pasarela de cliente TSO/ISPF) PTF UA51687</li></ul> TCP/IP: <ul style="list-style-type: none"><li>• APAR PK74282 (aumento de almacenamiento fijo de CSM) PTF UK41812</li></ul>

Número de programa	Nombre de producto	Arreglos PTF o niveles de servicio necesarios
5694-A01	z/OS v 1.8	<p>ISPF:</p> <ul style="list-style-type: none"> <li>• APAR OA20345 (salida de mandato anidada) PTF UA33575</li> <li>• APAR OA20449 (añadir soporte de NESTMACS) PTF UA34052</li> <li>• APAR OA29489 (Pasarela de cliente TSO/ISPF) PTF UA51686</li> </ul> <p>TCP/IP:</p> <ul style="list-style-type: none"> <li>• APAR PK74282 (aumento de almacenamiento fijo de CSM) PTF UK41811</li> </ul>

El sitio web correspondiente al producto es:

<http://www-03.ibm.com/systems/z/os/zos/>

**Notas:**

1. Para las acciones remotas (basadas en host) de los subproyectos z/OS UNIX, es necesario que la versión z/OS UNIX de REXEC o SSH esté activo en el host.
2. z/OS incluye los componentes siguientes que deben instalarse, configurarse y ser operativos:
  - Interactive System Productivity Facility (ISPF)  
– <http://www-01.ibm.com/software/awdtools/ispf/>
  - Language Environment  
– <http://www-03.ibm.com/servers/eserver/zseries/zos/le/>
  - RACF o un producto de seguridad equivalente  
– <http://www-03.ibm.com/servers/eserver/zseries/zos/racf/>
  - Componente VTAM de IBM Communications Server  
– <http://www-01.ibm.com/software/network/commserver/zos/>
  - Componente IP Services de IBM Communications Server  
– <http://www-01.ibm.com/software/network/commserver/zos/>
  - Binder
  - APPC (opcional)

**Nota:**

- APPC es un prerequisite obligatorio si el servicio para ISPF APAR OA29489 no está disponible en su sistema host.
- APPC puede sustituirse por las funciones de Pasarela de cliente ISPF proporcionada en ISPF for z/OS 1.10 y está disponible en ISPF for z/OS 1.8 y 1.9 con los PTF adecuados aplicados.

## SMP/E

A fin de instalar Developer for System z, uno de los niveles siguientes debe estar instalado:

Número de programa	Nombre de producto	Arreglos PTF o niveles de servicio necesarios
5655-G44	IBM System Modification Program Extended (SMP/E) for z/OS v 3.5	No se necesitan arreglos PTF ni niveles de servicio
5655-G44	IBM System Modification Program Extended (SMP/E) for z/OS v 3.4	No se necesitan arreglos PTF ni niveles de servicio

El sitio web correspondiente al producto es:

<http://www-03.ibm.com/systems/z/os/zos/smpe/>

## SDK for z/OS Java 2 Technology Edition

A fin de soportar aplicaciones que utilizan el Explorador de sistemas remotos (RSE), uno de los niveles siguientes debe estar instalado en el host:

Número de programa	Nombre de producto	Arreglos PTF o niveles de servicio necesarios
5655-R32	IBM 64-bit SDK for z/OS, Java 2 Technology Edition, v 6.0	release de servicio 7
5655-R31	IBM 31-bit SDK for z/OS, Java 2 Technology Edition, v 6.0	release de servicio 7
5655-N99	IBM 64-bit SDK for z/OS, Java 2 Technology Edition, v 5.0	release de servicio 11
5655-N98	IBM 31-bit SDK for z/OS, Java 2 Technology Edition, v 5.0	release de servicio 11

El sitio web correspondiente al producto es:

<http://www.ibm.com/servers/eserver/zseries/software/java/>

**Nota:** Es necesario aplicar el PTF para Developer for System z APAR PM07305 cuando se utiliza una versión de 64 bits de Java. El PTF está disponible a través de la página de servicio recomendada de Developer for System z, <http://www-01.ibm.com/support/docview.wss?rs=2294&context=SS2QJ2&uid=swg27006335>.

---

## Correquisitos de host z/OS

Para soportar características específicas de Developer for System z, se necesitan los productos enumerados en esta sección y el software adicional indicado. El cliente de estación de trabajo de Developer for System z se puede instalar satisfactoriamente sin estos requisitos. Sin embargo, los requisitos de host indicados deben estar instalados y ser operativos en tiempo de ejecución para que la característica correspondiente funcione como es debido.

## z/OS

Número de programa	Nombre de producto	Arreglos PTF o niveles de servicio necesarios
5694-A01	z/OS v 1.11	HLASM No se necesitan arreglos PTF ni niveles de servicio  XL C/C++ No se necesitan arreglos PTF ni niveles de servicio  SCLM No se necesitan arreglos PTF ni niveles de servicio  LE (PL/I) No se necesitan arreglos PTF ni niveles de servicio  TN3270 No se necesitan arreglos PTF ni niveles de servicio
5694-A01	z/OS v 1.10	HLASM No se necesitan arreglos PTF ni niveles de servicio  XL C/C++ No se necesitan arreglos PTF ni niveles de servicio  SCLM No se necesitan arreglos PTF ni niveles de servicio  LE (PL/I) No se necesitan arreglos PTF ni niveles de servicio  TN3270 No se necesitan arreglos PTF ni niveles de servicio
5694-A01	z/OS v 1.9	HLASM No se necesitan arreglos PTF ni niveles de servicio  XL C/C++ No se necesitan arreglos PTF ni niveles de servicio  SCLM • APAR OA27379 (Búsqueda SCLM) PTF UA46330 + UA46331, UA46332, UA46333, UA46334 (dependiente de idioma)  LE (PL/I) No se necesitan arreglos PTF ni niveles de servicio  TN3270 No se necesitan arreglos PTF ni niveles de servicio

Número de programa	Nombre de producto	Arreglos PTF o niveles de servicio necesarios
5694-A01	z/OS v 1.8	<p>HLASM No se necesitan arreglos PTF ni niveles de servicio</p> <p>XL C/C++ No se necesitan arreglos PTF ni niveles de servicio</p> <p>SCLM</p> <ul style="list-style-type: none"> <li>• APAR OA21104 (modalidad de constr. informativa) PTF UA35046 + UA35056, UA35057, UA35058 o UA35059 (depende del lenguaje)</li> <li>• APAR OA16924 (mejorar SCLMINFO) PTF UA29772 + UA29922, UA29923, UA29924 o UA29925 (depende del lenguaje)</li> <li>• APAR OA16804 (añadir soporte id usuario sustituto) PTF UA33524 + UA33533, UA33534, UA33535 o UA33536 (depende del lenguaje)</li> </ul> <p>LE (PL/I)</p> <ul style="list-style-type: none"> <li>• APAR PK41552 (mensajes de PL/I nuevos para Developer for System z ) PTF UK24482 (inglés) o UK24483 (japonés)</li> </ul> <p>TN3270 No se necesitan arreglos PTF ni niveles de servicio</p>

El sitio web correspondiente al producto es:

<http://www-03.ibm.com/systems/z/os/zos/>

**Nota:**

1. JES3 v 1.10 o superior es un correquisito para los usuarios de JES3 que deseen utilizar el soporte del Supervisor de trabajos para ver la salida de los trabajos activos.
2. El Assembler de alto nivel (HLASM) debe estar instalado en el host con el nivel de servicio listado, para compilar los programas del assembler desarrollados o editados en Developer for System z.

El sitio web correspondiente al producto es:

<http://www.ibm.com/software/awdtools/hlasm/>

3. El compilador XL C/C++ debe estar instalado en el host con el nivel de servicio listado, para compilar los programas del C/C++ desarrollados o editados en Developer for System z.

El sitio web correspondiente al producto es:

<http://www.ibm.com/software/awdtools/czos/>

4. SCLM debe estar instalado en el host con el nivel de servicio listado para soportar SCLM Developer Toolkit.

El sitio web correspondiente al producto es:

<http://www.ibm.com/software/awdtools/sclmsuite/sclm/>

**Nota:**

- El APAR OA16804 sólo es necesario si desea utilizar la construcción segura, la promoción y el despliegue.
- APAR OA26997 sólo es necesario para el soporte de seguridad de miembros.
- APAR OA27379 sólo es necesario para el soporte de seguridad de miembros o la funcionalidad de búsqueda SCLM.

5. Language Environment debe estar instalado en el host con el nivel de servicio listado para soportar Enterprise Service Tools para PL/I.

El sitio web correspondiente al producto es:

<http://www-03.ibm.com/servers/eserver/zseries/zos/le/>

6. TN3270 debe estar instalado en el host con el nivel de servicio listado para soportar en el emulador de Conexión de host. TN3270 es una parte del componente IP Services de IBM Communications Server.

El sitio web correspondiente al producto es:

<http://www-01.ibm.com/software/network/commserver/zos/>

## COBOL compiler

Para compilar los programas COBOL desarrollados o editados en Developer for System z, uno de los niveles siguientes debe estar instalado en el host:

Número de programa	Nombre de producto	Arreglos PTF o niveles de servicio necesarios
5655-S71	IBM Enterprise COBOL for z/OS v 4.2	No se necesitan arreglos PTF ni niveles de servicio
5655-S71	IBM Enterprise COBOL for z/OS v 4.1	No se necesitan arreglos PTF ni niveles de servicio
5535-G53	IBM Enterprise COBOL for z/OS v 3.4	No se necesitan arreglos PTF ni niveles de servicio

El sitio web correspondiente al producto es:

<http://www.ibm.com/software/awdtools/cobol/zos/>

**Nota:** IBM Enterprise COBOL for z/OS v es necesario para que Enterprise Service Tools genere una conversión XML compilada que utiliza la posibilidad XML PARSE basada en XMLSS en COBOL.

## PL/I compiler

Para compilar los programas PL/I desarrollados o editados en Developer for System z, uno de los niveles siguientes debe estar instalado en el host:

Número de programa	Nombre de producto	Arreglos PTF o niveles de servicio necesarios
5655-H31	IBM Enterprise PL/I for z/OS v 3.9	No se necesitan arreglos PTF ni niveles de servicio
5655-H31	IBM Enterprise PL/I for z/OS v 3.8	No se necesitan arreglos PTF ni niveles de servicio

Número de programa	Nombre de producto	Arreglos PTF o niveles de servicio necesarios
5655-H31	IBM Enterprise PL/I for z/OS v 3.7	No se necesitan arreglos PTF ni niveles de servicio
5655-H31	IBM Enterprise PL/I for z/OS v 3.6	No se necesitan arreglos PTF ni niveles de servicio

El sitio web correspondiente al producto es:

<http://www.ibm.com/software/awdtools/pli/plizos/>

## Debug Tool for z/OS

Para soportar la depuración remota desde Developer for System z, uno de los niveles siguientes debe estar instalado en el host:

Número de programa	Nombre de producto	Lenguaje de programación	Informes APAR, arreglos PTF o niveles de servicio necesarios
5655-V50	IBM Debug Tool for z/OS V10.1	COBOL, PL/I, C/C++, assembler y características adicionales	Todo el mantenimiento disponible
5655-U27	IBM Debug Tool for z/OS V9.1	COBOL, PL/I, C/C++, assembler y características adicionales	Todo el mantenimiento disponible
5655-S16	IBM Debug Tool Utilities and Advanced Functions for z/OS V8.1.0	COBOL, PL/I, C/C++, assembler y características adicionales	Todo el mantenimiento disponible
5655-S17	IBM Debug Tool for z/OS V8.1.0	COBOL, PL/I, Assembler, C/C++	Todo el mantenimiento disponible

**Nota:** El soporte para configuraciones de depuración CICS en IBM Rational Developer for System z v7.6.1 o una versión superior requiere IBM Debug Tool v10.1 o v9.1 (número de PTF - UK52904).

El sitio web correspondiente al producto es:

<http://www.ibm.com/software/awdtools/debugtool/>

**Nota:** Debug Tool Utilities y funciones avanzadas (DTU&AF) es un superconjunto de Debug Tool.

Iniciando la versión 9, Debug Tool para z/OS y Debug Tool Utilities y funciones avanzadas se han fusionado en una única oferta.

## CICS Transaction Server

Para soportar aplicaciones con sentencias CICS incluidas, uno de los niveles siguientes debe estar instalado en el host:

Número de programa	Nombre de producto	Arreglos PTF o niveles de servicio necesarios
5655-S97	IBM CICS Transaction Server for z/OS v 4.1	No se necesitan arreglos PTF ni niveles de servicio
5697-E93	IBM CICS Transaction Server for z/OS v 3.2	UK34221
5697-E93	IBM CICS Transaction Server for z/OS v 3.1	UK15767, UK15764, UK11782, UK11294, UK12233, UK12521, UK15261, UK15271, UK34221, UK34078

El sitio web correspondiente al producto es:

<http://www.ibm.com/software/http/cics/tserver/>

**Nota:**

- El servidor CICS Transaction Server requiere configuración adicional para trabajar con la Debug Tool.
- La interfaz RESTful disponible en CICS Transaction Server v 4.1 o superior es necesaria para soportar las características del Gestor de despliegue de aplicaciones, de Service Component Architecture y de Enterprise Service Tools que son nuevas para IBM Rational Developer for System z v 7.6 o superior.
- CICS Transaction Server v 3.2 o superior es necesario para poder utilizar numerosas características de las herramientas de servicio de empresa (EST).

Para obtener una lista completa de los requisitos específicos en tiempo de ejecución, consulte la documentación de las herramientas de servicio de empresa (EST) en el Information Center de IBM Rational Developer for System z en <http://publib.boulder.ibm.com/infocenter/ratdevz/v7r6/>.

- CICS Transaction Server v 3.1 con el servicio UK34221 es el mínimo para el Gestor de despliegue de aplicaciones.

## IMS

Para soportar aplicaciones que utilizan las comunicaciones de datos y la base de datos de IMS, uno de los niveles siguientes debe estar instalado en el host:

Número de programa	Nombre de producto	Arreglos PTF o niveles de servicio necesarios
5635-A02	IBM IMS v 11.1	No se necesitan arreglos PTF ni niveles de servicio
5635-A01	IBM IMS v 10.1	No se necesitan arreglos PTF ni niveles de servicio
5655-J38	IBM IMS v 9.1	No se necesitan arreglos PTF ni niveles de servicio

El sitio web correspondiente al producto es:

<http://www.ibm.com/software/data/ims/ims/>

**Nota:**

- Para poder funcionar con Debug Tool, se necesita una configuración adicional de IMS.
- Es necesaria la versión 10.1 o una versión superior de IMS, IMS Connect y IMS SOAP Gateway para Enterprise Service Tools.

## DB2 for z/OS

Para soportar DB2, uno de los niveles siguientes debe estar instalado en el host:

Número de programa	Nombre de producto	Arreglos PTF o niveles de servicio necesarios
5635-DB2	IBM DB2 for z/OS v 9.1	No se necesitan arreglos PTF ni niveles de servicio
5625-DB2	IBM DB2 Universal Database for z/OS v 8.1	No se necesitan arreglos PTF ni niveles de servicio

El sitio web correspondiente al producto es:

<http://www.ibm.com/software/data/db2/zos/>

## Rational Team Concert para System z

Para el control de fuente basadas Jazz mediante proyectos remotos de Developer for System z, debe estar instalado el siguiente nivel.

Número de programa	Nombre de producto	Arreglos PTF o niveles de servicio necesarios
5724-V82	Rational Team Concert para System z Server v 2.0	<p>FMID HAHA200 – Team Server</p> <ul style="list-style-type: none"> <li>• UK54064</li> <li>• UK54071</li> <li>• UK54073</li> <li>• UK54095</li> <li>• UK54098</li> </ul> <p>FMID HAHB200 – Kit de herramientas</p> <ul style="list-style-type: none"> <li>• UK54065</li> <li>• UK54066</li> <li>• UK54099</li> </ul> <p>FMID HAHC200 – Supervisor de trabajos</p> <ul style="list-style-type: none"> <li>• No se necesitan arreglos PTF ni niveles de servicio</li> </ul> <p>FMID HAHD200 – Agente BuildForge</p> <ul style="list-style-type: none"> <li>• UK54097</li> </ul>

El sitio web correspondiente al producto es:

<http://www-01.ibm.com/software/awdtools/rtc/>

**Nota:** Rational Team Concert Server v 1.0 o Rational Team Concert para System z Server v 1.0.1 proporciona soporte selectivo para algunas funciones de Developer for System z, como los proyectos locales. Recomendamos Rational

Team Concert para System z Server v 2.0.0.2 para conseguir una experiencia más integrada y con todas las características.

## Gestor de archivos

Para soportar la integración del Gestor de archivos, uno de los niveles siguientes debe estar instalado en el host:

Número de programa	Nombre de producto	Arreglos PTF o niveles de servicio necesarios
5655-U29	IBM File Manager para z/OS v 10.1	<ul style="list-style-type: none"><li>UK54428</li></ul>

El sitio web correspondiente al producto es:

<http://www.ibm.com/software/awdtools/filemanager/>

## Analizador de errores

Para soportar la integración del Analizador de errores, los niveles siguientes debe estar instalado en el host:

Número de programa	Nombre de producto	Arreglos PTF o niveles de servicio necesarios
5655-V51	IBM Fault Analyzer v 10.1	No se necesitan arreglos PTF ni niveles de servicio
5655-U28	IBM Fault Analyzer v 9.1	No se necesitan arreglos PTF ni niveles de servicio
5655-S15	IBM Fault Analyzer v 8.1	No se necesitan arreglos PTF ni niveles de servicio

El sitio web del producto relacionado es el siguiente:

<http://www.ibm.com/software/awdtools/faultanalyzer/>

## REXX

Para utilizar SCLM Developer Toolkit, uno de los niveles siguientes debe estar instalado en el host:

Número de programa	Nombre de producto	Arreglos PTF o niveles de servicio necesarios
5695-014	IBM Library for REXX on zSeries v 1.4	No se necesitan arreglos PTF ni niveles de servicio
5695-014	IBM Library for REXX on zSeries Alternate Library v 1.4.0 (FMIDs HWJ9143, JWJ9144)	No se necesitan arreglos PTF ni niveles de servicio

Una versión de REXX/370 Alternate Library está disponible en el sitio web del producto, en

<http://www.ibm.com/software/awdtools/rexx/rexxzseries/>

## Ported tools

IBM Ported Tools for z/OS debe estar instalado (en z/OS UNIX) para utilizar sftp o scp para realizar un despliegue seguro en SCLM Developer Toolkit.

En el sitio web del producto hay una versión de IBM Ported Tools for z/OS disponible:

[http://www-03.ibm.com/servers/eserver/zseries/zos/unix/port\\_tools.html](http://www-03.ibm.com/servers/eserver/zseries/zos/unix/port_tools.html)

## Ant

Apache Ant debe estar instalado (en z/OS UNIX) para hacer construcciones JAVA/J2EE en SCLM Developer Toolkit.

Apache Ant es una herramienta de construcción basada en Java, de código fuente abierto que puede descargar desde el sitio web del producto:

<http://ant.apache.org/>

## Endevor<sup>®</sup>

CA Endevor<sup>®</sup> Software Change Manager Release 12 debe estar instalado para poder utilizar Developer for System z Interface for CA Endevor<sup>®</sup> SCM.

CA Endevor<sup>®</sup> SCM es un producto de CA. El sitio web del producto relacionado es:

<http://www.ca.com/us/products/product.aspx?ID=259>



---

## Bibliografía

---

### Publicaciones a las que se hace referencia

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

*Tabla 52. Publicaciones a las que se hace referencia*

Título de la publicación	Número de pedido	Referencia	Sitio Web de referencia
Java Diagnostic Guide	SC34-6650	Java 5.0	<a href="http://www.ibm.com/developerworks/java/jdk/diagnosis/">http://www.ibm.com/developerworks/java/jdk/diagnosis/</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>
Directorio de programa para IBM Rational Developer for System z	GI11-8627 (GI11-8298)	Developer for System z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/">http://www-306.ibm.com/software/awdtools/rdz/library/</a>
Rational Developer for System z Common Access Repository Manager Developer's Guide	SC23-7660	Developer for System z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/">http://www-306.ibm.com/software/awdtools/rdz/library/</a>
Rational Developer for System z Requisitos previos	SC23-7659	Developer for System z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/">http://www-306.ibm.com/software/awdtools/rdz/library/</a>
Rational Developer for System z Inicio rápido de configuración de host	GI11-8628 (GI11-9201)	Developer for System z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/">http://www-306.ibm.com/software/awdtools/rdz/library/</a>
Rational Developer for System z Guía de planificación de host	GI11-7839 (GI11-8296)	Developer for System z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/">http://www-306.ibm.com/software/awdtools/rdz/library/</a>
Guía del administrador de SCLM Developer Toolkit	SC11-3815-00 (SC23-9801)	Developer for System z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/">http://www-306.ibm.com/software/awdtools/rdz/library/</a>
APPC and WebSphere Developer for System z	SC23-5885	Documento	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/">http://www-306.ibm.com/software/awdtools/rdz/library/</a>
Communications Server IP Configuration Guide	SC31-8775	z/OS 1.9	<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 Reference	SC31-8776	z/OS 1.9	<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 1.9	<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 1.9	<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 1.9	<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 Operations	SC31-8779	z/OS 1.9	<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 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>

Tabla 52. Publicaciones a las que se hace referencia (continuación)

Título de la publicación	Número de pedido	Referencia	Sitio Web de referencia
DFSMS Macro Instructions for Data Sets	SC26-7408	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
DFSMS Using data sets	SC26-7410	z/OS 1.9	<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 1.9	<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 1.9	<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 1.9	<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 1.9	<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 1.9	<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 1.9	<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 1.9	<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 1.9	<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 1.9	<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 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
TSO/E Customization	SA22-7783	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
TSO/E REXX Reference	SA22-7790	z/OS 1.9	<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 1.9	<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 1.9	<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 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Using REXX and z/OS UNIX System Services	SA22-7806	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Resource Definition Guide	SC34-6430	CICSTS 3.1	<a href="http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html">http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html</a>
Resource Definition Guide	SC34-6815	CICSTS 3.2	<a href="http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html">http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html</a>
Resource Definition Guide	SC34-7000	CICSTS 4.1	<a href="https://publib.boulder.ibm.com/infocenter/cicsts/v4r1/index.jsp?topic=/com.ibm.cics.ts.home.doc/library/library_html.html">https://publib.boulder.ibm.com/infocenter/cicsts/v4r1/index.jsp?topic=/com.ibm.cics.ts.home.doc/library/library_html.html</a>
RACF Security Guide	SC34-6454	CICSTS 3.1	<a href="http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html">http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html</a>

Tabla 52. Publicaciones a las que se hace referencia (continuación)

Título de la publicación	Número de pedido	Referencia	Sitio Web de referencia
RACF Security Guide	SC34-6835	CICSTS 3.2	<a href="http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplbooks.html">http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplbooks.html</a>
RACF Security Guide	SC34-7003	CICSTS 4.1	<a href="https://publib.boulder.ibm.com/infocenter/cicsts/v4r1/index.jsp?topic=/com.ibm.cics.ts.home.doc/library/library_html.html">https://publib.boulder.ibm.com/infocenter/cicsts/v4r1/index.jsp?topic=/com.ibm.cics.ts.home.doc/library/library_html.html</a>
Language Reference	SC27-1408	Enterprise COBOL para z/OS	<a href="http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplbooks.html">http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplbooks.html</a>

En este documento se hace referencia a los siguientes sitios Web:

Tabla 53. Sitios Web a los que se hace referencia

Descripción	Sitio Web de referencia
Information Center de Developer for System z	<a href="http://publib.boulder.ibm.com/infocenter/ratdevz/v7r6/index.jsp">http://publib.boulder.ibm.com/infocenter/ratdevz/v7r6/index.jsp</a>
Soporte de Developer for System z	<a href="http://www-306.ibm.com/software/awdtools/rdz/support/">http://www-306.ibm.com/software/awdtools/rdz/support/</a>
Biblioteca de Developer for System z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/">http://www-306.ibm.com/software/awdtools/rdz/library/</a>
Página inicial de Developer for System z	<a href="http://www-306.ibm.com/software/awdtools/rdz/">http://www-306.ibm.com/software/awdtools/rdz/</a>
Servicio recomendado de Developer for System z	<a href="http://www-01.ibm.com/support/docview.wss?rs=2294&amp;context=SS2QJ2&amp;uid=swg27006335">http://www-01.ibm.com/support/docview.wss?rs=2294&amp;context=SS2QJ2&amp;uid=swg27006335</a>
Solicitud de mejoras de Developer for System z	<a href="https://www.ibm.com/developerworks/support/rational/rfe/">https://www.ibm.com/developerworks/support/rational/rfe/</a>
Biblioteca internet de z/OS	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Information Center de CICSTS	<a href="https://publib.boulder.ibm.com/infocenter/cicsts/v4r1/index.jsp">https://publib.boulder.ibm.com/infocenter/cicsts/v4r1/index.jsp</a>
Descargar Apache Ant	<a href="http://ant.apache.org/">http://ant.apache.org/</a>
Documentación de keytool de Java	<a href="http://java.sun.com/j2se/1.5.0/docs/tooldocs/solaris/keytool.html">http://java.sun.com/j2se/1.5.0/docs/tooldocs/solaris/keytool.html</a>
Página inicial de soporte de CA	<a href="https://support.ca.com/">https://support.ca.com/</a>

## Publicaciones informativas

Las publicaciones siguientes pueden serle de utilidad para entender aspectos de configuración de los componentes de host obligatorios:

Tabla 54. Publicaciones informativas

Título de la publicación	Número de pedido	Referencia	Sitio Web de referencia
ABCs of z/OS System Programming Volume 9 (z/OS UNIX)	SG24-6989	Redbook	<a href="http://www.redbooks.ibm.com/">http://www.redbooks.ibm.com/</a>
System Programmer's Guide to: Workload Manager	SG24-6472	Redbook	<a href="http://www.redbooks.ibm.com/">http://www.redbooks.ibm.com/</a>
TCPIP Implementation Volume 1: Base Functions, Connectivity, and Routing	SG24-7532	Redbook	<a href="http://www.redbooks.ibm.com/">http://www.redbooks.ibm.com/</a>

Tabla 54. Publicaciones informativas (continuación)

Título de la publicación	Número de pedido	Referencia	Sitio Web de referencia
TCPIP Implementation Volume 3: High Availability, Scalability, and Performance	SG24-7534	Redbook	<a href="http://www.redbooks.ibm.com/">http://www.redbooks.ibm.com/</a>
TCP/IP Implementation Volume 4: Security and Policy-Based Networking	SG24-7535	Redbook	<a href="http://www.redbooks.ibm.com/">http://www.redbooks.ibm.com/</a>

---

## Glosario

### A

**Acción de bloquear.** Bloquea un miembro.

**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.

**atributo bidireccional.** Tipo de texto, orientación del texto, intercambio numérico e intercambio simétrico.

### B

**Base de datos.** Conjunto de elementos de datos interrelacionados o independientes, almacenados conjuntamente para servir a una o más aplicaciones.

**Biblioteca de carga.** Biblioteca que contiene módulos de carga.

**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).

### 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.

**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.

**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

**Depurar.** Detectar, diagnosticar y eliminar errores en programas.

**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.

### E

.

### F

.

### G

.

# H

.

# I

**ID de acción.** Identificador numérico de una acción, entre 0 y 999

**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.

**Instancia de repositorio.** Proyecto o componentes que existe en un SCM.

## 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

**Lista de tareas.** Lista de procedimientos que se pueden ejecutar mediante un único flujo de control.

# M

**Memoria intermedia de error.** Parte del almacenamiento utilizado para contener temporalmente la información de salida de errores.

# N

**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.

**Nombre de shell.** Nombre de la interfaz de shell.

# O

.

# P

## 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.

**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.

**Perspectiva.** Proporciona una interfaz para gestionar sistemas remotos utilizando convenciones similares a ISPF.

**Petición de construcción.** Petición procedente del cliente para realizar una transacción de construcción.

## Q

.

## R

**RAM.** Gestor de acceso a repositorios

### 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 estados 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.

## S

**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.

**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.

### 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.

**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.

**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.

**Sidedeck.** Biblioteca que publica las funciones de un programa DLL. Los nombres de entradas y de módulos se almacenan en la biblioteca una vez compilado el código fuente.

**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.

## T

**Transacción de construcción.** Trabajo iniciado en MVS para realizar construcciones después haberse recibido una petición de construcción procedente del cliente.

## U

**URL.** Localizador uniforme de recursos.

## V

**Vista Consola de salida.** Visualiza la salida de un proceso y permite proporcionar entrada de teclado para un proceso.

**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

**Vista Navegador.** Vista jerárquica de los recursos que hay en el entorno de trabajo.

**Vista repositorios.** Muestra la ubicación de los repositorios CVS que se han añadido al entorno de trabajo.

**Vista Salida.** Muestra los mensajes, parámetros y resultados relacionados con los objetos con los que se esté trabajando.

**Vista Servidores.** Visualiza una lista de todos los servidores, así como las configuraciones asociadas a ellos.

## W

.

## X

.

## Y

.

## Z

.

---

## Avisos de documentación de IBM Rational Developer for System z

© Copyright IBM Corporation - 2010

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

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:

Licencia de propiedad intelectual  
Ley de propiedad intelectual y legal  
IBM Japan, Ltd.  
3-2-12, Roppongi, Minato-ku, Tokyo 106-8711 Japan

El párrafo siguiente no se aplica en el Reino Unido ni en 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 O CONDICIONES 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 licenciatarios 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:

Dept. de propiedad intelectual del software Rational  
IBM Corporation  
3039 Cornwallis Road, PO Box 12195  
Research Triangle Park, NC 27709  
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 mediante 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 ilustran 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. Los programas de ejemplo se proporcionan "TAL CUAL", sin ningún tipo de garantía. IBM no se hace responsable de los daños que se hayan podido causar debido al uso de los programas de ejemplo.

---

## Reconocimientos de marcas registradas

IBM, el logotipo de IBM e `ibm.com` son marcas comerciales o marcas registradas de International Business Machines Corp., registradas en muchas jurisdicciones de todo el mundo. Otros nombres de productos y servicios pueden ser marcas registradas de IBM u otras empresas. Hay una lista actual de marcas registradas de IBM disponible en la web en [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml)

Rational es una marca registrada de International Business Machines Corporation y Rational Software Corporation en Estados Unidos y/o en otros países.

Intel® y Pentium® son marcas registradas de Intel Corporation en los Estados Unidos y/o en otros países.

Microsoft®, Windows® y el logotipo de Windows son marcas registradas de Microsoft Corporation en los Estados Unidos y/o en otros países.

Java y todas las marcas y logotipos basados en Java son marcas registradas de Sun Microsystems, Inc., en Estados Unidos y en otros países.

UNIX es una marca registrada de The Open Group en los Estados Unidos y/o en otros países.



---

# Índice

## Caracteres Especiales

.dstoreMemLogging 138  
.dstoreTrace 138  
/etc/inittab, inicialización de z/OS UNIX, INETD 321  
/etc/rc, inicio de z/OS UNIX INETD 321  
\_RSE\_CMDSERV\_OPTS, Definir parámetros de inicio Java adicionales con 47  
\_RSE\_JAVAOPTS, Definir parámetros de inicio Java adicionales con 42  
\_RSE\_PORTRANGE 161

## A

acceso a las bibliotecas del sistema, Mejorar 245  
Acceso condicional a los archivos de spool 169  
acciones basadas en host remoto, subproyectos z/OS UNIX 103  
Acciones condicionales en los trabajos 166  
Acciones en trabajos - limitaciones de ejecución 168  
Activar Common Access Repository Manager 51  
Activar el Gestor de acceso a repositorios del Gestor de configuraciones de software de CA Endevor® 63  
Activar el RAM (Repository Access Manager) de esqueleto 63  
Activar el RAM (Repository Access Manager) de PDS 63  
Activar el RAM (Repository Access Manager) de SCLM 63  
Activar los RAM (Repository Access Managers) de ejemplo 62  
Actualizaciones de SCLM para SCLMDT 87  
actualizar privilegios, no administradores del sistema 201  
ADM, personalización 73  
admin, SCLMDT 87  
ADNCSDAR, no primarias de RESTful y no primarias del servicio web 76  
ADNCSDRS, primaria de RESTful 75  
ADNCSDTX, ID de cambio de RESTful 76  
ADNCSDWS, primaria de servicio Web 78  
ADNJSPAU, Programa de utilidad administrativo 257  
ADNJSPAU, programa de utilidad administrativo de CICS 74  
ADNMSGHC, manejador de mensajes de conducto 77

ADNMSGHS, manejador de mensajes de conducto 77  
ADNTXNC, ID de cambio de RESTful 76  
ADNVCRD, repositorio de CRD 74  
ADNVMFST, repositorio de manifiesto 79  
Ajustar CRASERV.properties 60  
Ajustar ISPF.conf 61  
almacén de claves con keytool, Crear 306  
almacenamiento de ejemplo, análisis de utilización 226  
Almacenamiento dinámico Java de tamaño fijo 247  
alternativo del servidor CARMA, Inicio 60  
análisis de utilización, almacenamiento de ejemplo 226  
Analizador de faltas  
    correquisitos 342  
anillo de claves, Crear con RACF 297  
anomalía, Apertura de conjuntos de datos de MVS 156  
anomalía de apertura de conjuntos de datos MVS 156  
anotaciones, agrupaciones de hebras 139  
anotaciones, SCLM Developer Toolkit 142  
anotaciones de agrupaciones de hebras 139  
anotaciones de auditoría, gestionadas por el daemon RSE 165  
Anotaciones de CARMA  
    rsecomm.log 142  
Anotaciones de Common Access Repository Manager (CARMA) 142  
Anotaciones de instalación de recursos CICS 255  
Anotaciones de Integración del analizador de errores  
    fa.log 141  
    rsecomm.log 141  
Anotaciones de Integración del gestor de archivos  
    rsecomm.log 141  
Anotaciones de prueba IVP  
    fekfivpi.log 143  
    fekfivps.log 143  
Anotaciones de prueba IVP de fekfivpi fekfivpi.log 143  
anotaciones de SCLM Developer Toolkit rsecomm.log 142  
Anotaciones de servicio de mandatos TSO 142  
Anotaciones de transacción APPC 142  
Anotaciones de usuario de RSE  
    .dstoreMemLogging 140  
    .dstoreTrace 140  
    ffs.log 140  
    ffsget.log 140

Anotaciones de usuario de RSE (continuación)  
    ffsput.log 140  
    lock.log 140  
    rmt\_class\_loader.cache.jar 140  
    rsecomm.log 140  
    stderr.log 140  
    stdout.log 140  
Anotaciones del daemon de bloqueo 139  
Anotaciones del daemon RSE 139  
Anotaciones del supervisor de trabajos JES 139  
Anotar y configurar el análisis mediante FEKLOGS 137  
Ant  
    correquisitos 343  
Ant, Instalar y personalizar 86  
Apache Ant  
    correquisitos 343  
APF, autorización 150  
aplicaciones, Desarrollo de 246  
aplicaciones soportadas 3  
APPC, Configurar 325  
APPC, lista de comprobación de transacción 105  
APPCPMxx 327  
Archivo de configuración de anotaciones, rsecomm.properties 96  
archivo de configuración de la Pasarela de cliente TSO/ISPF 47  
archivo de configuración de RSE, rsed.envvars 32  
archivo de configuración del supervisor de trabajos JES, FEJJCNFG 27  
archivo temporal INETD /etc/inetd.pid 321  
archivos configurables 284, 292  
archivos configurados previamente, hacer copia de seguridad  
    Versión 7.0 279  
    Versión 7.1 279  
    Versión 7.5 279  
archivos controlados por programa UNIX para RSE, Definir 188  
archivos controlados por programa z/OS UNIX para RSE, Definir 188  
archivos de anotaciones  
    .dstoreMemLogging 138  
    .dstoreTrace 138  
    audit.log 138  
    fa.log 138  
    fekfivpi.log 138  
    fekfivps.log 138  
    ffs.log 138  
    ffsget.log 138  
    ffsput.log 138  
    lock.log 138  
    rmt\_class\_loader.cache.jar 138  
    rsecomm.log 138  
    rsedaemon.log 138  
    rseserver.log 138

- archivos de anotaciones (*continuación*)
  - serverlogs.count 138
  - stderr.log 138
  - stdout.log 138
- archivos de anotaciones de agrupaciones de hebras RSE
  - audit.log 140
  - rsedaemon.log 140
  - rserver.log 140
  - serverlogs.count 140
  - stderr\*.log 140
  - stdout\*.log 140
- archivos de anotaciones de daemon RSE
  - audit.log 140
  - rsedaemon.log 140
  - rserver.log 140
  - serverlogs.count 140
  - stderr\*.log 140
  - stdout\*.log 140
- archivos de configuración, Developer for System z 176
- Archivos de configuración de resolvente base 311
- archivos de configuración diferentes con niveles de software idénticos 274
- archivos de spool, Acceso condicional a 169
- Archivos de vuelco 143
- archivos ISPF.conf, utilizar con varias configuraciones 267
- ASCHPMxx 328
  - MAX 237
- ASSIZEMAX 183
- atributo del sistema de archivos, SETUID 148
- atributo del sistema de archivos SETUID 148
- audit.log 139
- autenticación, configurar SSL y X.509 295
- Autenticación de cliente mediante certificados X.509 171
- Autenticación del supervisor de trabajos JES 160
- autenticación por daemon RSE 174
- autenticación por software de seguridad 172
- autenticación x.509, configurar 295
- autorización de APF
  - FEK.SFEKAUTH 180
- Autorización de control de programa 149

## B

- base de datos de claves, Crear con gskkyman 303
- básica, personalización 15
- bibliotecas controladas por programa MVS para RSE, Definir 186
- Bibliotecas de tiempo de ejecución de Language Environment 246
- bibliotecas del sistema, Mejorar el acceso a 245
- bites de permiso de z/OS UNIX 148
- bits de permiso, z/OS UNIX 148
- BPXBATCH, inicio de INETD 322

- BPXPRMxx 243
  - INADDRANYCOUNT 236
  - MAXASSIZE 153, 183, 235
  - MAXFILEPROC 235
  - MAXMMAPAREA 235
  - MAXPROCSYS 156, 234
  - MAXPROCUSER 156, 234
  - MAXSOCKETS 236
  - MAXTHREADS 234
  - MAXTHREADTASKS 234
  - MAXUIDS 156, 235
- BPXPRMxx, establecer límites en
  - MAXASSIZE 16
  - MAXPROCUSER 16
  - MAXTHREADS 16
  - MAXTHREADTASKS 16

## C

- CA Endeavor® SCM RAM 63
- CA Endeavor® SCM RAM, personalizar 69
- cambios de APPC, Activar los 329
- cambios de archivo de configuración, otros productos 279
- cambios de DB2 90
- cambios de PROCLIB 90
- Cambios del Gestor de carga de trabajo 89
- Cambios entre la versión 7.0 y la versión 7.1 291
- Capa de sockets seguros, Cifrado de comunicaciones mediante 161
- Capa de sockets seguros, Configurar 295
- Caracteres no alfanuméricos, diagrama de sintaxis 135
- Caracteres no editables, personalizar para manejar 101
- Caracteres no editables,
  - uchars.settings 101
- cargas de trabajo, Gestión de 247
- CARMA, activar 51
- CARMA, Anotaciones 142
- CARMA (Common Access Repository Manager), FMID HCMA710 292
- CARMA, FMID HCMA710 292
- CARMA, interfaz RSE con 53
- CEE.SCEELPA
  - SYS1.PARMLIB(LPALSTxx) 246
- certificado firmado, autofirmado o firmado por una autoridad certificadora 298
- certificado X.509 160
- certificado X.509 de terceros 160
- certificados, autenticación de cliente mediante X.509 171
- certificados X.509, autenticación de cliente mediante 171
- certificados y claves privadas, decida dónde almacenar 296
- CICS, anotaciones de instalación de recursos 255
- CICS, región de conexión primaria 75
- CICS, regiones de conexión no primarias 76
- CICS no primarias, regiones de conexión 76
- CICS Transaction Server
  - correquisitos 339
- CICSplex SM Business Application Services (BAS) 254
- cifrado, ssl.properties 93
- Cifrado de comunicaciones mediante SSL 161
- cifrado SSL de RSE, ssl.properties, propiedades de daemon 93
- propiedades de servidor 93
- clasificación de carga de trabajo, WLM 203
- CLASSPATH 274
- cliente, lista de comprobación 13
- cliente, Lista de comprobación 13
- clonar la configuración RSE existente 299
- COBOL
  - comprobación remota 147
- COBOL Compiler
  - correquisitos 338
- coexistencia, actualizar rsed.envvars para habilitar la coexistencia 299
- COMMNDxx, añadir tareas iniciadas 17
- Common Access Repository Manager, Activar 51
- compartimiento de clases, habilitar en máquinas virtuales Java (JVM) 249
- compartimiento de clases entre máquinas virtuales Java (JVM) 248
- componentes, Instalar y configurar 3
- componentes del host, instalar y configurar 3
- Componentes opcionales 48
- comprobación de POE 162, 174
- comprobación de puerto de entrada 162, 174
- comprobación de seguridad, Mejorar el rendimiento de la 247
- comunicación, cifrada con 170, 256
- comunicación cifrada, SSL 170, 175, 256
- Comunicación cifrada con SSL 170, 175, 256
- Comunicación externa 162
- comunicación externa con puertos especificados, limitar 161
- Comunicación interna 163
- comunicaciones mediante SSL, Cifrado 161
- condicional a los archivos de spool, Acceso 169
- Conducto, manejador de mensajes 77
- conexión, daemon de bloqueo 116
- conexión, Seguridad 161
- conexión de configuración de host de Capa de sockets seguros, Probar la 300
- conexión de configuración de host SSL, Probar la 300
- conexión de daemon de bloqueo 116
- conexión de host, Emulador 157
- conexión de SCLM Developer Toolkit, verificar
  - comprobaciones de SCLMDT 118
- Conexión del daemon RSE 115
- Conexión del servicio de mandatos TSO (APPC) 118

Conexión del supervisor de trabajos  
   JES 115  
 conexión rehusada 156  
 Conexión rehusada 156  
 conexión REXEC, verificar 120  
 conexión SCLMDT, verificar 118  
 configuración, idéntica en todo un  
   sysplex 273  
 configuración, Resolución de  
   problemas 137  
 configuración APPC segura, VTAM 330  
 configuración de ejemplo 241  
   definir límites 242  
   determinar los límites mínimos 242  
   recuento de agrupaciones de  
     hebras 242  
 configuración de la personalización 15  
 configuración de los productos y el  
   software necesarios 9  
 configuración de REXEC 103  
 configuración de SSH 103  
 configuración de TCP/IP, netstat 114  
 configuración del rastreo,  
   rsecomm.properties 96  
 configuración del rastreo RSE,  
   rsecomm.properties, 96  
 Configuración del supervisor de trabajos  
   JES  
     GEN\_CONSOLE\_NAME 168  
 Configuración idéntica en todo un  
   sysplex 273  
 configurar los componentes del host 3  
 conjunto de datos, Definir perfiles 180  
 consideraciones, ID de usuario 9  
 consideraciones acerca de la  
   migración 279  
 Consideraciones acerca de la  
   migración 3  
 consideraciones acerca de los ajustes 213  
 consideraciones de CICSTS 253  
 consideraciones de predespliegue 12  
 consideraciones de utilización,  
   APPC 107  
 consideraciones de utilización de  
   APPC 107  
 consideraciones de WLM xvi, 203  
 consideraciones relativas a la  
   preconfiguración 9  
 Consideraciones relativas a la  
   preinstalación 4  
 Consideraciones relativas a la  
   seguridad 159  
 Consideraciones sobre el  
   rendimiento 245  
 consideraciones sobre el servidor 10  
 consideraciones sobre los ID de  
   usuario 9  
 consultar una Lista de certificados  
   revocados (CRL)  
   rsed.envvars 172  
   variables de entorno CRL 172  
 contraseña e ID de usuario 160  
 contraseña para una sola vez e ID de  
   usuario 160  
 control de auditoría  
   \_RSE\_HOST\_CODEPAGE 165  
   daemon.log 165

control de auditoría (*continuación*)  
   enable.audit.log 165  
   opciones de audit.\* 165  
 controladas por programa MVS para RSE,  
   Definir bibliotecas 186  
 conversión, Tablas de 312  
 conversión de nombres, SCLM 84  
 conversión de nombres  
   largos/abreviados, actualizaciones de  
   rsed.envvars para 86  
 Conversión de nombres  
   largos/abreviados, SCLM 84  
 correquisitos  
   Analizador de faltas 342  
   Ant 343  
   Apache Ant 343  
   CICS Transaction Server 339  
   COBOL Compiler 338  
   DB2 341  
   Debug Tool 339  
   File Manager 342  
   host 335  
   IMS 340  
   Java 2 Technology Edition 335  
   PL/I Compiler 338  
   Ported tools 343  
   Rational Team Concert para System  
     z 341  
   REXX 342  
   SDK for z/OS 335  
   z/OS 336  
 correquisitos, Developer for System  
   z 333  
 correquisitos de host 335  
 Correquisitos de z/OS  
   C/C++ 337  
   High Level Assembler 337  
   Language Environment 338  
   SCLM 337  
 CRA#ASLM, RAM de SCLM de  
   ejemplo 63  
 CRA#CIRX, IRXJCL versus CRAXJCL 71  
 CRA#CRAM, RAM de esqueleto de  
   ejemplo 63  
 CRA#UADD, componentes CARMA 52,  
   64  
 CRA#UQRY, componentes CARMA 52,  
   64  
 CRA#VCAD, Endeavor SCM RAM 64  
 CRA#VCAS, Endeavor SCM RAM 64  
 CRA\$VDEF, componentes CARMA 52  
 CRA\$VMSG, componentes CARMA 52,  
   64  
 CRA#VPDS, RAM (Repository Access  
   Manager) de PDS de ejemplo 63  
 CRA#VSLM, RAM de SCLM de  
   ejemplo 63  
 CRA\$VSTR, componentes CARMA 52  
 CRAISPRX, archivo de configuración de  
   ISPF 47  
 CRAISPRX, inicio de CARMA alternativo  
   de ISPF 61  
 CRANDVRA, personalizar 68  
 CRASERV.properties, Ajustar 60  
 CRASRV.properties  
   clist.dsname 54  
   crastart.configuration.file 54

CRASRV.properties (*continuación*)  
   crastart.steplib 54  
   crastart.stub 54  
   crastart.syslog 54  
   crastart.tasklib 54  
   crastart.timeout 54  
   port.range 54  
   port.start 54  
   startup.script.name 54  
 CRASRV.properties, ajustar 56, 58, 65, 67  
 CRASTART, inicio alternativo del  
   servidor CARMA mediante 57  
 CRASTART, inicio de CA Endeavor® SCM  
   RAM mediante 67  
 crastart.conf, ajustar 58  
 crastart.conf, inicio de crastart de  
   CARMA 58  
 crastart.endeavor.conf, ajustar 67  
 CRASUBCA, ajustar 65  
 CRASUBMT, inicio de proceso por lotes  
   de CARMA 56  
 CRAXJCL, IRXJCL versus CRAXJCL 71  
 CRD, repositorio 74  
 crear LSTRANS.FILE 84  
 cron, limpieza de directorios de  
   WORKAREA 108

## D

daemon, Bloqueo 23  
 daemon, RSE 22  
 daemon de bloque, Anotaciones 139  
 Daemon de bloque, mandato Iniciar 126  
 daemon de bloqueo  
   mensajes de consola 132  
 Daemon de bloqueo 23, 198  
 daemon de bloqueo, LOCKD 109  
 daemon de bloqueo (LOCKD) 191  
 Daemon de bloqueo, mandato  
   Modificar 131  
 daemon REXEC, permisos de ID de  
   usuario cuando lo inicia INETD 324  
 daemon RSE 10, 22, 162  
   mensajes de consola 132  
 daemon RSE, Anotaciones 139  
 daemon RSE, autenticación por 174  
 daemon RSE, Conexión 115  
 Daemon RSE, mandato Iniciar 126  
 Daemon RSE, mandato Modificar 128  
 daemon RSE, RSED 109  
 daemon RSE (RSED) 191  
 daemon RSE y anotaciones de  
   auditoría 165  
 daemon SSH, permisos de ID de usuario  
   cuando lo inicia INETD 324  
 datos de auditoría  
   acciones anotadas 165  
 DB2  
   correquisitos 341  
 DB2, cambios 90  
 DB2, procedimiento almacenado 89  
 Debug Tool  
   correquisitos 339  
 Debug Tool, IBM, correquisito 20  
 definiciones de puerto,  
   PROFILE.TCPIP 320

- definiciones de puerto
  - PROFILE.TCPIP 320
- definiciones de recursos, varios 236
- definiciones de recursos CICS,
  - administrador 253
- definiciones de recursos CICS,
  - desarrollador 253
- definiciones de recursos clave 233
  - rsed.envvars 233
  - SYS1.PARMLIB(BPXPRMxx) 234
- Definiciones de seguridad 26, 177
- definiciones de seguridad, Lista de
  - comprobación 178
- definiciones de varios recursos 236
  - FEJJCENFG 237
  - SYS1.PARMLIB(ASCHPMxx) 237
  - SYS1.PARMLIB(IEASYSxx) 237
  - SYS1.PARMLIB(IVTPRMxx) 237
  - tarjeta EXEC, servidor JCL 236
- definiciones disponibles para el
  - resolvente 315
- Definiciones locales disponibles para el
  - resolvente 315
- Definir archivos controlados por
  - programa z/OS UNIX para el
    - servidor 188
- Definir bibliotecas controladas por
  - programa MVS para RSE 186
- Definir comprobación de puerto de
  - entrada para RSE 174
- Definir el soporte de PassTicket para
  - RSE 187
- Definir servidor RSE como z/OS UNIX
  - seguro 186
- Dependencia del nombre de host 309
- Desarrollo de aplicaciones 246
- Developer for System z, qué es 191
- Developer for System z, visión general de
  - componentes
    - representación gráfica 191
- Diagnóstico de mensajes de error de
  - IRZ 93
- diagrama de sintaxis, Caracteres no
  - alfanuméricos y espacios en
    - blanco 135
- diagrama de sintaxis, Cómo leer 134
- diagrama de sintaxis, Longitud superior a
  - una línea 135
- diagrama de sintaxis, Operandos 134
- diagrama de sintaxis, Seleccionar más de
  - un operando 135
- diagrama de sintaxis, Símbolos 134
- dirección de host no resuelta, resolvente
  - TCP/IP
    - lock.log 315
- Disponibilidad de puertos 113
- dónde almacenar los certificados y claves
  - privadas 296

## E

- editor de definiciones de recurso CICS
  - (CRD), Gestor de despliegue de
    - aplicaciones 253
- Ejecución remota, utilizar 102
- Ejecutar varias instancias 273

- ejemplo, procedimientos ELAXF\*
  - ELAXFADT 25
  - ELAXFASM 25
  - ELAXFBMS 25
  - ELAXFCOC 25
  - ELAXFCOP 25
  - ELAXFCOT 25
  - ELAXFCP1 25
  - ELAXFCPC 25
  - ELAXFCPP 25
  - ELAXFDCL 25
  - ELAXFGO 25
  - ELAXFLNK 25
  - ELAXFMFS 25
  - ELAXFPL1 25
  - ELAXFPLP 25
  - ELAXFPLT 25
  - ELAXFTSO 25
  - ELAXFUOP 25
- ELAXF\*, lista de comprobación de
  - calificadores 26
- ELAXF\*, procedimientos de construcción
  - remota 25
- ELAXMJCL, cambios de DB2 90
- ELAXMSAM 90
- Eliminar archivos antiguos de
  - WORKAREA 88
- Emulador de conexión de host 157
- entorno TSO, Personalizar el 265
- entorno UNIX, Orden de búsqueda
  - utilizado en 311
- entorno z/OS UNIX, Orden de búsqueda
  - utilizado en 311
- espacio de direcciones, Tamaño 153
- espacio de disco, máquinas virtuales Java
  - (JVM) 250
- espacios, diagrama de sintaxis 135
- espacios en blanco, diagrama de
  - sintaxis 135
- establecer objetivos, WLM 205
- estimado de tamaño, directrices 226
- estructura de directorios, z/OS UNIX
  - representación gráfica 200
- Estructura de directorios de z/OS UNIX
  - representación gráfica 200
- ETC.SERVICES
  - alias 318
  - port\_number 318
  - protocolo 318
  - service\_name 318
- exec de asignación (JCL de transacción
  - APPC), Utilizar 269
- exec de asignación, utilizar 267
- Explorador de sistemas remotos 10
- externa, Comunicación 162

## F

- fa.log 138
- FEJJCENFG 163, 243, 275
  - CONSOLE\_NAME 168
  - MAX\_THREADS 237
- FEJJCENFG, archivo de configuración del
  - supervisor de trabajos JES 27
- FEJJCENFG, supervisor de trabajos
  - JES 176

- FEJJCJCL, cambios de PROCLIB,
  - supervisor de trabajos JES 21
- FEJTSO 27
- FEKAPPC, Implementación de
  - APPC 106
- FEKAPPCL, Implementación de
  - APPC 106
- FEKAPPCX, Implementación de
  - APPC 106
- FEKAPPL 160
- fekfivpi.log 138
- fekfivpi.log, anotaciones de prueba
  - IVP 143
- fekfivps.log, anotaciones de prueba
  - IVP 143
- FEKFRSRV 107
- FEKLOCKD 23
- FEKLOGS, anotar y configurar el análisis
  - mediante 137
- FEKRACE, definiciones de seguridad 26,
  - 177
- fekrivp 150
- FEKRSED 22
- FEKSETUP 15, 75, 82, 106
- ffs.log 138
- ffsget.log 138
- ffsput.log 138
- File Manager
  - correquisitos 342
- Flujo de conexión 196
  - representación gráfica 196
- Flujo de daemon de bloqueo
  - representación gráfica 198
- FMID HCMA710 292
- FMID HHOP710 291
- FMID HHOP750 288
- FMID HHOP760 282
- FMIEXT.properties
  - fmlistenport 101
  - habilitado 101
- Fragmentos de sintaxis 135

## G

- Gestión de cargas de trabajo 247
- Gestor de acceso a repositorios 63
- Gestor de acceso a repositorios, activar el
  - SCM de CA Endevor® 63
- gestor de carga de trabajo 203
- Gestor de carga de trabajo, cambios 89
- Gestor de configuraciones de
  - software 63
- Gestor de despliegue de aplicaciones
  - (ADM) 191
- Gestor de despliegue de aplicaciones,
  - editor de definiciones de recurso
    - CICS 253
- Gestor de despliegue de aplicaciones,
  - personalización 73
- Gestor de despliegue de aplicaciones,
  - personalizar 253
- Gestor de despliegue de aplicaciones,
  - servidor de definiciones de recurso
    - CICS 253
- grupos de propiedades, basadas en
  - host 98

grupos de propiedades basadas en host 98  
 gskkyman, Crear una base de datos de claves con 303

## H

habilitar el compartimiento de clases, máquinas virtuales Java (JVM) 249  
 hacer copia de seguridad de archivos configurados  
     Versión 7.0 279  
     Versión 7.1 279  
     Versión 7.5 279  
 HCMA710 292  
 HHOP710 291  
 HHOP750 288  
 HHOP760 282  
 host  
     prerrequisitos 333  
 host z/OS  
     prerrequisitos 333  
 hosts locales, Tablas de 312

## I

IBM Debug Tool, correquisito 20  
 ID de usuario, consideraciones 9  
 ID de usuario y contraseña 160  
 ID de usuario y contraseña para una sola vez 160  
 idéntico nivel de software, Archivos de configuración diferentes con 274  
 IDs de transacción de servidor CRD, personalizar 76  
 IEASYSxx 243  
     MAXUSER 156, 237  
 IMS  
     correquisitos 340  
 INETD, configurar 317  
 INETD, requisitos de Developer for System z 324  
 inetd.conf 317  
     idusuario 317  
     protocolo 317  
     server\_program 317  
     server\_program\_arguments 317  
     service\_name 317  
     socket\_type 317  
     wait\_flag 317  
 información de configuración, orden de búsqueda de 310  
 información de retorno de errores, Rastreo 147  
 inicialización, IVP 112  
 inicialización de IVP  
     ivpinit 112  
 Iniciar el daemon de bloqueo, mandatos 126  
 Iniciar el daemon RSE, mandatos 126  
 Iniciar el supervisor de mandatos JES, mandatos 125  
 inicio, Requisitos de JCL 153  
 inicio, z/OS UNIX INETD 321  
 Inicio alternativo del servidor CARMA 60  
 inicio de CA Endeavor® SCM RAM, CRASTART 67  
 inicio de CA Endeavor® SCM RAM, sometimiento por lotes 65  
 inicio de CARMA, sometimiento por lotes 55  
 instalación de SMP/E, sticky bit 151  
 Instalar y configurar los componentes del host 3  
 integración, Gestor de archivos 100  
 Integración del analizador de errores, Anotaciones 141  
 Integración del gestor de archivos 100  
 Integración del gestor de archivos, Anotaciones 141  
 interfaz, RESTful frente a servicio Web 75  
 interfaz de servicio Web 254  
     ADMI 77  
     ADMR 77  
     ADMS 77  
 Interfaz de servicio Web, servidor CRD utilizando 77  
 interfaz de servicio Web frente a interfaz RESTful 75  
 interfaz RESTful 254  
     ADMI 76  
     ADMR 76  
     ADMS 76  
 interfaz RESTful, servidor CRD utilizando 75  
 interfaz RESTful frente a interfaz de servicio Web 75, 254  
 Interfaz RSE con CARMA 53  
 interna, Comunicación 163  
 IRXJCL, IRXJCL versus CRAXJCL 71  
 ISPSISLOAD  
     pasarela de cliente TSO/ISPF de ISP 186  
 ISPF, Utilizar varios execs de asignación 267  
 ISPF.conf 47  
     allocjob 48  
     ISPF\_timeout 48  
     isplib 48  
     ispmlib 48  
     ispplib 48  
     ispslib 48  
     isptlib 48  
     sysproc 48  
 ISPF.conf, actualizaciones para SCLMDT 82  
 ISPF.conf, Ajustar 61  
 ISPF.conf, Personalización básica 266  
 IVP  
     fekfivpa 118  
     fekfivpd 115  
     fekfivpi 116  
     fekfivpj 115  
     fekfivpl 116  
     fekfivpr 120  
     fekfivps 118  
     fekfivpz 121  
 IVP, servicios básicos y opcionales  
     fekfivpa 111  
     fekfivpd 111  
     fekfivpi 111

IVP, servicios básicos y opcionales (continuación)

    fekfivpj 111  
     fekfivpl 111  
     fekfivpr 111  
     fekfivps 111  
     fekfivpt 111  
     fekfivpz 111  
     scripts de IVP 111  
 IVP fekfivpi, Anotaciones de prueba 143  
 IVTPRMxx  
     ECSA MAX 237  
     FIXED MAX 237

## J

J2EE 81, 86  
 Java 81, 86  
 Java, vuelcos 144  
 Java 2 Technology Edition  
     correquisitos 335  
 JAVA\_DUMP\_TDUMP\_PATTERN 144  
 Java Virtual Machines (JVM), compartimiento de clases entre 248  
 JCL de inicio, Requisitos 153  
 JCL de transacción APPC, Personalización básica 268  
 JCL de transacción APPC, Utilizar perfiles ISPF existentes con 269  
 JCL de transacción APPC, Utilizar un exec de asignación 269  
 JES, Definir la seguridad de mandatos 184  
 JES, Seguridad 166  
 JES, servidor supervisor de trabajos 21  
 JES JMON  
     GEN\_CONSOLE\_NAME 168  
 JES JMON, FEJCNFG  
     \_BPXK\_SETIBMOPT  
     \_TRANSPORT 28  
 APPLID 28  
 AUTHMETHOD 28  
 CODEPAGE 28  
 CONCHAR 28  
 CONSOLE\_NAME 28  
 GEN\_CONSOLE\_NAME 28  
 HOST\_CODEPAGE 28  
 LIMIT\_COMMANDS 28  
 LIMIT\_VIEW 28  
 LISTEN\_QUEUE\_LENGTH 28  
 MAX\_DATASETS 28  
 MAX\_THREADS 28  
 SERV\_PORT 28  
 SUBMITMETHOD 28  
 TIMEOUT 28  
 TIMEOUT\_INTERVAL 28  
 TSO\_TEMPLATE 28  
 TZ 28  
 JMON 21, 184, 275  
     fekfivpj 115  
 JMON, supervisor de trabajos JES 109  
 JMON, tarea iniciada del Supervisor de trabajos JES 21  
 JVM, compartimentode clases entre 248

## K

keytool, Crear un almacén de claves con 306

## L

Language Environment, bibliotecas de tiempo de ejecución 246  
leer un diagrama de sintaxis, Cómo 134  
liberar un bloqueo  
RSE, mandato de modificar cancelación 199  
LIMIT\_COMMANDS 167  
LIMIT\_VIEW 169  
Limitaciones de ejecución, Acciones en trabajos 168  
limitar la comunicación externa, puertos especificados 161  
límite de tamaño, almacenamiento dinámico Java 224  
límite de tamaño, espacio de direcciones 225  
límite de tamaño de almacenamiento dinámico, Java 224  
Límite de tamaño de almacenamiento dinámico Java 224  
límite de tamaño de espacio de direcciones 225  
límites de tamaño de memoria caché, máquinas virtuales Java (JVM) 249  
límites de z/OS UNIX establecidos en BPXPRMxx 16  
Límites del sistema 156  
limpieza de directorios, WORKAREA skulker 108  
limpieza de directorios de WORKAREA skulker 108  
LINKLIST, definiciones para otros productos 21  
Lista de certificados revocados (CRL), consulta  
rsed.envvars 172  
variables de entorno CRL 172  
lista de comprobación, administrador SCLM 87  
lista de comprobación, requisitos 15  
Lista de comprobación de calificadores de alto nivel de ELAXF\* 26  
lista de comprobación de requisitos 15  
Lista de comprobación del administrador de SCLM 87  
Lista de comprobación del cliente 13  
Lista de comprobación para la transacción APPC 105  
locales, Tablas de hosts 312  
lock.log 138  
LOCKD 10  
LOCKD, daemon de bloqueo 109  
Los conjuntos de datos de MVS no pueden abrirse 156  
LPALSTxx 246  
LPALSTxx, definiciones LPA en 18  
LSTRANS.FILE, crear 84

## M

Mandato Detener (P) 131  
Mandato Iniciar (S) 125  
Mandato Modificar (F) 127  
mandatos, Detener (P) 131  
mandatos, Iniciar (S) 125  
mandatos, Modificar (F) 127  
Mandatos de operador 125  
mandatos ISPF 48, 61  
mandatos JES, definir la seguridad 184  
mandatos TSO 61  
mandatos TSO (APPC), conexión del servicio 118  
Manejador, mensajes de conducto 77  
Manejador de mensajes de conducto 77  
manifiesto, repositorio 79  
Mejorar el acceso a las bibliotecas del sistema 245  
Mejorar el rendimiento de la comprobación de seguridad 247  
mensajes, consola 132  
mensajes, programa de utilidad administrativo 262  
mensajes de consola 132  
Supervisor de trabajos JES 132  
Mensajes de consola  
daemon de bloqueo 132  
daemon RSE 132  
Servidor de agrupaciones de hebras RSE 132  
mensajes de error, IRZ Diagnóstico 93  
mensajes del programa de utilidad administrativo 262  
método de acceso APPC, Utilizar el 268  
método de acceso de Pasarela de cliente TSO/ISPF, Utilizar el 266  
método de conexión RSE, alternativo 103  
método de conexión RSE alternativo 103  
métodos, Autenticación 159  
Métodos de acceso, TSO 265  
Métodos de autenticación 159  
migración 279  
migración, versión 7.5 a 7.6 282  
Modificar el daemon de bloqueo, mandatos 131  
Modificar el daemon RSE, mandatos 128  
Modificar el supervisor de mandatos JES, mandatos 127  
MVS, vuelcos 143  
MVS para RSE, Definir bibliotecas controladas por programa 186

## N

necesarios, recursos 5  
netstat 151  
netstat, configuración de TCP/IP 114  
nivel de software, idéntico en archivos de configuración diferentes 274  
niveles de software idénticos con archivos de configuración diferentes 274  
no administradores del sistema, actualizar privilegios 201  
no alfanuméricos, Caracteres en diagrama de sintaxis 135

nombre de host, Dependencia del 309  
nombre de LU asociada alternativo, servicio de mandatos TSO 330  
nombre de transacción, alternativo para FEKFRSRV para el servicio de mandatos TSO 330  
nombres de host, aplicar a Developer for System z 313  
notas de migración, programa de utilidad administrativo 261

## O

objetivos, establecer en WLM 205  
OMVS, Definir segmento 180  
opción Java Xquickstart 248  
opcionales, Componentes 48  
opcionales. tareas de personalización 89  
opciones de configuración alternativas, APPC y VTAM 330  
opciones de configuración alternativas de APPC 330  
opciones de configuración alternativas de VTAM 330  
operador, Mandatos 125  
operando, Seleccionar más de uno en un diagrama de sintaxis 135  
Operandos, diagrama de sintaxis 134  
orden de búsqueda, entorno MVS nativo 319  
orden de búsqueda, entorno z/OS UNIX 319  
Orden de búsqueda, entorno z/OS UNIX 311  
orden de búsqueda de la información de configuración 310

## P

parámetros adicionales de inicio con \_RSE\_CMDSERV\_OPTS, Definir Java 47  
parámetros adicionales de inicio con \_RSE\_JAVAOPTS, Definir Java 42  
parámetros de inicio Java con \_RSE\_CMDSERV\_OPTS, Definir 47  
parámetros de inicio Java con \_RSE\_JAVAOPTS, Definir 42  
PARMLIB, cambios 16  
Pasarela de cliente TSO/ISPF, Utilizar el método de acceso de 266  
pasarela de cliente TSO/ISPF de ISP ISP.SISPLOAD 186  
PassTicket para RSE, Definir el soporte 187  
PassTickets, utilizar 164  
perfil de seguridad, Limitaciones almacenadas en 153  
perfiles de conjunto de datos, Definir 180  
perfiles ISPF (JCL de transacción APPC) 269  
perfiles ISPF existentes (JCL de transacción APPC), Utilizar 269  
perfiles ISPF existentes, Utilizar 266  
personalización - ISPF.conf, 266

personalización, configuración de 15  
 personalización, SCLM Developer Toolkit 81  
 personalización, tareas opcionales 89  
 personalización básica 15  
 Personalizar el entorno TSO 265  
 personalizar el Gestor de despliegue de aplicaciones 253  
 personalizar los ID de transacción de servidor CRD 76  
 perspectiva de proyectos z/OS 99  
 PL/I Compiler  
   correquisitos 338  
 plataformas soportadas 3  
 Ported tools  
   correquisitos 343  
 PORTRANGE 41, 152  
 PORTRANGE disponible para RSE, Definir 41  
 preconfiguración, consideraciones relativas 9  
 predespliegue, consideraciones relativas 12  
 preinstalación, consideraciones relativas 4  
 prerrequisito, definiciones LINKLIST 20  
 prerrequisito, definiciones LPA 20  
 prerrequisito, productos 5  
 prerrequisito LINKLIST y LPA, definiciones 20  
 prerrequisitos  
   host 333  
   SMP/E 334  
 prerrequisitos, Developer for System z 333  
 prerrequisitos de host 333  
 Prerrequisitos del host z/OS 333  
 Probar la conexión de configuración de host SSL 300  
 problemas de los requisitos, conocidos 156  
 procedimiento almacenado DB2 89  
 procedimientos de construcción remota ELAXF\* 25  
 PROCLIB, cambios 21, 90  
 productos prerrequisito 5  
 productos y software, configuración de requisitos 9  
 productos y software necesarios, configuración de 9  
 Programa de utilidad administrativo, notas de migración 261  
 programa de utilidad administrativo de CICS 74  
 programa de utilidad administrativo para administradores CICS  
   funciones proporcionadas 257  
 programas de utilidad de gestión de memoria caché, máquinas virtuales Java (JVM) 250  
 programas de verificación de instalación 111  
 PROGxx, autorizaciones APF 18  
 PROGxx, definiciones LINKLIST 19  
 projectcfg.properties 99  
 PROJECT-HOME 100  
 WSED-VERSION 100

propertiescfg.properties 98  
   DEFAULT-VALUES 99  
   ENABLED 99  
   PROPERTY-GROUP 99  
   RDZ-VERSION 99  
 propietarios de tareas 194  
 protección de aplicaciones para RSE, Definir 187  
 proyectos, basados en host 99  
 proyectos basados en host 99  
 prueba IVP de fekfivpi, Anotaciones 143  
 Publicaciones a las que se hace referencia 345  
 puertos, CARMA y TCP/IP 164  
 Puertos, disponibilidad 113  
 puertos, limitar la comunicación externa a específicos 161  
 puertos, REXEC 104  
 puertos CARMA y TCP/IP 164  
 Puertos TCP/IP 162  
 puertos TCP/IP, representación gráfica 162  
 Puertos TCP/IP reservados 151

## Q

qué es Developer for System z 191  
 quickstart, opción Java (-Xquickstart) 248

## R

RACF  
   permisos 181  
 RACF, Crear un anillo de claves con 297  
 RAM (Repository Access Manager), Activar esqueleto 63  
 RAM (Repository Access Manager), Activar PDS 63  
 RAM (Repository Access Manager), Activar SCLM 63  
 RAM (Repository Access Manager) de esqueleto, Activar 63  
 RAM (Repository Access Manager) de PDS, Activar 63  
 RAM (Repository Access Manager) de SCLM, Activar 63  
 RAM (Repository Access Managers) de ejemplo, Activar 62  
 rastreo 146  
 rastreo, CARMA 147  
 rastreo, daemon de bloqueo 147  
 rastreo, RSE 96, 146  
 rastreo de CARMA 147  
 rastreo de daemon de bloqueo 147  
 Rastreo de información de retorno de errores 147  
 Rastreo del supervisor de trabajos JES 146  
 rastreo RSE 146  
 Rational Team Concert para System z  
   correquisitos 341  
 Recuento de espacios de direcciones 215  
 Recuento de hebras 221  
 Recuento de procesos 218

recursos CICS, anotaciones de instalación 255  
 recursos necesarios 5  
 red, supervisar 241  
 referenciadas, publicaciones 345  
 Región de conexión primaria CICS 75, 78  
 Región de conexión primaria no-CICS 78  
 regiones de conexión, primarias frente a no primarias 254  
 Regiones de conexión no primarias CICS 76  
 regiones de conexión primarias frente a no primarias 254  
 reglas de clasificación, WLM 204  
 reglas de clasificación WLM 204  
 reigón propietaria Web 254  
 rendimiento, Consideraciones relativas 245  
 rendimiento de la comprobación de seguridad, Mejorar el 247  
 Repositorio CRD 74, 175  
 repositorio de manifiesto 79  
 requisitos, Developer for System z 333  
 requisitos de acceso 9  
 Requisitos de JCL de inicio 153  
 reservados, Puertos TCP/IP 151  
 Resolución de problemas de configuración 137  
 resolvente, Definiciones locales disponibles para 315  
 resolvente TCP/IP, dirección de host no resuelta  
   lock.log 315  
 resolventes, Qué son 310  
 resolventes base, Archivos de configuración 311  
 retorno de errores, Rastreo de información 147  
 REXEC, conexión 120  
 REXEC, utilizar 102  
 REXEC/SSH, script de shell 121  
 REXX  
   correquisitos 342  
 rmt\_class\_loader\_cache.jar 138  
 RSE 10  
 RSE, Anotaciones de usuario 140  
 RSE, Definir archivos controlados por programa z/OS UNIX para 188  
 RSE, Definir bibliotecas controladas por programa MVS para 186  
 RSE, Definir como servidor z/OS UNIX seguro 186  
 RSE, Definir comprobación de puerto de entrada para 174  
 RSE, Definir el rango de puertos (PORTRANGE) 41  
 RSE, Definir el soporte de PassTicket para 187  
 RSE, definir protección de aplicaciones para 187  
 RSE, rsed.envvars  
   \_RSE\_JAVAOPTS 176  
 RSE, ssl.properties 177  
 RSE, supervisar 238

- RSE como aplicación Java
  - representación gráfica 193
- RSE existente, Clonar la
  - configuración 299
- rsecomm.log 138
  - Anotaciones de Integración del gestor de archivos 141
  - anotaciones de SCLM Developer Toolkit 142
- rsecomm.properties 146
  - debug\_level 96
  - log\_location 96
  - server.version 96
- rsecomm.properties, 96
- RSED 22
- RSED, daemon RSE 109
- rsed.envvars 128, 232, 274
  - \_BPX\_SHAREAS 40
  - \_BPX\_SPAWN\_SCRIPT 40
  - \_BPXK\_SETIBMOPT
  - \_TRANSPORT 38
  - \_CEE\_DMPTARG 36
  - \_CEE\_RUNOPTS 40
  - \_CMDSERV\_BASE\_HOME 37
  - \_CMDSERV\_CONF\_HOME 37, 268
  - \_CMDSERV\_WORK\_HOME 37
  - \_FEKFSCMD\_PARTNER\_LU\_ 38
  - \_FEKFSCMD\_TP\_NAME\_ 38
  - &ISPROF=&SYSUID..ISPPROF= 47
  - \_RSE\_CMDSERV\_OPTS 40
  - \_RSE\_DAEMON\_CLASS 40
  - \_RSE\_HOST\_CODEPAGE 36
  - \_RSE\_JAVAOPTS 40, 144, 265
  - \_RSE\_LOCKD\_CLASS 40
  - \_RSE\_LOCKD\_PORT 36
  - \_RSE\_POOL\_SERVER\_CLASS 40
  - \_RSE\_PORTRANGE 38, 161
  - \_RSE\_SERVER\_CLASS 40
  - \_RSE\_SERVER\_TIMEOUT 40
  - \_SCLMDT\_TRANTABLE 38
  - ANT\_HOME 38
  - CGI\_DTWORk 40
  - CLASSPATH 40
  - DAPPLID 46
  - Daudit.cycle 44
  - Daudit.retention.period 44
  - Ddaemon.log 42
  - Ddeny.nonzero.port 45
  - DDENY\_PASSWORD 46
  - DDSTORE\_IDLE\_SHUTDOWN
  - \_TIMEOUT 46
  - DDSTORE\_LOG\_DIRECTORY 42
  - DDSTORE\_MEMLOGGING\_ON 46
  - DDSTORE\_TRACING\_ON 46
  - Denable.audit.log 44
  - Denable.automount 44
  - Denable.certificate.mapping 44
  - Denable.port.of.entry 44
  - Denable.standard.log 44
  - DHIDE\_ZOS\_UNIX 46
  - Dipv6 44
  - Dkeep.last.log 44
  - Dmaximum.clients 42, 233
  - Dmaximum.threadpool.process 44, 233
  - Dmaximum.threads 42, 233

- rsed.envvars (continuación)
  - Dminimum.threadpool.process 42, 233
  - Dprocess.cleanup.interval 45
  - Dsingle.logon 45
  - DSTORE\_LOG\_DIRECTORY 142, 146
  - DTSO\_SERVER 46
  - Duser.log 42
  - GSK\_CRL\_SECURITY\_LEVEL 38
  - GSK\_LDAP\_PASSWORD 38
  - GSK\_LDAP\_PORT 38
  - GSK\_LDAP\_SERVER 38
  - GSK\_LDAP\_USER 38
  - JAVA\_HOME 36
  - JAVA\_PROPAGATE 40
  - LANG 36
  - LIBPATH 40
  - PATH 36, 40
  - RSE\_HOME 36
  - RSE\_JAVAOPTS 37
  - RSE\_LIB 40
  - RSE\_SAF\_CLASS 37
  - SCLMDT\_BASE\_HOME 40
  - SCLMDT\_CONF\_HOME 38
  - SCLMDT\_WORK\_HOME 40
  - STEPLIB 36, 37, 38, 170
  - TZ 36
  - Xms 42, 233
  - Xmx 42, 233
- rsed.envvars, actualizaciones para la
  - conversión de nombres
  - largos/abreviados 86
- rsed.envvars, actualizaciones para
  - SCLMDT 83
- rsed.envvars, actualizar para habilitar la
  - coexistencia 299
- rsed.envvars, archivo de configuración de
  - RSE 32
- rsedaemon.log 138, 139
- rseserver.log 138, 139

## S

- salidas del sistema, Limitaciones
  - aplicadas por 153
- SCLM 86
- SCLM, Conversión de nombres
  - largos/abreviados 84
- SCLM Developer Toolkit 186
- SCLM Developer Toolkit,
  - personalización 81
- SCLM Developer Toolkit (SCLMDT) 191
- SCLMDT, actualizaciones de ISPF.conf
  - para 82
- SCLMDT, actualizaciones de rsed.envvars
  - para 83
- SCLMDT admin 87
- Script de shell REXEC/SSH 121
- script de shell SSH 121
- script shell ivpinit, establecer variables de
  - entorno RSE con 112
- SDSF 82
- Secure Shell, utilizar 102
- segmento OMVS, Definir 180
- seguridad, Activar valores y clases 179
- seguridad, CICSTS 175
- seguridad, conducto 255

- seguridad, Consideraciones relativas
  - a 159
- Seguridad, definiciones 26, 177
- seguridad, Gestor de despliegue de
  - aplicaciones (ADM) 255
- seguridad, INETD 323
- seguridad, recurso 257
- seguridad, SCLM 175
- seguridad, transacción 255
- seguridad de CICSTS 175
- seguridad de conducto 255
- Seguridad de conexión 161
- seguridad de hebras en el servidor RSE
  - PassTickets 164
- seguridad de INETD 323
- Seguridad de JES 166
- seguridad de LU, VTAM 330
- seguridad de mandatos JES, Definir 184
- seguridad de memoria caché, máquinas
  - virtuales Java (JVM) 249
- seguridad de recursos 257
- seguridad de SCLM 175
- seguridad de transacciones 255
- seguridad del Gestor de despliegue de
  - aplicaciones 255
- seguridad del repositorio, CRD 255
- seguridad del repositorio CRD 255
- serverlogs.count 138
- Servicio de mandatos TSO 154, 191, 265
- servicio de mandatos TSO, transacción
  - APPC para 104
- servicio SCLM Developer Toolkit 19
- servidor, consideraciones 10
- servidor CARMA, Inicio alternativo 52, 57, 60
- Servidor CRD utilizando la interfaz
  - RESTful 75
- Servidor de agrupaciones de hebras RSE
  - mensajes de consola 132
- servidor de definiciones de recurso CICS (CRD), Gestor de despliegue de
  - aplicaciones 253
- servidor RSE 162
- servidor supervisor de trabajos JES 21
- servidor UNIX, Definir RSE como 186
- servidor z/OS seguro, Definir RSE
  - como 186
- servidor z/OS UNIX, Definir RSE
  - como 186
- sesión de shell, inicio de INETD 322
- Símbolos, diagrama de sintaxis 134
- Sintaxis, ejemplo 135
- sintaxis, Fragmentos de 135
- sistema, Límites 156
- sistema, Mejorar el acceso a las
  - bibliotecas del 245
- sistemas de archivos z/OS UNIX,
  - supervisar 241
- sistemas de archivos zFS, utilizar 245
- sistemas operativos soportados 3
- skulker, limpieza de directorios de
  - WORKAREA 108
- SMP/E
  - prerrequisitos 334
- software, configuración de productos
  - obligatorios y 9

software de seguridad, autenticación por 172  
sometimiento por lotes, inicio de CA Endeavor® SCM RAM mediante 65  
sometimiento por lotes, inicio del servidor CARMA mediante 55  
soporte, Enterprise Service Tools 91  
soporte, EST 91  
soporte de autorización de cliente, añadir X.509 303  
Soporte de Enterprise Service Tools 91  
soporte de EST 91  
Soporte de idiomas bidireccionales 92  
Soporte de idiomas bidireccionales CICS 92  
soporte de PassTicket para RSE, Definir 187  
SSH, utilizar 102  
SSL, Cifrado de comunicaciones mediante 161  
SSL, Configurar 295  
ssl.properties 93  
    daemon\_key\_label 95  
    daemon\_keydb\_file 95  
    daemon\_keydb\_password 95  
    enable\_ssl 95  
    server\_keystore\_file 95  
    server\_keystore\_label 95  
    server\_keystore\_password 95  
    server\_keystore\_type 95  
ssl.properties, Activar SSL actualizando 299  
ssl.properties, activar SSL creando un daemon RSE nuevo 300  
stderr\*.log 138  
stderr.log 138  
stdout\*.log 138  
stdout.log 138  
STEPLIB, Evitar el uso de 245  
sticky bit, disponibilidad de módulo de carga MVS para z/OS UNIX 151  
subproyectos z/OS UNIX, acciones basadas en host remoto 103  
subsistemas soportados 3  
supervisar, red 241  
supervisar RSE 238  
supervisar sistemas de archivos z/OS UNIX 241  
supervisar z/OS UNIX 239  
Supervisor de trabajos JES, Anotaciones 139  
Supervisor de trabajos JES, autenticación 160  
Supervisor de trabajos JES, conexión 115  
supervisor de trabajos JES, FEJCNFG 176  
supervisor de trabajos JES, JMON 109  
supervisor de trabajos JES (JMON) 191  
Supervisor de trabajos JES, mandato Iniciar 125  
Supervisor de trabajos JES, mandato Modificar 127  
Supervisor de trabajos JES, rastreo 146  
SYS1.PARMLIB(APPCPMxx) 327  
SYS1.PARMLIB(ASCHPMxx) 328  
SYS1.PARMLIB(BPXPRMxx) 243  
    MAXASSIZE 153, 183

SYS1.PARMLIB(BPXPRMxx) (continuación)  
    MAXPROCSYS 156  
    MAXPROCUSER 156  
    MAXUIDS 156  
SYS1.PARMLIB(BPXPRMxx), Limitaciones establecidas en 153  
SYS1.PARMLIB(BPXPRMxx), máquinas virtuales Java (JVM) 249  
SYS1.PARMLIB(IEASYSxx) 243  
    MAXUSER 156  
SYSEXEC 48  
sysplex, configuración idéntica en todo 273  
SYSPROC 48

## T

Tablas de conversión 312  
Tablas de hosts locales 312  
Tamaño del espacio de direcciones 153  
tamaño fijo, Almacenamiento dinámico Java de 247  
Tarea iniciada RSE, RSED 22  
tareas de personalización opcionales 89  
tareas iniciadas, Definir para Developer for System z  
    tareas iniciadas JMON 183  
    tareas iniciadas LOCKD 183  
    tareas iniciadas RSED 183  
tareas iniciadas, verificar 109  
tareas iniciadas, verificar servicios 111  
tareas iniciadas de Developer for System z, Definir 183  
TCP/IP, aplicar a Developer for System z 313  
TCP/IP, Configurar 309  
TCP/IP, Definiciones locales disponibles para el resolvente 315  
TCP/IP, Puertos 162  
TCP/IP, Puertos reservados 151  
tiempo de ejecución de Language Environment, bibliotecas de 246  
tipos de subsistema  
    ASCH 204  
    CICS 204  
    JES 204  
    OMVS 204  
    STC 204  
TN3270 338  
trabajos, Acciones condicionales en 166  
Transacción APPC  
    resolución de problemas de APPC 154  
transacción APPC, Anotaciones 142  
transacción APPC, implementación 106  
transacción APPC, lista de comprobación 105  
transacción APPC, preparación 104  
transacción APPC para el servicio de mandatos TSO 104  
Transacción APPC y el servicio de mandatos TSO 154  
transacción del servicio de mandatos TSO, Definir 329

transacciones APPC, configurar con varias configuraciones de Developer for System z 270  
transacciones CICS 175  
TSO, Métodos de acceso 265  
TSO/ISPF, archivo de configuración de la Pasarela de cliente 47  
TSO/ISPF, personalización - ISPF.conf, 266  
TSO/ISPF, utilizar con varias configuraciones 267  
TSO/ISPF, Utilizar el método de acceso de Pasarela de cliente 266  
TSO/ISPF, Utilizar perfiles ISPF existentes 266  
TSO/ISPF, Utilizar un exec de asignación 267  
TSO/ISPF, Utilizar varios execs de asignación 267

## U

Ubicaciones de vuelcos de UNIX 145  
Ubicaciones de vuelcos de z/OS UNIX 145  
uchars.settings, Caracteres no editables 101  
uso de almacenamiento 224  
uso de espacio, sistema de archivos z/OS UNIX 230  
uso de espacio del sistema de archivos, z/OS UNIX 230  
uso de espacio del sistema de archivos de z/OS UNIX 230  
uso de STEPLIB 20  
uso de STEPLIB, Evitar 245  
usuario de RSE, Anotaciones 140  
utilización de recursos, ajuste 213  
utilización de recursos, visión general 214  
utilizar el sometimiento por lotes para el inicio del servidor CARMA 55  
utilizar PassTickets 164  
utilizar perfiles ISPF existentes 266  
Utilizar un exec asignación 267

## V

validación de la autoridad certificadora conjunto de claves SAF 171  
gskkyman 171  
TRUST, HIGHTRUST 171  
valores de entorno, INETD 324  
valores de entorno INETD 324  
valores de seguridad, verificar 189  
valores y clases de seguridad, Activar 179  
variable PARM, limitaciones de JCL 24  
variables de patrón de volcado de transacciones 144  
varias configuraciones de Developer for System z, utilizar varias transacciones APPC con 270  
varias configuraciones de Developer for System z, utilizar varios archivos ISPF.conf con 267

- varias instancias, Ejecutar 273
- varias LU 330
- Varias transacciones APPC 270
- Varios archivos ISPF.conf 267
- varios execs de asignación,
  - TSO/ISPF 267
- verificación de la instalación 109
- verificar la conexión de Pasarela de cliente TSO/ISPF de ISPF 116
- verificar la conexión de Pasarela de cliente TSO/ISPF de ISPF, verificar 116
- verificar script de shell REXEC/SSH 121
- Verificar valores de seguridad 189
- versión 7.0 y versión 7.1, Cambios entre 291
- visión general de componentes,
  - Developer for System z
    - representación gráfica 191
- VSAM 325
- VTAM 326
- VTAM (método de acceso para telecomunicaciones virtuales) 326
- Vuelcos de Java 144
- Vuelcos de MVS 143

## W

- WORKAREA, Eliminar archivos antiguos de 88

## X

- X.509, añadir soporte de autorización de cliente 303
- X.509, certificado 160
- Xquickstart, opción Java 248

## Z

- z/OS
  - correquisitos 336
- z/OS UNIX, establecer límites en BPXPRMxx 16
- z/OS UNIX, supervisar 239
- z/OS UNIX, ubicaciones de vuelcos 145
- zFS, sistemas de archivos 245

---

# Hoja de Comentarios

IBM Rational Developer for System z  
Guía de configuración de host  
Versión 7.6.1

Número de Publicación SC11-3660-04

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)

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

SC11-3660-04

