

IBM Rational Developer für System z



# Hostkonfiguration

*Version 7.1.1*



IBM Rational Developer für System z



# Hostkonfiguration

*Version 7.1.1*

#### Hinweis

Vor Verwendung dieser Informationen sollten die allgemeinen Hinweise im Abschnitt „Bemerkungen“ auf Seite 133 gelesen werden.

#### **Zweite Ausgabe(Dezember 2007)**

Diese Ausgabe bezieht sich auf IBM Rational Developer für System z Version 7.1.1 (Programmnummer 5724-T07) und, sofern in neuen Ausgaben nichts anderes angegeben ist, auf alle folgenden Releases und Modifikationen.

Veröffentlichungen können über den zuständigen IBM Ansprechpartner oder die zuständige IBM Geschäftsstelle bezogen werden. Veröffentlichungen sind nicht bei der unten angegebenen Adresse erhältlich.

Am Ende dieser Veröffentlichung befindet sich ein Vordruck für ein Antwortschreiben.

Werden an IBM Informationen eingesandt, können diese beliebig verwendet werden, ohne dass eine Verpflichtung gegenüber dem Einsender entsteht.

Diese Veröffentlichung ist eine Übersetzung des Handbuchs  
*IBM Rational Developer for System z Version 7.1.1 Host Configuration*,  
IBM Form SC23-7658-01,  
herausgegeben von International Business Machines Corporation, USA

© Copyright International Business Machines Corporation 2007  
© Copyright IBM Deutschland GmbH 2007

Informationen, die nur für bestimmte Länder Gültigkeit haben und für Deutschland, Österreich und die Schweiz nicht zutreffen, wurden in dieser Veröffentlichung im Originaltext übernommen.

Möglicherweise sind nicht alle in dieser Übersetzung aufgeführten Produkte in Deutschland angekündigt und verfügbar; vor Entscheidungen empfiehlt sich der Kontakt mit der zuständigen IBM Geschäftsstelle.

Änderung des Textes bleibt vorbehalten.

Herausgegeben von:  
SW TSC Germany  
Kst. 2877  
Dezember 2007

---

# Inhaltsverzeichnis

<b>Abbildungsverzeichnis</b> . . . . .	<b>vii</b>
--	------------

<b>Tabellen</b> . . . . .	<b>ix</b>
---------------------------	-----------

<b>Zu diesem Handbuch</b> . . . . .	<b>xi</b>
-------------------------------------	-----------

Zielgruppe . . . . .	xi
Referenzierte Veröffentlichungen . . . . .	xi

## Kapitel 1. Hostkomponenten installieren und konfigurieren . . . . . 1

Hinweise zur Installationsvorbereitung . . . . .	2
Hinweise zur Konfigurationsvorbereitung . . . . .	3
Erforderliche Konfiguration für vorausgesetzte Produkte und Software . . . . .	3
Hinweise zur Benutzer-ID . . . . .	4
Hinweise zum Server . . . . .	5
Erforderliche Berechtigungen für die Implementierung der Konfigurations-Tasks . . . . .	6
Hinweise zur Deployment-Vorbereitung . . . . .	6
IBM Rational Developer für System z, FMID HHOP710 . . . . .	7
IBM Common Access Repository Manager (CARMA), FMID HCMA710 . . . . .	7

## Kapitel 2. Installations- und Konfigurationsänderungen . . . . . 9

Änderungen zwischen Version 7.0 und Version 7.1 . . . . .	9
IBM Rational Developer für System z, FMID HHOP710 . . . . .	9
IBM Common Access Repository Manager (CARMA), FMID HCMA710 . . . . .	9
Änderungen zwischen Version 6.0.1 und Version 7.0 . . . . .	10
IBM WebSphere Developer für System z, FMID HHOP700 . . . . .	10
IBM Common Access Repository Manager (CARMA), FMID HCMA700 . . . . .	11
Bereits konfigurierte Dateien sichern . . . . .	11

## Kapitel 3. MVS-Komponenten von Developer für System z aktivieren . . . . . 15

MAXASSIZE in SYS1.PARMLIB(BPXPRMxx) setzen . . . . .	15
HLQ.SFEKLOAD für APF berechtigen . . . . .	15
Konfigurationsdatei für JES Job Monitor (FEJJCNF) anpassen . . . . .	16
Start-JCL für JES Job Monitor anpassen . . . . .	18
Trace-Erstellung für JES Job Monitor . . . . .	19
JES Job Monitor als STC ausführen . . . . .	19
Serverberechtigungen . . . . .	20
Start-JCL für JES Job Monitor prüfen . . . . .	20
JES-Spool-Zugriff und Sicherheit . . . . .	20
Bedingter Spool-Zugriff . . . . .	20
Verfügbare Befehle . . . . .	21
Zugriff auf Spool-Dateien einschränken . . . . .	22

ELAXF*-Prozeduren für ferne Build-Erstellung anpassen . . . . .	22
APPC-Transaktion für den TSO Commands Service definieren . . . . .	24
Vorbereitungen . . . . .	24
Implementierung . . . . .	26
ELAXM*-Member für gespeicherte DB2-Prozedur anpassen (optional) . . . . .	27
Unterstützung bidirektionaler Sprachen für CICS anpassen (optional) . . . . .	28
Application Deployment Manager (ADM) anpassen (optional) . . . . .	29
CRD-Repository . . . . .	30
Primäre CICS-Verbindungsregion . . . . .	31
Pipelinesnachrichten-Handler . . . . .	31
Nicht primäre CICS-Verbindungsregionen (optional) . . . . .	32

## Kapitel 4. z/OS-UNIX-Komponenten von Developer für System z aktivieren . . . . . 33

Konfigurationsdatei rsed.envvars in einem anderen Verzeichnis speichern . . . . .	34
RSE-Konfigurationsdatei rsed.envvars anpassen . . . . .	35
Für RSE verfügbaren PORTRANGE definieren (optional) . . . . .	40
Zusätzliche Java-Startparameter mit _RSE_*OPTS definieren (optional) . . . . .	41
INETD-Dämon und -REXEC/SSH für RSE konfigurieren . . . . .	43
INETD-RSE-Dämon konfigurieren . . . . .	43
INETD-REXEC (oder -SSH) konfigurieren . . . . .	45
ISPF-Konfigurationsdatei ISPF.conf anpassen . . . . .	46
RSE-Serverkonfiguration prüfen . . . . .	47
Port-Verfügbarkeit . . . . .	48
REXEC-Verbindung . . . . .	49
REXEC/SSH-Shell-Script . . . . .	49
RSE-Dämonverbindung . . . . .	50
JES-Job-Monitor-Verbindung . . . . .	51
TSO-Commands-Service-Verbindung (bei Verwendung von SCLMDT) . . . . .	51
TSO-Commands-Service-Verbindung (bei Verwendung von APPC) . . . . .	52
RSE-SSL-Konfiguration in ssl.properties anpassen (optional) . . . . .	53
RSE-Trace-Konfiguration in rsecomm.properties anpassen (optional) . . . . .	54
Konfiguration von Hostprojekten in projectcfg.properties anpassen (optional) . . . . .	54
File Manager Integration in FMIEXT.properties anpassen (optional) . . . . .	55

## Kapitel 5. IBM Common Access Repository Manager (CARMA) aktivieren (optional) . . . . . 57

CARMA-MVS-Komponenten anpassen . . . . .	58
--	----

CARMA-z/OS-UNIX-Komponenten anpassen . . . . .	59
Beispiel-RAM (Repository Access Manager) aktivieren (optional). . . . .	60
SCLM-RAM aktivieren . . . . .	61
PDS-RAM aktivieren . . . . .	61
Skeleton-RAM aktivieren . . . . .	62

## **Kapitel 6. IBM Software Configuration and Library Manager (SCLM) Developer Toolkit aktivieren (optional). . . . . 63**

## **Kapitel 7. Hinweise zur Clientkomponente von Developer für System z . . . 65**

## **Kapitel 8. Leistungsaspekte. . . . . 67**

Dateisystem zFS verwenden . . . . .	67
Verwendung von STEPLIB vermeiden . . . . .	67
Zugriff auf Systembibliotheken verbessern . . . . .	68
LE-Laufzeitbibliotheken (Language Environment) . . . . .	68
Anwendungsentwicklung. . . . .	69
Durchsatz der Sicherheitsprüfung verbessern . . . . .	69
Gemeinsame Klassennutzung durch mehrere JVMs . . . . .	69
Gemeinsame Klassennutzung aktivieren. . . . .	70
Begrenzung der Cachegröße. . . . .	70
Cachesicherheit . . . . .	70
SYS1.PARMLIB(BPXPRMxx). . . . .	71
Plattenspeicherplatz . . . . .	71
Dienstprogramme für Cacheverwaltung . . . . .	71
Feste Java-Heap-Größe . . . . .	72
Workload-Management . . . . .	73

## **Anhang A. Mehrere Instanzen von Developer für System z ausführen. . . 75**

Identische Softwareversionen mit unterschiedlichen Konfigurationsdateien . . . . .	75
Alle anderen Situationen . . . . .	76

## **Anhang B. Konfigurationsprobleme lösen . . . . . 79**

Position von Protokolldateien . . . . .	79
Protokollierung von JES Job Monitor . . . . .	80
APPC-Transaktionsprotokollierung (TSO Commands Service) . . . . .	80
RSE-Protokollierung . . . . .	80
Protokollierung für IVP-Test fekfivpc . . . . .	81
Protokollierung von Fault Analyzer Integration . . . . .	81
Protokollierung von File Manager Integration . . . . .	81
CARMA-Protokollierung . . . . .	82
Speicherauszugsdateien . . . . .	82
MVS-Speicherauszüge . . . . .	82
Java-Speicherauszüge . . . . .	83
Positionen für z/OS UNIX-Speicherauszüge . . . . .	84
Programmsteuerung für RSE-Programme autorisieren . . . . .	84
Reservierte TCP/IP-Ports . . . . .	85
Größe des Adressbereichs . . . . .	86
INETD-Anforderungen . . . . .	87
In SYS1.PARMLIB(BPXPRMxx) festgelegte Begrenzungen . . . . .	87

Im Sicherheitsprofil gespeicherte Begrenzungen . . . . .	87
Von System-Exits erzwungene Begrenzungen . . . . .	87
Trace für Fehlerrückmeldungen. . . . .	87
APPC-Transaktion und TSO Commands Service . . . . .	89
Weitere Informationen. . . . .	90
Systemgrenzwerte . . . . .	90
Bekannte Probleme . . . . .	91
Host-Connect-Emulator . . . . .	91
Kontakt zur IBM Unterstützungsfunktion aufnehmen . . . . .	92

## **Anhang C. TCP/IP konfigurieren . . . . . 95**

Abhängigkeit vom Hostnamen . . . . .	95
Wissenswertes zu Auflösern . . . . .	95
Wissenswertes zur Suchreihenfolge für Konfigurationsdaten . . . . .	96
Suchreihenfolgen in der z/OS-UNIX-Umgebung . . . . .	96
Basiskonfigurationsdateien des Auflösers . . . . .	96
Umsetztabelle . . . . .	97
Lokale Hosttabellen . . . . .	98
Anwendung in Developer für System z . . . . .	98

## **Anhang D. INETD konfigurieren . . . . . 103**

inetd.conf . . . . .	103
ETC.SERVICES . . . . .	104
Suchreihenfolge in der z/OS-UNIX-Umgebung . . . . .	105
Suchreihenfolge in der nativen MVS-Umgebung . . . . .	105
Port-Definitionen in PROFILE.TCPIP . . . . .	106
/etc/inetd.pid . . . . .	107
Start . . . . .	107
/etc/rc . . . . .	107
/etc/inittab . . . . .	107
BPXBATCH . . . . .	108
Shell-Sitzung . . . . .	108
Sicherheit. . . . .	109
Anforderungen von Developer für System z . . . . .	110
INETD . . . . .	110
RSE-Dämon . . . . .	110

## **Anhang E. SSL konfigurieren. . . . . 111**

Vorhandene RSE-Konfiguration klonen . . . . .	112
Bestimmen, welche Dateien zu verwenden sind . . . . .	112
Keystore mit keytool erstellen . . . . .	113
Schlüsseldatenbank erstellen (nur für den Dämon) . . . . .	114
Schlüsseldatei mit RACF erstellen . . . . .	114
Schlüsseldatenbank mit gskkyman erstellen . . . . .	115
Datei ssl.properties aktualisieren, um SSL zu aktivieren . . . . .	120
Verbindung testen. . . . .	120

## **Anhang F. APPC konfigurieren . . . . . 123**

VSAM. . . . .	123
VTAM. . . . .	124
SYS1.PARMLIB(APPCPMxx) . . . . .	125
SYS1.PARMLIB(ASCHPMxx) . . . . .	126
APPC-Änderungen aktivieren . . . . .	127
Transaktion für den TSO Commands Service definieren . . . . .	127

## **Glossar . . . . . 129**

<b>Bemerkungen . . . . .</b>	<b>133</b>
------------------------------	------------

Marken und Servicemarken . . . . .	135
------------------------------------	-----





---

## Abbildungsverzeichnis

1.	Konfigurationsdatei für JES Job Monitor (FEJJCENFG) . . . . .	16
2.	JCL für JES Job Monitor . . . . .	19
3.	REXX für APPC-ISPF-Anzeigen . . . . .	25
4.	RSE-Konfigurationsdatei rsed.envvars. . . . .	36
5.	ISPF-Konfigurationsdatei ISPF.conf. . . . .	47
6.	SSL-Konfigurationsdatei ssl.properties. . . . .	53
7.	Konfigurationsdatei für Protokollierung rse-comm.properties . . . . .	54
8.	Konfigurationsdatei für hostbasierte Projekte projectcfg.properties. . . . .	55
9.	File-Manager-Konfigurationsdatei FMIEXT.properties . . . . .	56
10.	CARMA-Konfigurationsdatei CRASRV.properties . . . . .	60
11.	Start-JCL für INETD . . . . .	108
12.	Hostzertifikat importieren . . . . .	121
13.	Vorgaben . . . . .	122
14.	JCL zum Erstellen von APPC-VSAMs . . . . .	124
15.	SYS1.SAMPLIB(ATBAPPL) . . . . .	125
16.	SYS1.PARMLIB(APPCPMxx) . . . . .	126
17.	SYS1.PARMLIB(ASCHPMxx) . . . . .	127



---

## Tabellen

1. Referenzierte Veröffentlichungen . . . . .	xi	8. Prüfliste der High Level Qualifier in ELAXF* . . . . .	23
2. Installations- und Konfigurationsmatrix für Developer für System z . . . . .	1	9. Prüfliste für APPC-Transaktion . . . . .	25
3. Angepasste MVS-Member im Überblick . . . . .	11	10. ELAXM*-Beispiel-Member der gespeicherten DB2-Prozedur . . . . .	27
4. Angepasste z/OS-UNIX-Dateien im Überblick . . . . .	13	11. Optionale Konfigurationsdateien . . . . .	35
5. Anpassung in nicht zu Developer für System z gehörenden Bibliotheken . . . . .	14	12. Prüfliste für die Clientkomponente von Devel- oper für System z . . . . .	65
6. Konsolbefehle für JES Job Monitor . . . . .	21	13. Für den Auflöser verfügbare lokale Definitio- nen . . . . .	100
7. ELAXF*-Beispielprozeduren . . . . .	22		



---

## Zu diesem Handbuch

Dieses Handbuch beschäftigt sich mit der Konfiguration der Funktionen von IBM Rational Developer für System z. Es enthält Konfigurationsanweisungen für IBM Rational Developer für System z Server auf Ihrem z/OS-Hostsystem.

Im weiteren Verlauf dieses Handbuchs werden die folgenden Namen verwendet:

- Für *IBM Rational Developer für System z* wird die Bezeichnung *Developer für System z* verwendet.
- *IBM Rational Developer für System z - Common Access Repository Manager* wird als *Common Access Repository Manager* bezeichnet. An einigen Stellen wird die Abkürzung *CARMA* verwendet.
- Das *IBM Software Configuration and Library Manager (SCLM) Developer Toolkit* wird als *SCLM Developer Toolkit* bezeichnet. An einigen Stellen wird die Abkürzung *SCLMDT* verwendet.

**Anmerkung:** Die Konfigurationsdaten in diesem Dokument gelten für IBM Rational Developer für System z Version 7.1.1.

Die Konfigurationsdaten für frühere Versionen, einschließlich IBM WebSphere Developer für System z, IBM WebSphere Developer für zSeries und WebSphere Studio Enterprise Developer, sind in den Veröffentlichungen 'Hostkonfiguration' und 'Program Directory' der entsprechenden Releases enthalten.

---

## Zielgruppe

Dieses Handbuch wendet sich an Systemprogrammierer, die IBM Rational Developer für System z Version 7.1.1 auf ihrem z/OS-Hostsystem installieren und konfigurieren möchten. Voraussetzung für die Verwendung dieses Handbuchs ist, dass Sie mit z/OS-UNIX- und MVS-Hostsystemen vertraut sind.

---

## Referenzierte Veröffentlichungen

In diesem Dokument werden die folgenden Veröffentlichungen referenziert:

*Tabelle 1. Referenzierte Veröffentlichungen*

Titel der Veröffentlichung	IBM Form	Bezug	Referenz-Link
Java Diagnostic Guide	SC34-6650	Java 5.0	<a href="http://www.ibm.com/developerworks/java/jdk/diagnosis/index.html">http://www.ibm.com/developerworks/java/jdk/diagnosis/index.html</a>
Java SDK and Runtime Environment User Guide		Java 5.0	<a href="http://www-03.ibm.com/servers/eserver/zseries/software/java/">http://www-03.ibm.com/servers/eserver/zseries/software/java/</a>
Program Directory for IBM Rational Developer for System z	GI11-8298-00	RD/z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/index.html">http://www-306.ibm.com/software/awdtools/rdz/library/index.html</a>

Tabelle 1. Referenzierte Veröffentlichungen (Forts.)

Titel der Veröffentlichung	IBM Form	Bezug	Referenz-Link
Program Directory for IBM Rational Developer for System z Common Access Repository Manager	GI11-8299-00	RD/z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/index.html">http://www-306.ibm.com/software/awdtools/rdz/library/index.html</a>
Program Directory for IBM Software Configuration and Library Manager (SCLM) Developer Toolkit	GI11-8306-00	RD/z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/index.html">http://www-306.ibm.com/software/awdtools/rdz/library/index.html</a>
Rational Developer for System z Application Deployment Manager User's Guide	SC23-7661	RD/z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/index.html">http://www-306.ibm.com/software/awdtools/rdz/library/index.html</a>
Rational Developer for System z Common Access Repository Manager Developer's Guide	SC23-7660	RD/z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/index.html">http://www-306.ibm.com/software/awdtools/rdz/library/index.html</a>
Rational Developer für System z Hostplanung	GI11-3123-00	RD/z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/index.html">http://www-306.ibm.com/software/awdtools/rdz/library/index.html</a>
SCLM Developer Toolkit Installation und Anpassung	SC12-4101	RD/z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/index.html">http://www-306.ibm.com/software/awdtools/rdz/library/index.html</a>
ABCs of z/OS System Programming Volume 9	SG24-6989	Redbook	<a href="http://www.redbooks.ibm.com/">http://www.redbooks.ibm.com/</a>
APPC and WebSphere Developer for System z	SC23-5885	White Paper	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/index.html">http://www-306.ibm.com/software/awdtools/rdz/library/index.html</a>
Host Based Projects in WebSphere Developer for System z Version 7.0		White Paper	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/index.html">http://www-306.ibm.com/software/awdtools/rdz/library/index.html</a>
Setting up SSL for RSE Communication	SC23-5816	White Paper	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/index.html">http://www-306.ibm.com/software/awdtools/rdz/library/index.html</a>
Communications Server Bookshelf	F1A1BK61	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Communications Server IP Configuration Guide	SC31-8775	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Communications Server IP Configuration Reference	SC31-8776	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Communications Server IP Diagnosis Guide	GC31-8782	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>

Tabelle 1. Referenzierte Veröffentlichungen (Forts.)

Titel der Veröffentlichung	IBM Form	Bezug	Referenz-Link
Communications Server IP System Administrator's Commands	SC31-8781	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Communications Server SNA Network Implementation Guide	SC31-8777	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Cryptographic Services System SSL Programming	SC24-5901	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Language Environment Customization	SA22-7564	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Language Environment Debugging Guide	GA22-7560	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
MVS Bookshelf	IEA2BK60	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
MVS Initialization and Tuning Guide	SA22-7591	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
MVS Initialization and Tuning Reference	SA22-7592	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
MVS JCL Reference	SA22-7597	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
MVS Planning APPC/MVS Management	SA22-7599	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
MVS Planning Workload Management	SA22-7602	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
MVS System Commands	SA22-7627	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Security Server RACF Command Language Reference	SA22-7687	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Security Server RACF Security Administrator's Guide	SA22-7683	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
UNIX System Services Command Reference	SA22-7802	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
UNIX System Services File System Interface Reference	SA22-7808	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
UNIX System Services Planning	GA22-7800	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
UNIX System Services User's Guide	SA22-7801	z/OS 1.7	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>





---

## Kapitel 1. Hostkomponenten installieren und konfigurieren

Installieren Sie die erforderlichen FMIDs für die folgenden Funktionen. Installationsinformationen zu den verschiedenen FMIDs finden Sie in den entsprechenden Program-Directory-Veröffentlichungen.

*Tabelle 2. Installations- und Konfigurationsmatrix für Developer für System z*

Benötigte Funktion von IBM Rational Developer für System z	Zu installierende FMIDs	Installations- und Konfigurationsinformationen
<ul style="list-style-type: none"><li>• Hostkonnektivität</li><li>• JES-Konnektivität</li><li>• Fernkompilierung</li><li>• Fehlerrückmeldungen für Fernkompilierung</li><li>• Fernes Debugging</li><li>• Gespeicherte DB2-Prozeduren</li><li>• Unterstützung für IMS-MFS-Anzeigen</li><li>• Unterstützung für CICS-BMS-Masken</li><li>• Unterstützung bidirektionaler Sprachen (BIDI) für CICS</li><li>• Application Deployment Manager (ADM)</li><li>• File Manager Integration</li><li>• Fault Analyzer Integration</li></ul>	HHOP710, HSD3310*	<ul style="list-style-type: none"><li>• <i>Program Directory for IBM Rational Developer for System z (IBM Form GI11-8298-00)</i></li><li>• <i>Rational Developer für System z Hostkonfiguration (IBM Form SC12-4062)</i></li><li>• <i>Rational Developer für System z Hostplanung (IBM Form GI11-3123-00)</i></li></ul> optional <ul style="list-style-type: none"><li>• <i>Program Directory for IBM Software Configuration and Library Manager (SCLM) Developer Toolkit (IBM Form GI11-8306-00)</i></li><li>• <i>SCLM Developer Toolkit Installation und Anpassung (IBM Form SC12-4101)</i></li></ul>
<ul style="list-style-type: none"><li>• Common Software Configuration Management Access (CARMA)</li></ul>	HCMA710, HHOP710**	<ul style="list-style-type: none"><li>• <i>Program Directory for IBM Rational Developer for System z - Common Access Repository Manager (IBM Form GI11-8299-00)</i></li></ul> optional <ul style="list-style-type: none"><li>• <i>Program Directory for IBM Rational Developer for System z (IBM Form GI11-8299-00)</i></li><li>• <i>Rational Developer für System z Hostkonfiguration (IBM Form SC12-4062)</i></li><li>• <i>Rational Developer für System z Hostplanung (IBM Form GI11-3123-00)</i></li></ul>
<ul style="list-style-type: none"><li>• Software Configuration and Library Manager (SCLM) Developer Toolkit</li></ul>	HSD3310	<ul style="list-style-type: none"><li>• <i>Program Directory for IBM Software Configuration and Library Manager (SCLM) Developer Toolkit (IBM Form GI11-8306-00)</i></li><li>• <i>SCLM Developer Toolkit Installation und Anpassung (IBM Form SC12-4101)</i></li></ul>

(\*) Developer für System z erfordert eine Verbindung zum TSO Commands Service unter z/OS. Diese Verbindung wird auf eine der folgenden Arten bereitgestellt:

1. Installation und Konfiguration des SCLM Developer Toolkit, FMID HSD3310 (Standard)
2. Installation und Konfiguration einer APPC-Transaktion

(\*\*) CARMA erfordert eine hostbasierte Schnittstelle. HHOP710 stellt diese Funktion bereit. Wenn Sie auf die Installation von HHOP710 verzichten möchten, können Sie eine selbst erstellte hostbasierte Schnittstelle verwenden.

---

## Hinweise zur Installationsvorbereitung

- Wenn Sie mit einer früheren Version von IBM Rational Developer für System z, IBM WebSphere Developer für System z, IBM WebSphere Developer für zSeries oder WebSphere Studio Enterprise Developer arbeiten, sollten Sie die zugehörigen Anpassungsdateien speichern, bevor Sie das Upgrade auf IBM Rational Developer für System z Version 7.1.1 installieren. Im Abschnitt „Bereits konfigurierte Dateien sichern“ auf Seite 11 finden Sie einen Überblick über die Dateien, die eine Anpassung erforderten.
- Falls Sie planen, mehrere Instanzen von Developer für System z auszuführen, lesen Sie Anhang A, „Mehrere Instanzen von Developer für System z ausführen“, auf Seite 75.
- Ab Version 7.1 stellt Developer für System z zwei Zugriffsmethoden für den TSO Commands Service bereit, der implizit über das Produkt oder explizit durch den Benutzer TSO/ISPF-Befehle ausführen kann.
  - Aufrufen einer Funktion des SCLM Developer Toolkit. Diese Methode erfordert die Installation und Konfiguration des SCLMDT (FMID HSD3310), das im Lieferumfang von Developer für System z enthalten ist. Dies ist die in den Beispiel-Memberrn verwendete Standardmethode.

Sie müssen die folgenden, in der Veröffentlichung *SCLM Developer Toolkit Installation und Anpassung (IBM Form SC12-4101)* beschriebenen SCLMDT-Anpassungsschritte ausführen:

- Erstellen der Verzeichnisstruktur und Konfigurationsdateien
- Anpassen von ISPF.conf

**Anmerkung:** Weder das SCLM-Produkt noch der Eclipse-basierte Client ist zur Unterstützung dieser Funktion von SCLMDT erforderlich.

- Eine APPC-Transaktion (wie in den Vorgängerreleases von Version 7.1). Diese Methode erfordert die Installation und Konfiguration von APPC, einem Feature von z/OS.
- Der Name und die Codierung des BIDI-Lademoduls wurden in Version 7.0 geändert, so dass die älteren Module entfernt werden müssen und das neue Modul konfiguriert werden muss.
- Der mit früheren Releases (Vorgängerreleases von Version 7.0) erstellte BIDI-Code muss ERNEUT KOMPILIERT werden, damit er das neue Modul FEJB-DTRX verwendet.

Ausführliche Anweisungen für die SMP/E-Installation der einzelnen FMIDs können Sie den entsprechenden Program-Directory-Veröffentlichungen entnehmen, die in Tabelle 2 auf Seite 1 aufgelistet sind.

---

## Hinweise zur Konfigurationsvorbereitung

Für Developer für System z gibt es eine Liste vorausgesetzter Software, die installiert und betriebsbereit sein muss, damit das Produkt funktioniert. Außerdem gibt es eine Liste zusätzlich erforderlicher Software zur Unterstützung bestimmter Features von Developer für System z. Zur Laufzeit muss diese zusätzlich erforderliche Software installiert und betriebsbereit sein, damit das entsprechende Feature ordnungsgemäß funktionieren kann.

Eine Liste der Produkte, die für Ihre Version von Developer für System z vorausgesetzt werden bzw. zusätzlich erforderlich sind, enthält die Veröffentlichung *Rational Developer für System z Hostplanung* (IBM Form GI11-3123-00).

**Achtung:** Die 64-Bit-Java-Version wird NICHT unterstützt.

## Erforderliche Konfiguration für vorausgesetzte Produkte und Software

Fragen Sie bei Ihrem MVS-Systemprogrammierer, beim Sicherheitsadministrator und beim TCP/IP-Administrator nach, ob die vorausgesetzten Produkte und die erforderliche Software installiert und getestet sind und funktionieren.

- Ab Version 7.1 stellt Developer für System z zwei Methoden für die Ausführung von TSO/ISPF-Befehlen bereit. Die Befehle können implizit über das Produkt oder explizit durch den Benutzer ausgeführt werden.
  - Aufrufen einer Funktion des SCLM Developer Toolkit. Diese Methode erfordert die Installation und Konfiguration des SCLMDT (FMID HSD3310), das im Lieferumfang von Developer für System z enthalten ist. Dies ist die in den Beispiel-Memberrn verwendete Standardmethode.

Sie müssen die folgenden, in der Veröffentlichung *SCLM Developer Toolkit Installation und Anpassung* (IBM Form SC12-4101) beschriebenen SCLMDT-Anpassungsschritte ausführen:

    - Erstellen der Verzeichnisstruktur und Konfigurationsdateien
    - Anpassen von ISPF.conf

**Anmerkung:** Weder das SCLM-Produkt noch der Eclipse-basierte Client ist zur Unterstützung dieser Funktion von SCLMDT erforderlich.

- Eine APPC-Transaktion (wie in den Vorgängerreleases von Version 7.1). Diese Methode erfordert die Installation und Konfiguration von APPC, einem Feature von z/OS.
- Die C/C++-DLL-Klassenbibliothek CBC.SCLBDLL und die LE-Laufzeitbibliotheken CEE.SCEERUN und CEE.SCEERUN2 müssen in der LINKLIST enthalten sein.
- Alle z/OS-UNIX-Benutzer müssen für die Java-Verzeichnisse die Zugriffsrechte READ und EXECUTE haben.
- Developer für System z ist bei der Einrichtung der Verbindung zwischen Client und Host auf INETD angewiesen. Weitere Informationen zu REXEC und SSH enthält Anhang D, „INETD konfigurieren“, auf Seite 103.
- Ferne (hostbasierte) Aktionen für z/OS-UNIX-Unterprojekte erfordern, dass auf dem Host REXEC oder SSH aktiv ist.
- Developer für System z ist bei der Initialisierung davon abhängig, dass TCP/IP mit dem richtigen Hostnamen konfiguriert ist. Dies impliziert, dass die verschiedenen TCP/IP- und Resolver-Konfigurationsdateien ordnungsgemäß definiert sein müssen. Informationen zur Anpassung von TCP/IP und Auflöser (Resolver) finden Sie in Anhang C, „TCP/IP konfigurieren“, auf Seite 95 und im Abschnitt

*TCPIP.DATA configuration statements der Veröffentlichung Communications Server IP Configuration Reference (IBM Form SC31-8776).*

Sie können Ihre TCP/IP-Konfiguration mit dem TSO-Befehl **HOMETEST** testen. Weitere Informationen zu diesem Befehl enthält der Abschnitt *TSO Commands* der Veröffentlichung *Communications Server IP System Administrator's Commands* (IBM Form SC31-8781).

Beispielausgabe des Befehls **HOMETEST**:

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

The FTP configuration parameter file used will be "SYS1.TCPPARMS(FTPDATA)"

TCP Host Name is: CDFMVS08

Using Name Server to Resolve CDFMVS08

The following IP addresses correspond to TCP Host Name: CDFMVS08  
9.42.112.75

The following IP addresses are the HOME IP addresses defined in PROFILE.TCPIP:  
9.42.112.75  
127.0.0.1

All IP addresses for CDFMVS08 are in the HOME list!

Hometest was successful - all Tests Passed!

## Hinweise zur Benutzer-ID

Die Benutzer-ID eines Benutzers von Developer für System z muss (mindestens) die folgenden Attribute haben:

- TSO-Zugriff (mit normaler Regionsgröße)
- Ein für das Sicherheitssystem (z. B. RACF) definiertes OMVS-Segment für die Benutzer-ID und für die zugehörige Standardgruppe. Dieses OMVS-Segment kann ein individuelles oder ein systemweites Standardsegment sein. Weitere Informationen zum Standard-OMVS-Segment enthält der *Security Server RACF Security Administrator's Guide* (IBM Form SA22-7683).
  - Ein im OMVS-Segment angegebenes und dem Benutzer zugeordnetes Ausgangsverzeichnis (mit den Zugriffsrechten WRITE, READ und EXECUTE).
  - Das Feld PROGRAM im OMVS-Segment sollte auf /bin/sh oder eine andere gültige z/OS-UNIX-Shell, z. B. /bin/tcsh, gesetzt sein.
  - Das Feld ASSIZEMAX sollte nicht gesetzt sein, so dass Systemstandardwerte verwendet werden.
  - UID 0 ist für die Benutzer-ID nicht erforderlich.

Beispiel (Befehl **LISTUSER userid NORACF OMVS**):

USER=userid

OMVS INFORMATION

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

- Für die Standardgruppe, zu der die Benutzer-ID gehört, ist eine Gruppen-ID (GID) erforderlich.

Beispiel (Befehl **LISTGRP group NORACF OMVS**):

GROUP group

OMVS INFORMATION

-----  
GID= 0000003243

- Zugriffsrechte READ und EXECUTE für das Installations- und Konfigurationsverzeichnis von Developer für System z (standardmäßig /usr/lpp/wd4z/\*)

## Hinweise zum Server

Zu den folgenden Hostservices (und somit zu den zugehörigen Ports) muss der Client eine Verbindung herstellen können. An allen anderen von Developer für System z verwendeten Ports gibt es nur Hostdatenverkehr. Das bedeutet, dass für die Firewall, die den Host schützt, nur die aufgelisteten Ports von Developer für System z definiert werden müssen.

- INETD-Service zum Starten des RSE-Servers. Developer für System z bietet hierfür mehrere Möglichkeiten an:
  - RSE-Dämon (Standard-Port 4035)
  - REXEC (Standard-Port 512)
  - SSH (Standard-Port 22)
- RSE-Server für die Kommunikation zwischen Client und Host. Standardmäßig wird ein beliebiger verfügbarer Port verwendet werden. Eine Einschränkung auf einen bestimmten Port-Bereich ist jedoch möglich. Die Kommunikation über diesen Port kann mit SSL verschlüsselt werden.
- TN3270-Telnet-Service für den Host-Connect-Emulator (Standard-Port 23). Die Kommunikation über diesen Port kann mit SSL verschlüsselt werden. (Der Standard-Port ist dann 992.) Welcher Standard-Port dem Telnet-Service TN3270 zugeordnet wird, hängt davon ab, ob der Benutzer sich für oder gegen die Verwendung der Verschlüsselung entscheidet.

**Anmerkung:** Ältere Clientversionen (bis Version 7.0) kommunizieren direkt mit dem JES Job Monitor am Standard-Port 6715.

**Anmerkung:** Während einer fernen Debug-Sitzung für COBOL, PL/I oder Assembler wird das IBM Debug Tool für z/OS aufgerufen. Dieses Produkt kommuniziert direkt mit dem Client. Die Kommunikation wird auf dem Host eingeleitet und stellt eine Verbindung zum Port 8001 des Clients her.

INETD ist ein z/OS-UNIX-Serverprozess, der für die Einrichtung der Client-Host-Verbindungen von Developer für System z erforderlich ist. Die Umgebungseinstellungen von INETD werden übergeben, wenn ein Prozess gestartet wird. Die Berechtigungen für die INETD-Benutzer-ID müssen ordnungsgemäß gesetzt sein, damit INETD den RSE-Server starten kann.

- Wenn INETD über JCL mit BPXBATCH gestartet wird, muss die Regionsgröße gleich null sein.
- Wird INETD in einer TSO/OMVS-Sitzung gestartet, muss die Regionsgröße bei mindestens 2096128 liegen.
- Zugriffsrechte READ und EXECUTE für das Installationsverzeichnis von Developer für System z (standardmäßig /usr/lpp/wd4z/\*)
- Berechtigung für BPX.DAEMON und nach Möglichkeit für UID 0

Weitere Informationen zu INETD-Berechtigungen enthält Anhang D, „INETD konfigurieren“, auf Seite 103.

Remote Systems Explorer (RSE) ist die Komponente von Developer für System z, die Kernservices wie die Verbindung vom Client zum Host bereitstellt.

- Für jede Client-Host-Verbindung gibt es einen privaten RSE-Server.
- Neben den in diesem Handbuch dokumentierten Ports für die Kommunikation mit anderen Servern, auf denen Developer für System z installiert ist, benötigt der RSE-Server mehrere Ports zur internen Verwendung.

## Erforderliche Berechtigungen für die Implementierung der Konfigurations-Tasks

Die Konfiguration von Developer für System z erfordert mehr als die üblichen Systemprogrammiererberechtigungen, so dass eine gewisse Mindestunterstützung von anderen eingeplant werden sollte. In der folgenden Liste sind die wichtigsten Bereiche aufgeführt:

- Aktualisierungsberechtigung für Systemdateigruppen (wie SYS1.PARMLIB)
- Berechtigung zum Absetzen von Konsolbefehlen
- Berechtigung zum Erstellen von z/OS-UNIX-Verzeichnissen
- Berechtigung zum Aktualisieren des z/OS-UNIX-Verzeichnisses /etc
- Berechtigung zum Setzen des z/OS-UNIX-Programmsteuerungsbits
- Berechtigung zum Definieren neuer TCP/IP-Ports
- Berechtigung für BPX.DAEMON und nach Möglichkeit für UID 0 zum Starten bzw. erneuten Starten von INETD
- Berechtigungen für das WLM/RSM-Management
- APPC-Managementberechtigungen für die Erstellung eines Transaktionsprogramms
- Berechtigungen eines Sicherheitsadministrators zum Abfragen und Erstellen von Sicherheitsprofilen
- Berechtigungen eines DB2-Administrators zum Erstellen einer gespeicherten Prozedur (optional)
- Berechtigungen eines CICS-Administrators zum Definieren von Programmen für CICS (optional)

---

## Hinweise zur Deployment-Vorbereitung

Developer für System z und Common Access Repository Manager (CARMA) unterstützen das Klonen einer Installation auf einem anderen System, so dass Sie nicht auf jedem System eine SMP/E-Installation durchführen müssen.

Für das Deployment auf anderen Systemen sind die nachfolgenden Dateigruppen, Verzeichnisse und Dateien obligatorisch. Falls Sie eine Datei an eine andere Position kopiert haben, um sie anzupassen, muss die entsprechende Datei in den folgenden Listen durch Ihre angepasste Datei ersetzt werden.

**Anmerkung:** Die folgenden Listen enthalten nicht die für das Deployment vorausgesetzter und zusätzlich erforderlicher Software benötigten Dateien.

## IBM Rational Developer für System z, FMID HHOP710

- HLQ.SFEKLOAD(\*)
- HLQ.SFEKPROC(\*)
- HLQ.SFEKSAMP(FEJJCENFG)
- HLQ.SFEKSAMP(FEJJJCL)
- HLQ.SFEKSAMP(ELAXF\*)
- /usr/lpp/wd4z/\*
- Optionale Komponenten
  - HLQ.SFEKSAMP(FEJTSO) - TSO Submit
  - HLQ.SFEKSAMP(ELAXM\*) - gespeicherte DB2-Prozedur
  - Von HLQ.SFEKSAMP(ADNPCCSD) erstellte CICS-Definitionen - primäre ADM-Verbindung
  - Von HLQ.SFEKSAMP(ADNARCSD) erstellte CICS-Definitionen - nicht primäre ADM-Verbindung
  - Von HLQ.SFEKSAMP(ADNVSAM) erstellte VSAM - ADM-CRD-Server
  - Von HLQ.SFEKSAMP(ADNCSMGH) erstelltes Lademodul - ADM-Pipeline-Handler
  - Von HLQ.SFEKSAMP(FEKAPPCC) erstellte APPC-Transaktion - APPC-basierter TSO Commands Service
  - /etc/SCLMDT/CONFIG/ISPF.conf - Hinzufügen des TSO Commands Server zur ISPF-Umgebung des SCLM Developer Toolkit

**Anmerkung:** HLQ und /usr/lpp/wd4z sind der während der Produktinstallation verwendete High Level Qualifier (standardmäßig FEK) und Pfad.

## IBM Common Access Repository Manager (CARMA), FMID HCMA710

- HLQ.SCRALOAD(\*)
- HLQ.SCRACLIST(CRASUBMT)
- Von folgenden HLQ.SCRASAMP-Mitgliedern erstellte VSAMs
  - HLQ.SCRASAMP(CRA\$VMSG) - VSAM für Nachrichten
  - HLQ.SCRASAMP(CRA\$VDEF) - VSAM für Konfiguration
  - HLQ.SCRASAMP(CRA\$SVSTR) - VSAM für angepasste Informationen

**Anmerkung:** HLQ ist der während der Produktinstallation verwendete High Level Qualifier (standardmäßig CRA).





---

## Kapitel 2. Installations- und Konfigurationsänderungen

Dieser Abschnitt hebt die Installations- und Konfigurationsänderungen im Vergleich zu früheren Produktreleases hervor. Weitere Informationen hierzu finden Sie in den entsprechenden Abschnitten dieses Handbuchs.

---

### Änderungen zwischen Version 7.0 und Version 7.1

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

- **Ergänzung:** Konfigurationsoption - Ausführung von TSO/ISPF-Befehlen über eine APPC-Transaktion oder über das SCLM Developer Toolkit
- **Änderung:** Die APPC-Transaktion nutzt ein neues ISPF-Feature.
- **Ergänzung:** Die folgenden anpassbaren Member sind neu hinzugekommen:
  - SAMPLIB ELAXFADT
  - SAMPLIB ADNCMSGH
  - /usr/lpp/wd4z/rse/lib/FMIEXT.properties
- **Änderung:** Die folgenden Member wurden verschoben:
  - SFEKDLL(FEJBDTRX) -> SFEKLOAD(FEJBDTRX)
- **Änderung:** Die folgenden anpassbaren Member wurden geändert:
  - 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 Common Access Repository Manager (CARMA), FMID HCMA710

- **Änderung:** Protokollaufzeichnungen werden in die DD-Anweisung CARMA-LOG geschrieben.
- **Änderung:** Die VSAM für CARMA-Nachrichten (CRAMSG) und die VSAM für Konfiguration wurden aktualisiert.
- **Ergänzung:** Die folgenden anpassbaren Member sind neu hinzugekommen:
  - SAMPLIB CRA#ECOB
  - SAMPLIB CRA#EPDS
  - SAMPLIB CRA#ERAM
  - SAMPLIB CRA#ESLM
- **Umbenennung:** Die folgenden anpassbaren Member wurden umbenannt:
  - SAMPLIB CRAREPR -> CRA\$VDEF
  - SAMPLIB CRAMREPR -> CRA\$VMSG
  - SAMPLIB CRASREPR -> CRA\$VSTR
  - SAMPLIB CRASALX -> CRA#ASLM
  - SAMPLIB CRACOBJ1 -> CRA#CCB1
  - SAMPLIB CRACOBJ2 -> CRA#CCB2
  - SAMPLIB CRACLICM -> CRA#CCLT
  - SAMPLIB CRARAMCS -> CRA#CPDS

- SAMPLIB CRARAMCM -> CRA#CRAM
- SAMPLIB CRATREPR -> CRA#VPDS
- SAMPLIB CRALREPR -> CRA#VSLM
- SAMPLIB CRACLIRN -> CRA#XCLT
- **Änderung:** Die folgenden anpassbaren Member wurden geändert:
  - CLIST CRASUBMT

---

## Änderungen zwischen Version 6.0.1 und Version 7.0

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

- **Änderung:** IBM WebSphere Developer für System z (FMID HHOP700) vereint in sich jetzt die Funktionen von fünf verschiedenen Produkten, um den Installationsaufwand zu verringern.
  - V601 FMID H001600, IBM WebSphere Developer für zSeries RSE + ICU
  - V601 FMID H002600, IBM WebSphere Developer für zSeries JES Job Monitor
  - V601 FMID HEDS500, IBM WebSphere Developer für zSeries Optionen für z/OS (in HHOP700 sind nicht alle Funktionen enthalten)
  - V601 FMID HBDI601, IBM WebSphere Developer für zSeries BIDI-Unterstützung
  - Neu: Version 700, IBM Application Deployment Manager
- **Änderung:** Die Installationsjobs wurden so umbenannt, dass die Schritte in alphabetischer Reihenfolge ausgeführt werden.
- **Änderung:** Die SMP/E-Installation setzt jetzt Programmsteuerungsbits in z/OS UNIX.
- **Änderung:** Die REXX-Servercodierung wurde von SAMPLIB in die neue Dateigruppe SFEKPROC verschoben.
- **Änderung:** Für BIDI wird ein neues Lademodul verwendet. Alle Verweise auf die alten Module müssen entfernt werden.
- **Änderung:** Der Parameter JESNAME ist nicht mehr in der Konfigurationsdatei von JES Job Monitor (SAMPLIB FEJJCNGF) enthalten.
- **Änderung:** Folgende Member müssen nicht mehr angepasst werden:
  - /usr/lpp/wd4z/rse/lib/setup.env.zseries
- **Ergänzung:** Die folgenden Member sind neu hinzugekommen:
  - SAMPLIB ELAXFCPC
  - SAMPLIB ELAXFCPP
  - SAMPLIB ELAXFMFS
  - /usr/lpp/wd4z/rse/lib/projectcfg.properties
- **Umbenennung:** Die folgenden anpassbaren Member wurden umbenannt:
  - SAMPLIB FEKFRDIS -> FEKAPPCL
  - SAMPLIB FEKFRDEL -> FEKAPPCX
  - SAMPLIB FEKFRDAD -> FEKAPPCC
- **Löschung:** Die folgenden Member sind nicht mehr im Lieferumfang enthalten:
  - SAMLIB ELAXFLNK
  - SAMPLIB FEJIVJCL
  - LOADLIB FEJBDTRN
  - LOADLIB FEJBDTRE

## IBM Common Access Repository Manager (CARMA), FMID HCMA700

- **Änderung:** Die Installationsjobs wurden so umbenannt, dass die Schritte in alphabetischer Reihenfolge ausgeführt werden.
- **Änderung:** Folgende Member müssen nicht mehr angepasst werden:
  - /usr/lpp/wd4z/rse/lib/rsed.envvars
  - /usr/lpp/wd4z/rse/lib/rexxsub

---

### Bereits konfigurierte Dateien sichern

Wenn Sie mit einer früheren Version von Developer für System z arbeiten, sollten Sie die zugehörigen Anpassungsdateien speichern, bevor Sie das Upgrade auf Version 7.1.1 installieren. Falls Sie planen, mehrere Instanzen von Developer für System z auszuführen, lesen Sie vor Beginn der Anpassung Anhang A, „Mehrere Instanzen von Developer für System z ausführen“, auf Seite 75.

Tabelle 3 und Tabelle 4 auf Seite 13 geben einen Überblick über die Dateien, die ab Developer für System z und CARMA Version 6.0.1 angepasst worden sein können. In Tabelle 5 auf Seite 14 sind die Anpassungen aufgelistet, die während der Installation von Developer für System z und CARMA ab Version 6.0.1 an Dateigruppen sowie an vorausgesetzten und zusätzlich erforderlichen Produkten vorgenommen werden.

*Tabelle 3. Angepasste MVS-Member im Überblick*

Member	Standardposition in Version 6.0.1	Standardposition in Version 7.0/7.1	Zweck
FEJJCNFG	HLQ.SFEJSAMP (HLQ = FEJ)	HLQ.SFEKSAMP (HLQ = FEK)	Konfigurationsdatei für JES Job Monitor
FEJJJCL	HLQ.SFEJSAMP (HLQ = FEJ)	HLQ.SFEKSAMP (HLQ = FEK)	JCL für JES Job Monitor
FEJTSO	HLQ.SFEJSAMP (HLQ = FEJ)	HLQ.SFEKSAMP (HLQ = FEK)	JCL für TSO-Übergabe
FEKAPPCC	nicht vorhanden	HLQ.SFEKSAMP (HLQ = FEK)	JCL zum Erstellen einer APPC-Transaktion
FEKAPPCL	nicht vorhanden	HLQ.SFEKSAMP (HLQ = FEK)	JCL zum Anzeigen einer APPC-Transaktion
FEKAPPCX	nicht vorhanden	HLQ.SFEKSAMP (HLQ = FEK)	JCL zum Löschen einer APPC-Transaktion
FEKFRTAD	HLQ.SFEKSAMP (HLQ = FEK)	(neuer Member-Name FEKAPPCC)	JCL zum Erstellen einer APPC-Transaktion
FEKFRDIS	HLQ.SFEKSAMP (HLQ = FEK)	(neuer Member-Name FEKAPPCL)	JCL zum Anzeigen einer APPC-Transaktion

Tabelle 3. Angepasste MVS-Member im Überblick (Forts.)

Member	Standardposition in Version 6.0.1	Standardposition in Version 7.0/7.1	Zweck
FEKFRDEL	HLQ.SFEKSAMP (HLQ = FEK)	(neuer Member-Name FEKAPPLX)	JCL zum Löschen einer APPC-Transaktion
ELAXF*	HLQ.SCCUSAMP (HLQ = CCU)	HLQ.SFEKSAMP (HLQ = FEK)	JCL für ferne Projekt-Builds usw.
ELAXMSAM	HLQ.SCCUSAMP (HLQ = CCU)	HLQ.SFEKSAMP (HLQ = FEK)	JCL-Prozedur des WLM-Adressbereichs für den Stored Procedure Builder für PL/I und COBOL
ELAXMJCL	HLQ.SCCUSAMP (HLQ = CCU)	HLQ.SFEKSAMP (HLQ = FEK)	JCL zum Definieren des Stored Procedure Builder für COBOL und PL/I für DB2
ADNVSAM	nicht vorhanden	HLQ.SFEKSAMP (HLQ = FEK)	JCL zum Definieren des ADM-CRD-Repositorys
ADNPCCSD	nicht vorhanden	HLQ.SFEKSAMP (HLQ = FEK)	JCL zum Aktivieren des CRD-Servers in der primären CICS-Region
ADNCMSGH	nicht vorhanden	HLQ.SFEKSAMP (HLQ = FEK) (in Version 7.0 nicht vorhanden)	JCL zum Kompilieren des Pipelinenachrichten-Handlers
ADNSMSGH	nicht vorhanden	HLQ.SFEKSAMP (HLQ = FEK)	Beispielquellcode für den Pipelinenachrichten-Handler
ADNARCS	nicht vorhanden	HLQ.SFEKSAMP (HLQ = FEK)	JCL zum Aktivieren des CRD-Servers in den nicht primären CICS-Regionen
CRASUBMT	HLQ.SCRACLST (HLQ = CRA)	HLQ.SCRACLST (HLQ = CRA)	CLIST zum Übergeben von JCL für einen CARMA-Server

Tabelle 3. Angepasste MVS-Member im Überblick (Forts.)

Member	Standardposition in Version 6.0.1	Standardposition in Version 7.0/7.1	Zweck
CRAMREPR	HLQ.SCRASAM (HLQ = CRA)	HLQ.SCRASAMP (HLQ = CRA) (neuer Member-Name in Version 7.1: CRA\$VMSG)	JCL zur Erstellung der VSAM für CARMA-Nachrichten
CRAREPR	HLQ.SCRASAM (HLQ = CRA)	HLQ.SCRASAMP (HLQ = CRA) (neuer Member-Name in Version 7.1: CRA\$VDEF)	JCL zur Erstellung der VSAM für CARMA-Konfiguration
CRASREPR	HLQ.SCRASAM (HLQ = CRA)	HLQ.SCRASAMP (HLQ = CRA) (neuer Member-Name in Version 7.1: CRA\$VSTR)	JCL zur Erstellung der VSAM für angepasste CARMA-Informationen
CRALREPR	HLQ.SCRASAM (HLQ = CRA)	HLQ.SCRASAMP (HLQ = CRA) (neuer Member-Name in Version 7.1: CRA#VSLM)	JCL zur Erstellung der VSAM für SCLM-RAM-Nachrichten
CRASALX	HLQ.SCRASAM (HLQ = CRA)	HLQ.SCRASAMP (HLQ = CRA) (neuer Member-Name in Version 7.1: CRA#ASLM)	JCL zum Erstellen der SCLM-RAM-Dateigruppen
CRATREPR	HLQ.SCRASAM (HLQ = CRA)	HLQ.SCRASAMP (HLQ = CRA) (neuer Member-Name in Version 7.1: CRA#VPDS)	JCL zur Erstellung der VSAM für PDS-RAM-Nachrichten

**Anmerkung:** HLQ ist der während der Produktinstallation verwendete High Level Qualifier. Die IBM Standardeinstellungen für HLQ sind aufgelistet, müssen für Ihren Standort jedoch nicht zutreffen.

Tabelle 4. Angepasste z/OS-UNIX-Dateien im Überblick

Datei	Standardposition in Version 6.0.1	Standardposition in Version 7.0/7.1	Zweck
rse.envvars	/usr/lpp/wd4z/rse/lib/	/usr/lpp/wd4z/rse/lib/	RSE-Konfigurationsdatei
rsecomm.properties	/usr/lpp/wd4z/rse/lib/	/usr/lpp/wd4z/rse/lib/	RSE-Trace-Konfigurationsdatei
ssl.properties	/usr/lpp/wd4z/rse/lib/	/usr/lpp/wd4z/rse/lib/	RSE-SSL-Konfigurationsdatei
setup.env.zseries	/usr/lpp/wd4z/rse/lib/	(wird nicht mehr angepasst)	Export von RSE-Umgebungsvariablen
projectcfg.properties	(nicht vorhanden)	/usr/lpp/wd4z/rse/lib/	Konfigurationsdatei für Hostprojekte

Tabelle 4. Angepasste z/OS-UNIX-Dateien im Überblick (Forts.)

Datei	Standardposition in Version 6.0.1	Standardposition in Version 7.0/7.1	Zweck
FMIEXT.properties	(nicht vorhanden)	/usr/lpp/wd4z/rse/lib/ (in Version 7.0 nicht vorhanden)	File-Manager-Konfigurationsdatei
CRASRV.properties	/usr/lpp/wd4z/rse/lib/	/usr/lpp/wd4z/rse/lib/	CARMA-Konfigurationsdatei
rexsub	/usr/lpp/wd4z/rse/lib/	(wird nicht mehr angepasst)	CARMA-z/OS-UNIX-REXX zum Ausführen der MVS-CLIST CRASUBMT

**Anmerkung:** Der Pfad /usr/lpp/wd4z wird während der Produktinstallation verwendet. Die IBM Standardeinstellung ist angegeben, muss für Ihren Standort jedoch nicht zutreffen.

Tabelle 5. Anpassung in nicht zu Developer für System z gehörenden Bibliotheken

Member/ Datei	Standardposition	Erforderliche Anpassung
BPXPRMxx	SYS1.PARMLIB	Setzen Sie MAXASSIZE auf mindestens 2147483647.
PROGxx	SYS1.PARMLIB	Berechtigen Sie HLQ.SFEKLOAD für APF.
ASCHPMxx	SYS1.PARMLIB	Definieren Sie eine APPC-Transaktionsklasse für den TSO Commands Service.
services	/etc/	Definieren Sie den RSE-Dämon.
inetd.conf	/etc/	Definieren Sie den RSE-Dämon.
ISPF.conf	/etc/SCLMDT/ CONFIG/	Definieren Sie die Position für den TSO Commands Server.
/	APPC	Definieren Sie eine APPC-Transaktion für den TSO Commands Service.
/	WLM	Ordnen Sie das APPC-Transaktionsprogramm einer TSO-Leistungsgruppe zu.
/	WLM	Weisen Sie der gespeicherten DB2-Prozedur eine Anwendungsumgebung zu.

---

## Kapitel 3. MVS-Komponenten von Developer für System z aktivieren

Wenn Sie mit einer früheren Version von Developer für System z arbeiten, sollten Sie die zugehörigen Anpassungsdateien speichern, bevor Sie Version 7.1.1 installieren. Lesen Sie hierzu den Abschnitt „Bereits konfigurierte Dateien sichern“ auf Seite 11.

Alle in diesem Abschnitt enthaltenen Verweise auf HLQ beziehen sich auf den während der Installation von Developer für System z verwendeten High Level Qualifier. Die Standardeinstellung für die Installation ist FEK, die jedoch nicht für Ihren Standort zutreffen muss.

**Anmerkung:** Die C/C++-DLL-Klassenbibliothek CBC.SCLBDLL und die LE-Laufzeitbibliotheken CEE.SCEERUN und CEE.SCEERUN2 müssen in der LINKLIST enthalten sein.

Sie können die LINKLIST-Anforderung umgehen, indem Sie eine STEPLIB-Anweisung zu rsed.envvars hinzufügen. Lesen Sie hierzu den Abschnitt „RSE-Konfigurationsdatei rsed.envvars anpassen“ auf Seite 35.

---

### MAXASSIZE in SYS1.PARMLIB(BPXPRMxx) setzen

MAXASSIZE definiert die Regionsgröße des Adressbereichs/Adressierungsprozesses. Setzen Sie MAXASSIZE in SYS1.PARMLIB(BPXPRMxx) auf mindestens 2147483647.

Dieser Wert kann mit folgenden Konsolbefehlen überprüft und dynamisch (bis zum nächsten IPL) gesetzt werden. Lesen Sie hierzu die Beschreibung in der Veröffentlichung *MVS System Commands (IBM Form SA22-7627)*.

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

Weitere Informationen zu anderen Positionen, an denen die Größe des Adressbereichs gesetzt oder eingeschränkt werden kann, finden Sie im Abschnitt „Größe des Adressbereichs“ auf Seite 86.

---

### HLQ.SFEKLOAD für APF berechtigen

Das Modul FEJMON in der Ladebibliothek HLQ.SFEKLOAD muss für APF berechtigt werden, damit JES Job Monitor auf JES-Spool-Dateien zugreifen kann.

Wenn Sie sich an Ihrem Standort nach den IBM Empfehlungen gerichtet haben, sind die APF-Berechtigungen in SYS1.PARMLIB(PROGxx) definiert. Weitere Informationen zum Definieren von APF-Berechtigungen enthält die Veröffentlichung *MVS Initialization and Tuning Reference (IBM Form SA22-7592)*.

Zu Testzwecken können APF-Berechtigungen auch dynamisch erteilt werden. Solche Berechtigungen sind bis zum nächsten IPL des Systems gültig. Der erforderliche Konsolbefehl sieht wie folgt aus:

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

---

## Konfigurationsdatei für JES Job Monitor (FEJJCNFG) anpassen

JES Job Monitor ist die Komponente für die gesamte JES-Konnektivität von Developer für System z. Bestimmte Aspekte können in einer Konfigurationsdatei an die Anforderungen Ihres Standorts angepasst werden.

**Anmerkung:** Sie sollten die Beispielkonfigurationsdatei in eine neue Dateigruppe kopieren und diese Kopie anpassen, damit die Konfiguration im Falle einer Wartung nicht überschrieben wird.

Sie finden die Beispielkonfigurationsdatei in:

HLQ.SFEKSAMP(FEJJCNFG)

Die Datei umfasst eine Reihe von Definitionen für Umgebungsvariablen. Kommentarzeilen beginnen mit dem Nummernzeichen (#). Die Beispielkonfigurationsdatei ist in Abb. 1 aufgelistet.

```
SERV_PORT=6715
CODEPAGE=UTF-8
HOST_CODEPAGE=IBM-1047
TZ=EST5EDT
#LIMIT_VIEW=USERID
LISTEN_QUEUE_LENGTH=5
MAX_DATASETS=32
MAX_THREADS=200
TIMEOUT_INTERVAL=1200
TIMEOUT=3600
AUTHMETHOD=RACF
#_BPXK_SETIBMOPT_TRANSPORT=<TCP/IP-Stack-Name>
```

*Abbildung 1. Konfigurationsdatei für JES Job Monitor (FEJJCNFG)*

Folgende Definitionen sind erforderlich:

### **SERV\_PORT**

Die Port-Nummer für den Hostserver mit JES Job Monitor. Der Standard-Port ist 6715. Eine Änderung wird angeraten, allerdings müssen Client UND Server mit derselben Port-Nummer konfiguriert werden. Wenn Sie die Server-Port-Nummer ändern, müssen alle Benutzer der Clientkomponente von Developer für System z in der Ansicht 'Ferne Systeme' den JES-Job-Monitor-Port für dieses System ändern.

**Anmerkung:** Überprüfen Sie vor Auswahl eines Ports, ob der Port auf Ihrem System verfügbar ist. Verwenden Sie dazu die TSO-Befehle **NETSTAT** und **NETSTAT PORTL**. Weitere Informationen hierzu enthält der Abschnitt „Reservierte TCP/IP-Ports“ auf Seite 85.

**Anmerkung:** Wenn Sie einen Client ab Version 7.1 verwenden, ist die Kommunikation über diesen Port auf Ihre Hostmaschine beschränkt.

### **CODEPAGE**

Die Codepage der Workstation. Die Standardeinstellung ist UTF-8. Die Workstation-Codepage ist auf UTF-8 gesetzt und sollte generell nicht geändert werden.



dert werden. Falls Sie Schwierigkeiten mit Zeichen Ihrer Nationalsprache haben, z. B. mit dem Währungssymbol, müssen Sie UTF-8 möglicherweise an die Codepage der Workstation anpassen.

#### **HOST\_CODEPAGE**

Die Host-Codepage. Die Standardeinstellung ist IBM-1047. Diese Einstellung muss geändert werden.

#### **TZ**

Zeitzonenselektor. Die Standardeinstellung ist EST5EDT. Die Standardzeitzone ist UTC + 5 Stunden (Eastern Standard Time mit Sommerzeit). Setzen Sie diesen Wert auf Ihre Zeitzone. Zusätzliche Informationen hierzu finden Sie in der Veröffentlichung *UNIX System Services File System Interface Reference* (IBM Form SA22-7808).

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

Die Länge der TCP/IP-Warteschlange für eingehende Verbindungen. Die Standardeinstellung ist 5. Ändern Sie diese Einstellung nur auf Anweisung des IBM Support Center.

#### **MAX\_DATASETS**

Die Standardeinstellung ist 32. Dies ist die maximale Anzahl von Spool-Ausgabedateigruppen, die JES Job Monitor an den Client zurückgibt (z. B. SYSOUT, SYSPRINT, SYS00001 usw.).

#### **MAX\_THREADS**

Die Standardeinstellung ist 200. Dies ist die maximale Anzahl Benutzer, die JES Job Monitor gleichzeitig benutzen können. Wenn Sie diese Anzahl erhöhen, müssen Sie unter Umständen auch den Adressbereich von JES Job Monitor vergrößern.

#### **TIMEOUT\_INTERVAL**

Die Standardeinstellung ist 1200. Dieser Parameter steuert, wie oft der Server nach Threads mit überschrittenem Zeitlimit sucht und diese dann (mit kill) beendet (siehe **TIMEOUT**). Der Wert von **TIMEOUT\_INTERVAL** gibt die Zeit zwischen den Überprüfungen in Sekunden an.

#### **TIMEOUT**

Die Standardeinstellung ist 3600. Dieser Parameter gibt die Zeitspanne (in Sekunden) an, nach der ein Thread bei fehlender Interaktion mit dem Client (mit kill) beendet wird. Der Maximalwert ist 2147483647.

#### **AUTHMETHOD**

Die Standardeinstellung ist RACF und bedeutet, dass die Standardsicherheitsschnittstelle verwendet wird. Sie sollten diesen Wert nicht ändern, auch wenn Sie ein anderes Produkt als RACF verwenden.

Folgende Definitionen sind optional. Wenn Sie diese Definitionen übergehen, werden die angegebenen Standardwerte verwendet.

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

Diese Einstellung definiert, welche Ausgaben der Benutzer anzeigen kann. Wenn die Standardeinstellung (**LIMIT\_VIEW=NOLIMIT**) verwendet wird, kann der Benutzer alle JES-Ausgaben anzeigen. Bei Angabe von **USERID** werden nur die Ausgaben angezeigt, deren Eigner der Benutzer ist.

**Anmerkung:** Die einzigen gültigen Einstellungen sind **USERID** und **NOLIMIT**.

#### **\_BPXK\_SETIBMOPT\_TRANSPORT=<TCP/IP-Stack-Name>**

Gibt an, welcher TCP/IP-Stack verwendet werden soll, wenn es mehrere aktive Stacks gibt. Die Standardeinstellung ist TCPIP. <TCP/IP-Stack-Na-

me> ist der angeforderte TCP/IP-Stack-Name, wie der in der Anweisung TCPIPJOBNAME der zugehörigen TCPIP.DATA definiert ist.

**Anmerkung:** Die angeforderte Stack-Affinität kann nicht durch das Codieren einer DD-Anweisung SYSTCPD in der JCL gesetzt werden.

### CONCHAR

Gibt das Befehlszeichen für die JES-Konsole an. Standardmäßig ist CONCHAR für JES2 auf CONCHAR=\$ und für JES3 auf CONCHAR=\* gesetzt. Falls Ihre Installation ein anderes Befehlszeichen definiert hat, geben Sie diesen Wert für CONCHAR an.

### SUBMITMETHOD=TSO

Übergabe mit Hilfe von TSO. Bei Verwendung der Standardeinstellung (SUBMITMETHOD=JES) werden Jobs direkt an JES übergeben. Bei Angabe von SUBMITMETHOD=TSO werden Jobs mit dem TSO-Befehl **SUBMIT** übergeben. Bei dieser Methode können TSO-Exits aufgerufen werden. Da sich diese Methode jedoch nachteilig auf die Leistung auswirkt, wird sie nicht empfohlen.

**Anmerkung:** Die einzigen gültigen Einstellungen sind TSO und JES.

**Anmerkung:** Wenn SUBMITMETHOD=TSO angegeben ist, muss TSO\_TEMPLATE ebenfalls definiert sein.

### TSO\_TEMPLATE=HLQ.SFEKSAMP(FEJTSO)

Wrapper-JCL für die Übergabe von Jobs mit Hilfe von TSO. Es gibt keinen Standardwert. Diese Anweisung referenziert den vollständig qualifizierten Namen der JCL, die als Wrapper für die TSO-Übergabe verwendet werden soll. Weitere Informationen finden Sie in der Beschreibung der Anweisung SUBMITMETHOD.

**Anmerkung:** Ein Beispiel-Wrapper-Job ist in HLQ.SFEKSAMP(FEJTSO) enthalten. Dieser Member stellt weitere Informationen zur erforderlichen Anpassung bereit.

**Anmerkung:** TSO\_TEMPLATE hat keine Auswirkung, wenn SUBMITMETHOD=TSO nicht ebenfalls angegeben ist.

**Anmerkung:** Die JESNAME-Definition ist nicht mehr erforderlich. JES Job Monitor erkennt ab Version 7.0 automatisch, ob Ihr Primär-JES JES2 oder JES3 ist.

---

## Start-JCL für JES Job Monitor anpassen

JES Job Monitor ist die Komponente für die gesamte JES-Konnektivität von Developer für System z. Dazu muss auf dem System ein Server aktiv sein (Benutzerjob oder STC). Erstellen Sie den JES Job Monitor Server gemäß den Standards an Ihrem Standort. Führen Sie dazu die Anpassungsschritte in HLQ.SFEKSAMP(FEJJJCL) aus.

**Anmerkung:** Sie sollten die Beispiel-JCL in eine neue Dateigruppe kopieren und diese Kopie anpassen, damit die JCL im Falle einer Wartung nicht überschrieben wird.

## Trace-Erstellung für JES Job Monitor

Falls Sie die Trace-Erstellung für JES Job Monitor zu Diagnosezwecken aktivieren müssen, fügen Sie zur PARM-Zeile -TV hinzu. Der Trace bringt Leistungseinbußen mit sich und sollte nur auf Anweisung des IBM Support Center durchgeführt werden.

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

Die Trace-Erstellung kann auch mit Konsolbefehlen gesteuert werden. Der erste Befehl in der folgenden Liste aktiviert den Trace für einen Job mit dem Namen JMON und der zweite inaktiviert den Trace.

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

## JES Job Monitor als STC ausführen

JES Job Monitor kann als Benutzerjob oder gestartete Task (STC) ausgeführt werden. Wenn Sie JMON als STC definieren möchten, müssen Sie die folgende Task implementieren:

1. Die JCL muss keine JOB-Karte enthalten (und sollte bevorzugt keine solche Karte enthalten). Die meisten STCs beginnen wie das Beispiel in Abb. 2 mit einer PROC-Karte.

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

Abbildung 2. JCL für JES Job Monitor

2. Die JCL muss in einer Systemprozedurenbibliothek enthalten sein (die IBM Standardbibliothek ist SYS1.PROCLIB).  
Zur leichteren Referenzierung sollte der Member-Name mit dem Namen der Prozedur übereinstimmen. (Im obigen Beispiel ist dies der Name JMON.)
3. STCs sollte eine dedizierte Benutzer-ID zugeordnet sein. Sie sollten diese Benutzer-ID aus Sicherheitsgründen schützen, indem Sie (in RACF) das Schlüsselwort NOPASSWORD definieren. Jeder RACF-Anmeldeversuch, für den ein Kennwort erforderlich ist, wird dann scheitern. Die Benutzer-ID wird jedoch nicht entzogen.

Verwenden Sie zum Erstellen einer solchen Benutzer-ID den folgenden Beispielbefehl. Die Angabe userid steht hier für die besprochene Benutzer-ID, groupid für die Gruppen-ID und uid für die UNIX-ID.

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

4. STCs müssen für die Sicherheitssoftware (z. B. für RACF) definiert werden. Eine STC kann auf verschiedene Weise definiert werden. Empfohlen wird die Verwendung der Klasse STARTED. Zum Definieren der Klasse STARTED würde Ihr Sicherheitsadministrator die folgenden RACF-Befehle absetzen:

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

Zum Hinzufügen von JES Job Monitor als STC sind die folgenden RACF-Befehle erforderlich. In diesem Befehlen ist jmon der Name des JCL-Members und userid die ID des Benutzers, dessen Berechtigungen verwendet werden.

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

**Anmerkung:** Wenn Sie das Schlüsselwort TRUSTED(YES) zum Feld STDATA hinzufügen [STDATA(USER(userid) TRUSTED(YES))], müssen Sie nicht die notwendigen individuellen Berechtigungen für die STC-Benutzer-ID definieren. Für anerkannte STCs übergeht RACF die Sicherheitsprüfung von Dateigruppen. Stellen Sie jedoch im Vorfeld sicher, dass diese Benutzer-ID nicht missbraucht werden kann, indem Sie sie wie oben beschrieben schützen.

Weitere Informationen zu gestarteten Tasks und zur Sicherheit enthält der *Security Server RACF Security Administrator's Guide (IBM Form SA22-7683)*.

- 5. Wenn die obigen Tasks abgeschlossen sind, kann JES Job Monitor als STC gestartet und gestoppt werden.

Der Systembediener kann auf der Konsole die folgenden Befehle absetzen. (In diesem Befehlen ist JMON der STC-Name von JES Job Monitor.)

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

Mit den entsprechenden Berechtigungen können Sie diese Befehle in SDSF eingeben. In diesem Fall müssen Sie den Befehlen allerdings einen Schrägstrich ( '/') voranstellen. Weitere Informationen zu Konsolbefehlen enthält die Veröffentlichung *MVS System Commands (IBM Form SA22-7627)*.

## Serverberechtigungen

Die dem JES Job Monitor Server zugeordnete Benutzer-ID benötigt die Zugriffsberechtigung READ für die Ladebibliothek HLQ.SFEKLOAD von Developer für System z.

## Start-JCL für JES Job Monitor prüfen

Starten Sie den Benutzerjob oder die STC, den bzw. die Sie mit den obigen Schritten definiert haben. Falls der Job mit dem Rückkehrcode 66 endet, ist HLQ.SFEKLOAD nicht für APF berechtigt.

---

## JES-Spool-Zugriff und Sicherheit

### Bedingter Spool-Zugriff

Benutzer können erst Operationen über JES Job Monitor ausführen, wenn sie berechtigt sind, auf die Klasse OPERCMDS zuzugreifen. Sie können diese Zugriffsberechtigung bedingt erteilen, so dass Benutzer nur Zugriff haben, wenn sie JES

Job Monitor verwenden. Für eine Überprüfung dieses bedingten Zugriffs muss die Klasse CONSOLE aktiv und auf der Konsole der Name JMON definiert sein. (JMON ist der einzige gültige Name.)

Ihr Sicherheitsadministrator müsste beispielsweise die folgenden RACF-Befehle absetzen:

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

Mit den folgenden RACF-Befehlen können Sie Benutzer für das ausschließliche Absetzen von JES2-Befehlen über JES Job Monitor berechtigen. Die Angabe id steht hier für die Benutzer- oder Gruppen-ID der Benutzer von Developer für System z.

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

**Achtung:**

Wenn Sie in Ihrer Sicherheitssoftware die JES-Befehle mit dem universellen Zugriffsrecht NONE definieren, kann sich das negativ auf andere Anwendungen und Operationen auswirken. Testen Sie eine solche Definition, bevor Sie sie auf einem Produktionssystem aktivieren.

## Verfügbare Befehle

JES Job Monitor ermöglicht Benutzern von Developer für System z keinen umfassenden JES-Spool-Zugriff. Es stehen nur die in Tabelle 6 aufgelisteten Befehle zur Verfügung. Die Befehle werden durch Auswahl der entsprechenden Option in der Clientmenüstruktur abgesetzt (keine Eingabeaufforderung). Durch die unten beschriebenen Techniken wird der Geltungsbereich der Befehle weiter eingeschränkt.

*Tabelle 6. Konsolbefehle für JES Job Monitor*

Befehl	JES2	JES3
Hold	\$Hjobid	*F,J=jobid,H
Release	\$Ajobid	*F,J=jobid,R
Cancel	\$Cjobid	*F,J=jobid,C
Purge	\$Cjobid,P	*F,J=jobid,C

**Anmerkung:** Wenn der Benutzer nicht der Eigner der Spool-Datei ist, ist nur eine Anzeige möglich. Alle in Tabelle 6 aufgelisteten Aktionen führen in dem Fall zu einer Nachricht "Sie haben keine Berechtigung für den Job JOBxxx".

Benutzer, die nicht berechtigt sind, diese Konsolbefehle auszuführen, können trotzdem Jobs übergeben und Jobausgaben lesen, wenn sie für den Zugriff auf Spool-Dateien berechtigt sind.

## Zugriff auf Spool-Dateien einschränken

Wenn Benutzer nur ihre eigenen JES-Spool-Jobs anzeigen können sollen, definieren Sie in der Konfigurationsdatei von JES Job Monitor (FEJJCNGF) die Anweisung `LIMIT_VIEW=USERID`. Falls Benutzer auf weitere Jobs zugreifen müssen, jedoch nicht auf alle Jobs, können Sie die Standardschutzfeatures für Spool-Dateien verwenden, z. B. die Klasse `JESSPOOL` in RACF.

Denken Sie beim Definieren weiterer Schutzmaßnahmen daran, dass JES Job Monitor für den Zugriff auf Spool-Dateien (die `SYSOUT`-Anwendungsprogrammierschnittstelle) `SAPI` verwendet. Damit ist impliziert, dass der Benutzer für Spool-Dateien (selbst zum Anzeigen) zumindest die Berechtigung `UPDATE` haben muss. Diese Voraussetzung gilt nicht unter `z/OS` ab Version 1.7 (für JES3 unter `z/OS` ab Version 1.8). Für Anzeigefunktionen ist die Berechtigung `READ` ausreichend.

Weitere Informationen zum Schutz von JES-Spool-Dateien finden Sie im *Security Server RACF Security Administrator's Guide (IBM Form SA22-7683)*.

---

## ELAXF\*-Prozeduren für ferne Build-Erstellung anpassen

Developer für System `z` stellt Beispiel-JCL-Prozeduren bereit, die für die JCL-Generierung, die Fernerstellung von Projekt-Builds und die ferne Syntaxprüfung von CICS-BMS-Masken, IMS-MFS-Anzeigen sowie von COBOL-, PL/I-, Assembler- und C/C++-Programmen verwendet werden können.

Diese Prozeduren ermöglichen Installationen, eigene Standards anzuwenden. Damit wird außerdem sichergestellt, dass die Entwickler dieselben Prozeduren mit denselben Compileroptionen und Compilerversionen verwenden.

SMP/E installiert diese Beispiel-JCL-Prozeduren in `HLQ.SFEKSAMP`. Wenn Sie diese Prozeduren verwenden möchten, gehen Sie wie folgt vor:

1. Kopieren Sie alle Prozeduren mit dem Namen `ELAXF*` in eine Prozedurenbibliothek des Systems (wie `SYS1.PROCLIB`), die für die Benutzer verfügbar ist.
2. Die kopierten Prozeduren müssen an die Namenskonventionen des Zielsystems angepasst werden. Die erforderliche Anpassung ist innerhalb der einzelnen JCL-Prozeduren dokumentiert.

Die zu kopierenden und anzupassenden Prozeduren sind in Tabelle 7 aufgelistet.

*Tabelle 7. ELAXF\*-Beispielprozeduren*

Member	Zweck
ELAXFADT	Beispielprozedur für die Assemblierung und das Debugging von High-Level-Assembler-Programmen
ELAXFASM	Beispielprozedur für die Assemblierung von High-Level-Assembler-Programmen
ELAXFBMS	Beispielprozedur für die Erstellung eines CICS-BMS-Objekts und des entsprechenden Copy-, Dsect- oder Include-Members
ELAXFCOC	Beispielprozedur für COBOL-Kompilierung, integrierte CICS-Umsetzung und integrierte DB2-Umsetzung
ELAXFCOP	Beispielprozeduren für die DB2-Vorverarbeitung von 'EXEC SQL'-Anweisungen, die in COBOL-Programme eingebettet sind
ELAXFCOT	Beispielprozeduren für die CICS-Umsetzung von 'EXEC CICS'-Anweisungen, die in COBOL-Programme eingebettet sind
ELAXFCPC	Beispielprozedur für C-Kompilierungen

Tabelle 7. ELAXF\*-Beispielprozeduren (Forts.)

Member	Zweck
ELAXFCPP	Beispielprozedur für C++-Kompilierungen
ELAXFGO	Beispielprozedur für den GO-Schritt
ELAXFLNK	Beispielprozedur für die Verknüpfung von C/C++-, COBOL-, PLI- und High-Level-Assembler-Programmen
ELAXFMFS	Beispielprozedur für die Erstellung von IMS-MFS-Anzeigen
ELAXFPLP	Beispielprozeduren für die DB2-Vorverarbeitung von 'EXEC SQL'-Anweisungen, die in PLI-Programme eingebettet sind
ELAXFPLT	Beispielprozeduren für die CICS-Umsetzung von 'EXEC CICS'-Anweisungen, die in PLI-Programme eingebettet sind
ELAXFPL1	Beispielprozedur für PL/I-Kompilierung, integrierte CICS-Umsetzung und integrierte DB2-Umsetzung
ELAXFUOP	Beispielprozedur für die Generierung des UOPT-Schritts beim Erstellen von Programmen, die in CICS- oder IMS-Subsystemen ausgeführt werden

Tabelle 8 enthält eine Prüfliste für die in den ELAXF\*-Prozeduren verwendeten High Level Qualifier der verschiedenen Produkte.

Tabelle 8. Prüfliste der High Level Qualifier in ELAXF\*

Produkt	Standard-HLQ	Wert
RD/z	FEK	
CICS	CICSTS31.CICS	
DB2	DSN810	
IMS	IMS	
COBOL	IGY.V3R4M0	
PL/I	IBMZ.V3R6M0	
C/C++	CBC	
LE	CEE	
LINKLIB des Systems	SYS1	
MACLIB des Systems	SYS1	

Wenn die ELAXF\*-Prozeduren nicht in eine Prozedurenbibliothek des Systems kopiert werden können, kopieren Sie sie in eine private Bibliothek, und fordern Sie die Benutzer von Developer für System z auf, zu den Jobmerkmalen auf dem Client (direkt nach der JOB-Karte) eine JCLLIB-Karte hinzuzufügen. Passen Sie nicht die Beispiel-JCL in der Installationsdateigruppe an, denn diese Member könnten bei einer Wartung ersetzt werden und Ihre Anpassungen löschen.

```
//MYJOB    JOB <Jobparameter>
//PROCS JCLLIB ORDER=(HLQ.SFEKSAMP)
```

**Anmerkung:** Bei der JCL-Generierung, der Fernerstellung von Projekt-Builds und der fernen Syntaxprüfung von einer Clientkomponente von Developer für System z aus wird vorausgesetzt, dass diese Prozeduren angepasst und für den Benutzer verfügbar sind.



Die Namen der Prozeduren und der einzelnen Prozedurschritte stimmen mit den Standardmerkmalen des Clients überein. Falls Sie den Namen einer Prozedur oder eines Prozedurschritts ändern möchten, sollten Sie auch die entsprechende Merkmaldatei des Clients aktualisieren. Das Ändern der Namen von Prozeduren oder Prozedurschritten ist nicht zu empfehlen.

**Anmerkung:** Zur Unterstützung des fernen Debugging von Assembler-, COBOL- und PL/I-Programmen müssen Sie das IBM Debug Tool bestellen, installieren und konfigurieren. Welche Version des Debug Tool für Ihre Version von Developer für System z erforderlich ist, können Sie der Veröffentlichung *Rational Developer für System z Hostplanung (IBM Form GI11-3123-00)* entnehmen. Die Installation und Anpassung dieses Produkts ist nicht in diesem Handbuch beschrieben.

---

## APPC-Transaktion für den TSO Commands Service definieren

Ab Version 7.1 ist das Definieren der APPC-Transaktion optional. Standardmäßig verwendet Developer für System z jetzt das SCLM Developer Toolkit für die Bereitstellung des TSO Commands Service.

Die entsprechende Konfiguration wird in `rsed.envvars` vorgenommen und ist im Abschnitt „RSE-Konfigurationsdatei `rsed.envvars` anpassen“ auf Seite 35 beschrieben.

**Anmerkung:** In Version 7.1 hat sich die von APPC zum Starten des TSO Commands Service verwendete Transaktionsprogramm-JCL geändert. Wenn Sie den TSO Commands Service früher für die Erfassung von ISPEXEC-Ausgaben definiert hatten, müssen Sie jetzt eine neue APPC-Transaktion definieren oder zur Zeile PARM das Schlüsselwort NESTMACS hinzufügen. Beispiel:

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

Der TSO Commands Service wird als das APPC-Transaktionsprogramm FEKFRSRV implementiert und muss aktiv sein, damit Developer-für-System-z-Verbindungen zwischen Host und Client hergestellt werden können. FEKFRSRV fungiert als ein Hostserver, der die von der Workstation über TCP/IP abgesetzten TSO-Befehle ausführt. Auf der Workstation ist APPC nicht erforderlich. Die Workstation kommuniziert über TCP/IP mit FEKFRSRV. Jede Workstation kann gleichzeitig eine aktive Verbindung zu mehreren Hosts haben.

**Anmerkung:** Falls Sie sich nicht mit APPC auskennen, lesen Sie Anhang F, „APPC konfigurieren“, auf Seite 123, bevor Sie mit den Informationen in diesem Abschnitt fortfahren.

## Vorbereitungen

- Vor dem Konfigurieren des TSO Commands Server wird vorausgesetzt, dass Sie die folgenden Tasks abgeschlossen haben. Die genannten Tasks sind in den angegebenen Handbüchern beschrieben.
  1. Installieren, konfigurieren und starten Sie VTAM auf Ihrem z/OS-System. Weitere Informationen hierzu finden Sie im *Communications Server Bookshelf (F1A1BK61)*.
  2. Installieren, konfigurieren und starten Sie TCP/IP auf Ihrem z/OS-System. Weitere Informationen hierzu finden Sie in Anhang C, „TCP/IP konfigurieren“, auf Seite 95 und im *Communications Server Bookshelf (F1A1BK61)*.



3. Konfigurieren und starten Sie APPC und das ASCH-Subsystem (APPC-Transaktions-Scheduler). Weitere Informationen hierzu finden Sie in Anhang F, „APPC konfigurieren“, auf Seite 123 und im *MVS Bookshelf (IEA2BK60)*.
- Mit der REXX in Abb. 3 können Sie APPC in ISPF-Anzeigen verwalten.

```
/* REXX – APPC-Verwaltung in ISPF-Anzeigen*/
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"
exit
```

Abbildung 3. REXX für APPC-ISPF-Anzeigen

**Anmerkung:** Mit diesem Tool können Sie die APPC-Transaktion inaktivieren. Die Transaktion ist unverändert vorhanden, akzeptiert dann jedoch keine Verbindungen mehr.

- Das Definieren der APPC-Transaktion erfordert Know-how zu verschiedenen Bereichen des Betriebssystems MVS. Gehen Sie die folgende Prüfliste mit erfahrenen Administratoren durch, bevor Sie Ihre Arbeit fortsetzen.

Tabelle 9. Prüfliste für APPC-Transaktion

Fachwissen	Erforderliche Informationen: <ul style="list-style-type: none"> <li>• Standardwert</li> <li>• Entsprechende Quelle</li> </ul>	Wert
APPC	Dateigruppenname von TPDATA <ul style="list-style-type: none"> <li>• Standardwert: SYS1.APPCTP</li> <li>• Der Wert ist in SYS1.PARMLIB(APPCPMxx) enthalten.</li> </ul>	
APPC	Zu verwendender Transaktionsname (möglicherweise nicht vorhanden) <ul style="list-style-type: none"> <li>• Standardwert: FEKFRSRV</li> <li>• Vorhandene Transaktionen können durch Auswahl von "TP Profile Administration" im ISPF-Menü 'APPC' abgefragt werden.</li> </ul>	
APPC	Zu verwendende APPC-Transaktionsklasse <ul style="list-style-type: none"> <li>• Standardwert: A</li> <li>• APPC-Klassen sind in SYS1.PARMLIB(ASCHPMxx) definiert.</li> </ul>	
WLM/SRM	TSO-Leistungsgruppe und -Domäne <ul style="list-style-type: none"> <li>• Kein IBM Standardwert (standortabhängig)</li> </ul>	
RACF	Jeder Benutzer von Developer für System z hat Zugriff auf ein OMVS-Segment. (Dies ist erforderlich.) <ul style="list-style-type: none"> <li>• Kein IBM Standardwert (standortabhängig)</li> <li>• Der TSO-RACF-Befehl <b>LU userid OMVS</b> zeigt ein vorhandenes persönliches OMVS-Segment an.</li> </ul>	

Tabelle 9. Prüfliste für APPC-Transaktion (Forts.)

Fachwissen	Erforderliche Informationen:	Wert
RACF	<ul style="list-style-type: none"> <li>• Standardwert</li> <li>• Entsprechende Quelle</li> </ul> <p>Jeder Benutzer von Developer für System z muss Lesezugriff (READ) auf HLQ.SFEKPROC (FEKFRSRV) haben.</p> <ul style="list-style-type: none"> <li>• Kein IBM Standardwert (standortabhängig)</li> <li>• Der TSO-RACF-Befehl <b>LD AUTHUSER DATASET('HLQ.SFEKPROC.**')</b> zeigt Benutzer und Gruppen mit der Zugriffsebene für die Dateigruppen dieses Dateigruppenprofils an.</li> </ul>	

Weitere Informationen zum WLM/SRM-Management finden Sie in der Veröffentlichung *MVS Planning Workload Management (IBM Form SA22-7602)*. Weitere Informationen zu OMVS-Segmenten und Profilen für Dateigruppenschutz enthält der *Security Server RACF Security Administrator's Guide (IBM Form SA22-7683)*.

**Anmerkung:** Die verwendete APPC-Transaktionsklasse muss genug APPC-Initiatoren haben, damit für jeden Benutzer von Developer für System z ein Initiator verfügbar ist.

**Anmerkung:** Die APPC-Transaktion verwendet die REXX-Exec FEKFRSRV aus HLQ.SFEKPROC. Ändern Sie diese Position nicht, wenn Sie möchten, dass die SMP/E-Wartung automatisch aktiviert wird.

## Implementierung

Als Systemprogrammierer oder APPC-Administrator müssen Sie die Commands-Funktion wie folgt konfigurieren:

1. Definieren Sie die Planungsinformationen (Klasse) für den APPC-Transaktions-Scheduler, wenn Sie keine externe Transaktionsklasse verwenden. Nehmen Sie eine Definition für die Klasse in SYS1.PARMLIB(ASCHPMxx) auf, damit sie vom Transaktionsprogramm FEKFRSRV verwendet wird. Diese Klasse wird in der Beispiel-JCL HLQ.SFEKSAMP(FEKAPPCC) verwendet. Die Klasse in FEKAPPCC muss deshalb mit der in SYS1.PARMLIB(ASCHPMxx) definierten Klasse übereinstimmen.

Beispiel:

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

**Anmerkung:** Der TSO Commands Service erfordert außerdem die Angabe der Standardspezifikationen in den Abschnitten OPTIONS und TPDEFAULT von SYS1.PARMLIB(ASCHPMxx). Weitere Informationen hierzu enthält Anhang F, „APPC konfigurieren“, auf Seite 123.

2. Definieren Sie die APPC-Transaktion, die als Befehlsserver verwendet werden soll. Zum Definieren dieser Transaktion können Sie die Beispiel-JCL HLQ.SFEKSAMP(FEKAPPCC) verwenden. Anweisungen zum Anpassen dieser JCL finden Sie direkt in der JCL. Zum Anzeigen der Transaktion steht die Beispiel-JCL HLQ.SFEKSAMP(FEKAPPCL) und zum Löschen der Transaktion HLQ.SFEKSAMP(FEKAPPCX) zur Verfügung.

**Anmerkung:** Falls Sie den Namen des Transaktionsprogramms (standardmäßig FEKFRSRV) geändert haben, müssen Sie den neuen Namen

\_FEKFSCMD\_TP\_NAME\_ in rsed.envvars zuordnen. Lesen Sie hierzu die Beschreibung im Abschnitt „RSE-Konfigurationsdatei rsed.envvars anpassen“ auf Seite 35.

3. Steuern Sie die Zuteilungspriorität des Transaktionsprogramms HLQ.SFEKPROC (FEKFRSRV), indem Sie FEKFRSRV in WLM (Workload Manager) eine Domäne und eine Leistungsgruppe zuordnen. Da FEKFRSRV TSO-Befehle absetzt, sollte das Transaktionsprogramm einer TSO-Leistungsgruppe zugeordnet werden.
4. Definieren Sie ein Standard-OMVS-Segment für das System oder ein individuelles Segment für jeden Benutzer, der die Fernbearbeitung, die Fernkompilierung oder das Fern-Debugging verwenden muss.
5. Erteilen Sie Benutzern von Developer für System z Lesezugriff (READ) auf den TSO Commands Server HLQ.SFEKPROC (FEKFSERV).

**Anmerkung:** Die Konfiguration wird später zusammen mit RSE überprüft, weil RSE die TCP/IP-Verbindung zum TSO Commands Server implementiert.

## ELAXM\*-Member für gespeicherte DB2-Prozedur anpassen (optional)

Für die Integration der gespeicherten DB2-Beispielprozedur (Stored Procedure Builder für PL/I und COBOL) in Ihr System sind die folgenden Anpassungsschritte erforderlich. Für die Ausführung dieser Schritte benötigen Sie die Unterstützung eines WLM-Administrators und eines DB2-Administrators.

Nach der Ausführung von SMP/E enthalten die Beispielbibliothek HLQ.SFEKSAMP und die Prozedurenbibliothek HLQ.SFEKPROC die in Tabelle 10 aufgelisteten Member der gespeicherten DB2-Prozedur.

*Tabelle 10. ELAXM\*-Beispiel-Member der gespeicherten DB2-Prozedur*

Member	Zweck
HLQ.SFEKSAMP(ELAXMJCL)	Beispiel-JCL zum Definieren des Stored Procedure Builder für COBOL und PL/I für DB2
HLQ.SFEKSAMP(ELAXMSAM)	Beispiel-JCL-Prozedur des WLM-Adressbereichs für den Stored Procedure Builder für PL/I und COBOL
HLQ.SFEKPROC(ELAXMREX)	REXX-Code für den Stored Procedure Builder für PL/I und COBOL

**Anmerkung:** Die gespeicherte DB2-Prozedur verwendet die REXX-Exec ELAXMREX in HLQ.SFEKPROC. Ändern Sie diese Position nicht, wenn Sie möchten, dass die SMP/E-Wartung automatisch aktiviert wird.

**Anmerkung:** Falls Sie den Member ELAXMSAM oder ELAXMREX umbenennen möchten, lesen Sie Anhang A, „Mehrere Instanzen von Developer für System z ausführen“, auf Seite 75.

1. Kopieren Sie ELAXMSAM in eine Prozedurenbibliothek (wie SYS1.PROCLIB), die Benutzern der gespeicherten DB2-Prozedur zur Verfügung steht, und passen Sie die JCL wie in den Kommentaren innerhalb der JCL angegeben an. Vergewissern Sie sich, dass die DD-Karte SYSEXEC auf die Bibliothek zeigt, die den Member ELAXMREX enthält (standardmäßig HLQ.SFEKPROC).
2. Ordnen Sie der JCL-Prozedur des WLM-Adressbereichs für den Stored Procedure Builder für PL/I und COBOL in den WLM-Anzeigen eine Anwendungsumgebung zu. Informationen hierzu finden in der Veröffentlichung *MVS Planning Workload Management (IBM Form SA22-7602)*.

**Anmerkung:** Sie können in WLM eine neue Umgebung für den Stored Procedure Builder für PL/I und COBOL erstellen oder die erforderlichen Definitionen zu einer vorhandenen Umgebung hinzufügen.

3. Kopieren Sie ELAXMJCL in eine private JCL-Bibliothek, passen Sie die Kopie wie in den Kommentaren beschrieben an, und bitten Sie einen DB2-Administrator, den Job zu übergeben. Stellen Sie sicher, dass die Klausel WLM ENVIRONMENT der Anweisung CREATE PROCEDURE den Namen der WLM-Umgebungsprozedur angibt, die für den Stored Procedure Builder für PL/I und COBOL definiert wurde (standardmäßig ELAXMSAM).

---

## Unterstützung bidirektionaler Sprachen für CICS anpassen (optional)

Die Komponente Enterprise Service Tools (EST) von Developer für System z unterstützt verschiedene Formate für arabische und hebräische Schnittstellennachrichten und die bidirektionale Datendarstellung und -bearbeitung in allen Editoren und Ansichten. In Terminalanwendungen werden Anzeigen von links nach rechts und von rechts nach links sowie numerische Felder und Felder mit entgegengesetzter Anzeigearrichtung unterstützt.

Zu den zusätzlichen bidirektionalen Features und Funktionen gehören unter anderem:

- Der EST-Serviceanforderer gibt dynamisch bidirektionale Attribute von Schnittstellennachrichten an.
- Die bidirektionale Datenverarbeitung in Service-Flows basiert auf bidirektionalen Attributen (Texttyp, Textausrichtung, numerische Ersetzung und symmetrische Ersetzung). Diese Attribute können in verschiedenen Stadien der Erstellung von Schnittstellen- und Terminal-Flows angegeben werden.
- Der von EST generierte Laufzeitcode umfasst die Umsetzung von Daten in Feldern von Nachrichten mit verschiedenen bidirektionalen Attributen.

**Achtung:** Das Lademodul hat sich gegenüber früheren Releases (Vorgängerreleases von 7.0) geändert (Name und Codierung). Falls Sie ein früheres BIDI-Release aktiviert haben, müssen Sie die alten Lademodule aus der CICS-RPL-Kette entfernen. Die Standardpositionen sind:

- FEJ.SFEJDLL(FEJBDTRN)
- FEJ.SFEJLMD(FEJBDTRE) (optionale Nicht-PDSE-Version von FEJBDTRN für Hebräisch)

Für die Ausführung der folgenden Tasks benötigen Sie die Unterstützung Ihres CICS-Administrators:

1. Stellen Sie das Programm HLQ.SFEKLOAD(FEJBDTRX) in die CICS-RPL-Kette (DD-Anweisung DFHRPL). Zu diesem Zweck sollten Sie die Installationsdateigruppe zur Kette hinzufügen, damit angewendete Wartungen automatisch für CICS verfügbar sind.

Die bidirektionalen Umsetzungen von EST werden vom Modul FEJBDTRX in der Umgebung CICS Service Flow Runtime (SFR) ausgeführt. Das Modul FEJBDTRN wird aufgerufen, wenn in dem von EST generierten Code bidirektionale Umsetzungen erforderlich sind (d. h., wenn die Felder von Nachrichten mit verschiedenen bidirektionalen Attributen einander zugeordnet werden).

**Anmerkung:** Wenn Sie nicht das Installationsverzeichnis zur Kette hinzufügen, sondern das Modul FEJBDTRX in eine neue/vorhandene Dateigruppe kopieren, beachten Sie, dass dieses Modul eine DLL ist und in einer PDSE-Bibliothek enthalten sein MUSS.

**Anmerkung:** In Version 7.1 wurde das Programm HLQ.SFEKDLL(FEJBDTRX) in die neue Ladebibliothek HLQ.SFEKLOAD(FEJBDTRX) verschoben.

2. Bitten Sie den CICS-Administrator, `autoinstall` auf `autoactive` zu setzen.  
Wenn `autoinstall` auf `autoinactive` gesetzt ist, definieren Sie das Programm FEJBDTRX mit dem Befehl CEDA für CICS. Beispiel:  

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

Von EST generierter Code kann die BIDI-Konvertierung auch in anderen Umgebungen als CICS SFR unterstützen (z. B. in Batch-Anwendungen). Sie können die EST-Generatoren veranlassen, alle Aufrufe bidirektionaler Umsetzungsroutinen aufzunehmen, indem Sie in den EST-Generierungsassistenten die entsprechenden BIDI-Konvertierungsattribute angeben und die generierten Programme mit der entsprechenden Bibliothek für bidirektionale Umsetzung (HLQ.SFEKLOAD) verknüpfen.

**Achtung:** Der mit früheren Releases (Vorgängerreleases von Version 7.0) erstellte BIDI-Code muss ERNEUT KOMPILIERT werden, damit er das neue Modul FEJBDTRX verwendet.

---

## Application Deployment Manager (ADM) anpassen (optional)

Application Deployment Manager (ADM) bietet eine einheitliche Deployment-Strategie und Deployment-API für alle Komponenten von Rational Developer für System z.

Darüber hinaus stellt ADM einen CRD-Client (CICS Resource Definition) und einen hostbasierten Server bereit, die Folgendes ermöglichen:

1. Anwendungsentwickler können CICS-Ressourcen begrenzt, kontrolliert und geschützt definieren.
  - Der CICS-Administrator stellt Standardwerte für CICS-Ressourcendefinitionen bereit, die im CRD-Server-Repository auf dem Host gespeichert werden.
  - Der CICS-Administrator steuert (mit 'update', 'protect', 'hidden') die Möglichkeit der Aktualisierung von Ressourcenattributen.
  - CRD ist auf eine kleine Gruppe von Ressourcen begrenzt, die in der Regel von Anwendungsentwicklern benötigt werden. Dazu gehören DB2Tran, Doctemplate, File, Mapset, Processtype, Program, TDQ und Transaction.
  - Die Berechtigung zur Erstellung von CICS-Ressourcendefinitionen wird von RACF oder einem anderen externen Sicherheitsmanager gesteuert.
2. Vermeiden Sie während der CICS-Entwicklung den Zugriff auf nicht autorisierte oder falsche VSAM-Dateigruppen, indem Sie dem CICS-Administrator die Kontrolle über das Attribut für physische Dateigruppennamen in Dateidefinitionen überlassen. Diese Bindungsinformation wird im CRD-Repository auf dem Host gespeichert.
3. Verschiedene Unterstützungsoptionen für die CRD-Serverentwicklung
  - Neue Kopie für Programme und Maskengruppen
  - Auflistung von CICS-Regionen
  - DFHRPL-Liste
4. Verschiedene Unterstützungsoptionen für die CRD-Server-Web-Service-Entwicklung
  - Ausführung eines Pipelinescans für die automatische Installation von URI-MAP- und WEBSERVICE-Definitionen
  - Bereitstellung von Listen mit Pipeline- und WSBind-Pickup-Verzeichnissen

- Bereitstellung einer Liste mit WSDL-Dateiverzeichnissen
- Bereitstellung einer Liste mit Endpunkt-URIs

Developer für System z stellt drei Transaktionen bereit, die der CRD-Server beim Definieren und Abfragen von CICS-Ressourcen verwendet. Für standortspezifische Anpassungen steht COBOL-Beispielquellcode zur Verfügung.

#### **ADMD**

Für Anforderungen, die Standardeinstellungen für Web-Service- und CICS-Ressourcen festlegen. Diese Transaktion ist normalerweise für CICS-Administratoren bestimmt.

**ADMI** Für Anforderungen, die CICS-Ressourcen definieren, installieren oder deinstallieren.

#### **ADMR**

Für alle anderen Anforderungen, die CICS-Umgebungsinformationen oder -Ressourceninformationen abrufen.

Weitere Informationen zu ADM enthält der *Rational Developer für System z Application Deployment Manager User's Guide* (IBM Form SC23-7661).

Die folgenden Anpassungsschritte sind zum Aktivieren des CRD-Servers erforderlich.

**Anmerkung:** Für die Ausführung einiger der nachfolgenden Tasks benötigen Sie die Unterstützung Ihres CICS-Administrators.

Wenn Sie mit einer früheren Version des CRD-Servers arbeiten, sollten Sie die zugehörigen Anpassungsdateien speichern, bevor Sie Version 7.1.1 installieren. Lesen Sie hierzu den Abschnitt „Bereits konfigurierte Dateien sichern“ auf Seite 11.

Kopieren Sie die anzupassenden Member aus dem Installationsverzeichnis in eine persönliche Bibliothek, und passen Sie die Kopien an, um das Überschreiben der Member bei einer Wartung zu verhindern:

- HLQ.SFEKSAMP(ADNVSAM)
- HLQ.SFEKSAMP(ADNPCCSD)

optional (Pipelinesnachrichten-Handler)

- HLQ.SFEKSAMP(ADNCMSGH)
- HLQ.SFEKSAMP(ADNSMSGH)

optional (CSD-Update für nicht primäre Regionen)

- HLQ.SFEKSAMP(ADNARCSD)

## **CRD-Repository**

Passen Sie den Job ADNVSAM an, und übergeben Sie ihn, um die VSAM-Datei für das CRD-Server-Repository anzulegen und zu initialisieren. Anpassungsanweisungen finden Sie in der in ADNVSAM enthaltenen Dokumentation.

Sie sollten für jede primäre CICS-Verbindungsregion ein gesondertes Repository erstellen. Eine gemeinsame Nutzung des Repositories impliziert, dass alle zugehörigen CICS-Regionen dieselben im Repository gespeicherten Werte verwenden und dass mehrere Adressbereiche in die VSAM schreiben. Hierfür ist eine korrekte Konfiguration erforderlich.



**Anmerkung:** Sofern Sie keine anders lautende Benachrichtigung erhalten, können Sie Ihr aktuelles CRD-Server-Repository (mit Ihren angepassten Werten) für alle Releases von Developer für System z wiederverwenden.

## Primäre CICS-Verbindungsregion

Der CRD-Server muss für die primäre Verbindungsregion definiert werden. Diese Region verarbeitet die Web-Service-Anforderungen von Developer für System z.

- Stellen Sie die ADM-Lademodule HLQ.SFEKLOAD(ADNCRDS) und HLQ.SFEKLOAD(ADNCRDR) in die CICS-RPL-Kette (DD-Anweisung DFHRPL) der primären CICS-Verbindungsregion. Zu diesem Zweck sollten Sie die Installationsdateigruppe zur Kette hinzufügen, damit angewendete Wartungen automatisch für CICS verfügbar sind.
- Passen Sie den Job ADNPCCSD an, und übergeben Sie ihn, um die CICS-Systemdefinition (CSD) für die primäre CICS-Verbindungsregion zu aktualisieren. Anpassungsanweisungen finden Sie in der in ADNPCCSD enthaltenen Dokumentation.
- Installieren Sie die ADM-Gruppe für diese Region mit dem entsprechenden CEDA-Befehl. Beispiel:  
`CEDA INSTALL GROUP(ADNPCRGP)`

## Pipelinesnachrichten-Handler

Der Pipelinesnachrichten-Handler (ADNSMSGH) wird für die Sicherheit verwendet. Er verarbeitet die Benutzer-ID und das Kennwort im SOAP-Header. ADNSMSGH wird von der Pipelinekonfigurationsdatei referenziert und muss deshalb in die CICS-RPL-Kette gestellt werden. ADNSMSGH wird auch verwendet, um die Transaktions-ID - je nach angeforderter Operation - auf ADMD, ADMR oder ADMI zu setzen. Bei Bedarf können Sie ADNSMSGH für die Verwendung anderer Transaktions-IDs anpassen.

Standardmodul verwenden:

- Stellen Sie das Lademodul HLQ.SFEKLOAD(ADNSMSGH) in die CICS-RPL-Kette (DD-Anweisung DFHRPL) der primären CICS-Verbindungsregion. Zu diesem Zweck sollten Sie die Installationsdateigruppe zur Kette hinzufügen, damit angewendete Wartungen automatisch für CICS verfügbar sind.

ADNSMSGH anpassen:

- Passen Sie den ADNSMSGH-Beispiel Quellcode (COBOL) für den Pipelinesnachrichten-Handler an.
- Passen Sie den Job ADNCMSGH an, und übergeben Sie ihn, um den angepassten ADNSMSGH-Quellcode zu kompilieren. Anpassungsanweisungen finden Sie in der in ADNCMSGH enthaltenen Dokumentation.
- Stellen Sie das resultierende Lademodul ADNSMSGH in die CICS-RPL-Kette (DD-Anweisung DFHRPL) der primären CICS-Verbindungsregion.

**Anmerkung:** Stellen Sie sicher, dass das angepasste Lademodul ADNSMSGH vor allen Verweisen auf HLQ.SFEKLOAD angegeben ist, da andernfalls das Standardmodul verwendet wird.

## Nicht primäre CICS-Verbindungsregionen (optional)

Der CRD-Server kann auch mit nicht primären Verbindungsregionen verwendet werden. Dabei handelt es sich in der Regel um AOR-Regionen (Application Owning Regions).

**Anmerkung:** Sie müssen diese Schritte nicht ausführen, wenn Ihre CICS-Umgebung mit CICSplex SM verwaltet wird.

- Stellen Sie das ADM-Lademodul HLQ.SFEKLOAD(ADNCRDS) in die CICS-RPL-Kette (DD-Anweisung DFHRPL) dieser nicht primären Verbindungsregionen. Zu diesem Zweck sollten Sie die Installationsdateigruppe zur Kette hinzufügen, damit angewendete Wartungen automatisch für CICS verfügbar sind.
- Passen Sie den Job ADNARCSD an, und übergeben Sie ihn, um CSD für andere, nicht primäre Verbindungsregionen zu aktualisieren. Anpassungsanweisungen finden Sie in der in ADNARCSD enthaltenen Dokumentation.
- Installieren Sie die ADM-Gruppe für diese Regionen mit dem entsprechenden CEDA-Befehl. Beispiel:  
CEDA INSTALL GROUP(ADNARRGP)



---

## Kapitel 4. z/OS-UNIX-Komponenten von Developer für System z aktivieren

Wenn Sie mit einer früheren Version von Developer für System z arbeiten, sollten Sie die zugehörigen Anpassungsdateien speichern, bevor Sie Version 7.1.1 installieren. Lesen Sie hierzu den Abschnitt „Bereits konfigurierte Dateien sichern“ auf Seite 11.

Falls Sie sich nicht mit z/OS UNIX auskennen, sollten Sie einen erfahrenen z/OS-UNIX-Administrator oder anderen UNIX-Administrator bitten, Sie bei der Ausführung der in diesem Kapitel beschriebenen Tasks zu unterstützen.

Die für die aufgeführten Tasks erforderlichen z/OS-UNIX-Befehle werden hier kurz für Sie beschrieben. Weitere Informationen zu diesen Befehlen entnehmen Sie bitte der Veröffentlichung *UNIX System Services Command Reference (IBM Form SA22-7802)*, sofern nichts anderes angegeben ist.

Für die nachfolgenden Tasks wird vorausgesetzt, dass Sie aktivierter zOS-UNIX-Benutzer sind. Zum Aktivieren können Sie den TSO-Befehl **OMVS** absetzen. Mit dem Befehl **exit** können Sie zu TSO zurückkehren.

MVS bietet mit dem Befehl **OEDIT** die Möglichkeit, z/OS-UNIX-Dateien mit ISPF zu bearbeiten. Sie können diesen Befehl in TSO und OMVS verwenden.

Bei den meisten z/OS-UNIX-Dateien ist der Schreibzugriff auf den Eigner der Datei beschränkt. Es gibt mehrere Möglichkeiten, diese Einschränkung zu umgehen.

- UID 0

Für Benutzer-IDs von Personen wird dies nicht empfohlen, weil es keine z/OS-UNIX-bezogenen Einschränkungen gibt.

- Zugriffsberechtigung READ für das Profil BPX.SUPERUSER in der Klasse FACILITY  
Über den Befehl **su** kann der Benutzer ein Benutzer mit der UID 0 werden. Dies ist die empfohlene Konfiguration.

- Zugriffsberechtigung UPDATE für das Profil SUPERUSER.FILESYS in der Klasse UNIXPRIV

Der Benutzer hat Lese- und Schreibrechte für alle Dateien und kann alle Verzeichnisse lesen oder durchsuchen. Die Zugriffsberechtigung CONTROL (oder eine höhere Zugriffsberechtigung) für dieses Sicherheitsprofil fügt zur Liste der Berechtigungen den Schreibzugriff für alle Verzeichnisse hinzu.

Wenn Sie mehr über die z/OS-UNIX-Sicherheit erfahren möchten, lesen Sie die Veröffentlichung *UNIX System Services Planning (IBM Form GA22-7800)*.

Alle Pfadangaben in diesem Kapitel, die `/usr/lpp/wd4z/` enthalten, beziehen sich auf den während der Installation von Developer für System z verwendeten Pfad. Der Standardpfad ist `/usr/lpp/wd4z/`; an Ihrem Standort kann jedoch ein anderer Pfad verwendet worden sein.

**Anmerkung:** Developer für System z ist bei der Initialisierung davon abhängig, dass TCP/IP mit dem richtigen Hostnamen konfiguriert ist. Dies impliziert, dass die verschiedenen TCP/IP- und Resolver-Konfigu-

rationsdateien ordnungsgemäß definiert sein müssen. Weitere Informationen zur erforderlichen Anpassung enthält Anhang C, „TCP/IP konfigurieren“, auf Seite 95.

Sie können Ihre TCP/IP-Konfiguration mit dem TSO-Befehl **HOME-TEST** testen. Weitere Informationen zu diesem Befehl enthält die Veröffentlichung *Communications Server IP System Administrator's Commands* (IBM Form SC31-8781).

**Anmerkung:** Developer für System z ist bei der Einrichtung der Verbindung zwischen Client und Host auf INETD angewiesen. Weitere Informationen zu INETD enthält Anhang D, „INETD konfigurieren“, auf Seite 103.

**Anmerkung:** Ferne (hostbasierte) Aktionen für z/OS-UNIX-Unterprojekte erfordern, dass auf dem Host REXEC oder SSH aktiv ist.

**Anmerkung:** Die 31-Bit-Java-Version ist erforderlich, und alle z/OS-UNIX-Benutzer müssen die Zugriffsrechte EXECUTE und READ für die Java-HFS-Verzeichnisse haben.

**Achtung:** Die 64-Bit-Java-Version wird NICHT unterstützt.

Die folgenden Veröffentlichungen enthalten hilfreiche Informationen zu z/OS UNIX:

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

---

## Konfigurationsdatei `rsed.envvars` in einem anderen Verzeichnis speichern

Sie sollten die Datei `/usr/lpp/wd4z/rse/lib/rsed.envvars` in ein neues Verzeichnis kopieren (z. B. in `/etc/wd4z/`) und die Kopie anpassen, damit die Datei im Falle einer Wartung nicht überschrieben wird. Sie müssen dann aber auch die folgenden Dateien aus dem Installationsverzeichnis (standardmäßig `/usr/lpp/wd4z/rse/lib/`) an die neue Position kopieren:

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

**Anmerkung:** Obwohl die Dateien `*.zseries` nicht angepasst werden müssen, ist es wichtig, dass Sie die früheren Versionen durch die aktuellen Versionen ersetzen. So ist sichergestellt, dass sie mit der aktuellen Datei `rsed.envvars` synchron sind.

Die in Tabelle 11 auf Seite 35 aufgelisteten Dateien müssen auch kopiert werden, wenn Sie planen, die optionalen Features zu verwenden, zu denen diese Dateien gehören:

Tabelle 11. Optionale Konfigurationsdateien

Datei	Funktion
projectcfg.properties	Hostbasierte Projekte  Lesen Sie hierzu den Abschnitt „Konfiguration von Hostprojekten in projectcfg.properties anpassen (optional)“ auf Seite 54.
FMIEXT.properties	File Manager Integration  Lesen Sie hierzu den Abschnitt „File Manager Integration in FMIEXT.properties anpassen (optional)“ auf Seite 55.
CRASRV.properties	CARMA  Lesen Sie hierzu Kapitel 5, „IBM Common Access Repository Manager (CARMA) aktivieren (optional)“, auf Seite 57.

Die Beispielbefehle

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

erstellen ein Verzeichnis mit dem Namen `/etc/wd4z` und kopieren `rsed.envvars` aus dem aktuellen Verzeichnis in das Verzeichnis `/etc/wd4z`. Wiederholen Sie den Kopierbefehl für die verbleibenden Dateien.

Das Ergebnis des Kopiervorgangs können Sie mit dem Befehl `ls /etc/wd4z` überprüfen. Die Ausgabe dieses Befehls sollte in etwa wie die folgende aussehen, wobei \$ die z/OS UNIX-Eingabeaufforderung ist:

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

**Anmerkung:** Falls Sie alle Dateien von Developer für System z in demselben (privaten) HFS behalten, die Konfigurationsdateien aber trotzdem in das Verzeichnis `/etc/wd4z` stellen möchten, können Sie das Problem mit symbolischen Verbindungen lösen. Die folgenden Beispielbefehle erstellen im Installations-HFS ein neues Verzeichnis (`/usr/lpp/wd4z/rse/lib/cust`) und definieren eine symbolische Verbindung (`/etc/wd4z`) zu diesem Verzeichnis:

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

## RSE-Konfigurationsdatei `rsed.envvars` anpassen

Alle Clientverbindungs Methoden von Developer für System z verwenden die in der Datei `rsed.envvars` gesetzten Variablen. Standardmäßig befindet sich die Datei im Installationsverzeichnis `/usr/lpp/wd4z/rse/lib/`. Sie kann aber auch wie im vorherigen Schritt in ein anderes Verzeichnis kopiert worden sein. Die bereitgestellte Beispieldatei enthält die in Abb. 4 auf Seite 36 aufgelisteten Anweisungen. Kommentarzeilen beginnen mit einem Nummernzeichen (#).

```

#=====
# (1) erforderliche Anpassungen
JAVA_HOME=/usr/lpp/java/J1.4
RSE_HOME=/usr/lpp/wd4z
TZ=EST5EDT
LANG=C
PATH=/bin:/usr/sbin:.
_RSE_CLASS_OPTS=""
_RSE_JAVAOPTS=""
#_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DTSO_SERVER=APPC"
#=====
# (2) erforderliche Anpassungen, wenn SCLMDT verwendet wird
_CMDSERV_BASE_HOME=/usr/lpp/SCLMDT
_CMDSERV_BASE_LOAD=BWB.SBWBLOAD
_CMDSERV_CONF_HOME=/etc/SCLMDT
_CMDSERV_WORK_HOME=/var/SCLMDT
STEPLIB=NONE
#STEPLIB=$_CMDSERV_BASE_LOAD
_RSE_CMDSERV_OPTS=""
#=====
# (3) optionale Anpassungen
#_RSE_PORTRANGE=8108-8118
#_FEKFLOCK_USERID=userid
#_FEKFLOCK_JOBNAME=job_name
#_FEKFSCMD_TP_NAME=tp_name
#_FEKFSCMD_PARTNER_LU=lu_name
#=====
# (4) nur auf Anweisung des IBM Support Center ändern
RSE_LIB=$RSE_HOME/rse/lib
ICU_LIB=$RSE_HOME/icuc/lib
_CEE_RUNOPTS="ALL31(ON) HEAP(32M,32K,ANYWHERE,KEEP,,) TRAP(ON)"
_CEE_DMPTARG=$HOME/.eclipse/RSE/$RSE_USER_ID
_BPX_SHAREAS=YES
_BPX_SPAWN_SCRIPT=YES
PATH=$JAVA_HOME/bin:$RSE_LIB:$_CMDSERV_BASE_HOME/bin:$PATH
LIBPATH=$JAVA_HOME/bin:$JAVA_HOME/bin/classic:$ICU_LIB:$RSE_LIB:.
CLASSPATH=$RSE_LIB:$RSE_LIB/dstore_core.jar:$RSE_LIB/clientserver.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/dstore_extra_server.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/dstore_miners.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/universalminers.jar:$RSE_LIB/mvsminers.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/carma.jar:$RSE_LIB/luceneminer.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/mvsluceneminer.jar:$RSE_LIB/cdzminer.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/mvscdzminer.jar:$RSE_LIB/jesminers.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/FAMiner.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/mvsutil.jar:$RSE_LIB/jesutils.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/lucene-1.4.3.jar:$RSE_LIB/cdtparser.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/wdzBidi.jar:$RSE_LIB/fmiExtensions.jar
CLASSPATH=.:$CLASSPATH
_RSE_CMDSERV_OPTS="$&SESSION=SPAWN$_RSE_CMDSERV_OPTS"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DSCLMDT_OPTS='$_RSE_CMDSERV_OPTS'"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DA_PLUGIN_PATH=$RSE_LIB"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Xbootclasspath/p:$RSE_LIB/bidiTools.jar"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Dcom.ibm.cacheLocalHost=true"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -showversion"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Duser.home=$HOME"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Dclient.username=$RSE_USER_ID"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS $_RSE_CLASS_OPTS"
_RSE_SERVER_CLASS=com.ibm.etools.systems.dstore.core.server.Server
_RSE_SERVER_TIMEOUT=120000
#=====
# (5) zusätzliche Umgebungsvariablen

```

Abbildung 4. RSE-Konfigurationsdatei *rsed.envvars*

Folgende Definitionen sind erforderlich:

#### **JAVA\_HOME**

Java-Home-Verzeichnis. Die Standardeinstellung ist /usr/lpp/java/J1.4. Passen Sie das Verzeichnis an Ihre Java-Installation an.

#### **RSE\_HOME**

RSE-Ausgangsverzeichnis. Die Standardeinstellung ist /usr/lpp/wd4z. Passen Sie das Verzeichnis an Ihre Installation von Developer für System z an.

#### **TZ**

Zeitzone-selector. Die Standardeinstellung ist EST5EDT. Die Standardzeitzone ist UTC + 5 Stunden (Eastern Standard Time mit Sommerzeit). Passen Sie diesen Wert an Ihre Zeitzone an. Zusätzliche Informationen hierzu finden Sie in der Veröffentlichung *UNIX System Services File System Interface Reference (IBM Form SA22-7808)*.

#### **LANG**

Gibt den Namen der Standardländereinstellung an. Der Standardwert ist C. C gibt die POSIX-Ländereinstellung und Ja\_JP die japanische Ländereinstellung an. Passen Sie den Wert an Ihre Ländereinstellung an.

#### **PATH**

Befehlspfad. Die Standardeinstellung ist /bin:/usr/sbin:.. Bei Bedarf können Sie diesen Pfad ändern.

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

Zusätzliche Java-Optionen für gemeinsame Klassennutzung. Die Standardeinstellung ist "". Weitere Informationen zu dieser Definition enthält der Abschnitt „Zusätzliche Java-Startparameter mit \_RSE\_\*OPTS definieren (optional)“ auf Seite 41.

#### **\_RSE\_JAVAOPTS**

Zusätzliche RSE-spezifische Java-Optionen. Die Standardeinstellung ist "". Weitere Informationen zu dieser Definition enthält der Abschnitt „Zusätzliche Java-Startparameter mit \_RSE\_\*OPTS definieren (optional)“ auf Seite 41.

Standardmäßig verwendet Developer für System z das SCLM Developer Toolkit für den TSO Commands Service. APPC wird verwendet, wenn die folgende \_RSE\_JAVAOPTS-Option nicht auf Kommentar gesetzt ist:

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

**Anmerkung:** Beide Methoden für den TSO Commands Service erfordern mehr Anpassungen als die in rsed.envvars. Die für die APPC-Konfiguration notwendigen Anpassungen sind im Abschnitt „APPC-Transaktion für den TSO Commands Service definieren“ auf Seite 24 beschrieben und die Anpassungen für SCLMDT im Abschnitt „ISPF-Konfigurationsdatei ISPF.conf anpassen“ auf Seite 46.

Wenn das SCLM Developer Toolkit für den TSO Commands Service verwendet wird oder das SCLMDT-Plug-in in der Clientkomponente von Developer für System z installiert ist, sind die folgenden Definitionen erforderlich.

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

Ausgangsverzeichnis des SCLM Developer Toolkit. Die Standardeinstellung ist /usr/lpp/SCLMDT. Passen Sie diese Einstellung an Ihre Installation des SCLM Developer Toolkit an. Diese Anweisung ist nur erforderlich, wenn das SCLM Developer Toolkit verwendet wird (TSO Commands Service oder Client-Plug-in).

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

Ladebibliothek des SCLM Developer Toolkit. Die Standardeinstellung ist BWB.SBWBLOAD. Passen Sie diese Einstellung an Ihre Installation des SCLM Developer Toolkit an. Diese Anweisung ist nur erforderlich, wenn das SCLM Developer Toolkit verwendet wird (TSO Commands Service oder Client-Plug-in).

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

Basiskonfigurationsverzeichnis des SCLM Developer Toolkit. Die Standardeinstellung ist /etc/SCLMDT. Setzen Sie diesen Wert auf Ihr angepasstes Verzeichnis für das SCLM Developer Toolkit. Diese Anweisung ist nur erforderlich, wenn das SCLM Developer Toolkit verwendet wird (TSO Commands Service oder Client-Plug-in).

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

Basisarbeitsverzeichnis des SCLM Developer Toolkit. Die Standardeinstellung ist /var/SCLMDT. Setzen Sie diesen Wert auf Ihr angepasstes Verzeichnis für das SCLM Developer Toolkit. Diese Anweisung ist nur erforderlich, wenn das SCLM Developer Toolkit verwendet wird (TSO Commands Service oder Client-Plug-in).

#### **STEPLIB**

STEPLIB für den RSE-Server. Die Standardeinstellung ist NONE. Ändern Sie diese Zeile nicht. Sie wird als Standardwert verwendet. Developer für System z verwendet standardmäßig die LINKLIST, um auf die Ladebibliothek des SCLM Developer Toolkit zuzugreifen. STEPLIB wird verwendet, wenn die folgende STEPLIB-Anweisung nicht auf Kommentar gesetzt ist:

STEPLIB=\$\_CMDSERV\_BASE\_LOAD

**Anmerkung:** Die Verwendung von STEPLIB unter z/OS UNIX wirkt sich negativ auf die Leistung aus. Lesen Sie hierzu die Informationen im Abschnitt „Verwendung von STEPLIB vermeiden“ auf Seite 67.

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

Zusätzliche, für den TSO Commands Service spezifische Java-Optionen. Die Standardeinstellung ist "". Weitere Informationen zu dieser Definition enthält der Abschnitt „Zusätzliche Java-Startparameter mit \_RSE\_\*OPTS definieren (optional)“ auf Seite 41. Diese Anweisung ist nur erforderlich, wenn das SCLM Developer Toolkit verwendet wird (TSO Commands Service oder Client-Plug-in).

Folgende Definitionen sind optional. Wenn Sie diese Definitionen übergehen, werden Standardwerte verwendet.

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

Gibt den Bereich der Ports an, die der RSE-Server für die Kommunikation mit einem Client öffnen kann. Standardmäßig kann jeder Port verwendet werden. Weitere Informationen zu dieser Definition enthält der Abschnitt „Für RSE verfügbaren PORTRANGE definieren (optional)“ auf Seite 40.

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

Vom Sperrenmanager zu verwendende Benutzer-ID. Standardmäßig wird die ID des angemeldeten Benutzers verwendet.

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

Vom Sperrenmanager zu verwendender Jobname. Die Standardeinstellung ist FEKFLK00.

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

Name des APPC-Transaktionsprogramms. Die Standardeinstellung ist FEKFRSRV. Entfernen Sie das Kommentarsymbol vor dieser Definition, und ändern Sie sie, wenn Sie beim Definieren der APPC-Transaktion nicht den Standardnamen verwendet haben. Weitere Informationen zur APPC-Transaktion enthält der Abschnitt „APPC-Transaktion für den TSO Commands Service definieren“ auf Seite 24.

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

Zwingt RSE, diese APPC-Basis-LU zu verwenden. Weitere Informationen zu dieser Definition enthält Anhang F, „APPC konfigurieren“, auf Seite 123.

Die folgenden Definitionen sind erforderlich und sollten nur auf Anweisung des IBM Support Center geändert werden.

#### **RSE\_LIB**

RSE-Bibliothekspfad. Die Standardeinstellung ist \$RSE\_HOME/rse/lib. Modifizieren Sie diese Einstellung nicht.

#### **ICU\_LIB**

ICU-Bibliothekspfad (International Components for Unicode). Die Standardeinstellung ist \$RSE\_HOME/icuc/lib. Modifizieren Sie diese Einstellung nicht.

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

Von den gestarteten Prozessen verwendete LE-Laufzeitoptionen (Language Environment). Die Standardeinstellung ist "ALL31(ON) HEAP(32M,32K,ANYWHERE,KEEP,,) TRAP(ON)". Modifizieren Sie diese Einstellung nicht.

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

Von der Java Virtual Machine (JVM) verwendete z/OS-UNIX-Position für den LE-Speicherauszug (Language Environment). Die Standardeinstellung ist \$HOME/.eclipse/RSE/\$RSE\_USER\_ID. Modifizieren Sie diese Einstellung nicht.

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

Ausführung von Vordergrundprozessen in demselben Adressbereich wie die Shell. Die Standardeinstellung ist YES. Modifizieren Sie diese Einstellung nicht.

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

Direkte Ausführung von Shell-Skripten von der Funktion spawn() aus. Die Standardeinstellung ist YES. Modifizieren Sie diese Einstellung nicht.

**PATH** Befehlspfad. Die Standardeinstellung ist \$JAVA\_HOME/bin:\$RSE\_LIB:\$CMDSEV\_BASE\_HOME/lib:\$PATH. Modifizieren Sie diese Einstellung nicht.

#### **LIBPATH**

Bibliothekspfad. Die Standardeinstellung ist \$JAVA\_HOME/bin:\$JAVA\_HOME/bin/classic:\$ICU\_LIB:\$RSE\_LIB:.. Modifizieren Sie diese Einstellung nicht.

#### **CLASSPATH**

Klassenpfad. Die Standardeinstellung ist zu lang, um sie hier wiederzugeben. Modifizieren Sie diese Einstellung nicht.



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

Zusätzliche, für den TSO Commands Service spezifische Java-Optionen. Die Standardeinstellung ist "&SESSION=SPAWN\$\_RSE\_CMDSERV\_OPTS". Modifizieren Sie diese Einstellung nicht.

#### **\_RSE\_JAVAOPTS**

Zusätzliche RSE-spezifische Java-Optionen. Die Standardeinstellung ist zu lang, um sie hier wiederzugeben. Modifizieren Sie diese Einstellung nicht.

#### **\_RSE\_SERVER\_CLASS**

Java-Klassen für den RSE-Server. Die Standardeinstellung ist `com.ibm.e-tools.systems.dstore.core.server.Server`. Modifizieren Sie diese Einstellung nicht.

#### **\_RSE\_SERVER\_TIMEOUT**

Zeitlimit für den RSE-Server (der auf den Client wartet) in Millisekunden. Die Standardeinstellung ist 120000 (2 Minuten). Modifizieren Sie diese Einstellung nicht.

**Anmerkung:** Wenn Sie die folgende STEPLIB-Anweisung am ENDE der Datei `rsed.envvars` hinzufügen, müssen keine C/C++- und LE-Bibliotheken (Language Environment) in der LINKLIST enthalten sein. Beachten Sie jedoch, dass sich die Verwendung von STEPLIB unter z/OS UNIX negativ auf die Leistung auswirkt. Lesen Sie hierzu die Informationen im Abschnitt „Verwendung von STEPLIB vermeiden“ auf Seite 67.

- Wenn die letzte STEPLIB-Anweisung, die zuvor in `rsed.envvars` definiert wurde, `STEPLIB=NONE` wie folgt lautet:  
`STEPLIB=CEE.SCEERUN:CEE.SCEERUN2:CBC.SCLBDLL`
- Wenn die letzte STEPLIB-Anweisung, die zuvor in `rsed.envvars` definiert wurde, nicht wie folgt lautet: `STEPLIB=NONE`  
`STEPLIB=$STEPLIB:CEE.SCEERUN:CEE.SCEERUN2:CBC.SCLBDLL`

**Anmerkung:** Für die Angabe von Verzeichnissen in `rsed.envvars` können Sie symbolische Links verwenden.

## **Für RSE verfügbaren PORTRANGE definieren (optional)**

Dieser Schritt gehört zur Anpassung der Datei `rsed.envvars`, die die Ports angibt, über die der RSE-Server mit dem Client kommunizieren kann. Dieser Port-Bereich steht nicht in Verbindung mit den Ports des RSE-Dämons oder den REXEC/SSH-Ports.

Nachfolgend sehen Sie eine kurze Beschreibung des RSE-Verbindungsprozesses, die Ihnen helfen soll, die Port-Verwendung zu verstehen.

1. Der Client stellt eine Verbindung zum Host-Port 4035 her (INETD-RSE-Dämonservice) oder zum Host-Port 512 (INETD-REXEC-Service) bzw. zum Host-Port 22 (INETD-SSH-Service).
2. Der ausgewählte INETD-Service erstellt einen RSE-Prozess.
3. Der RSE-Prozess öffnet einen Host-Port, zu dem der Client eine Verbindung herstellen kann. Der Benutzer kann die Auswahl dieses Ports auf dem Client auf der Merkmalregisterseite für das Subsystem (nicht zu empfehlen) oder mit der Definition `_RSE_PORTRANGE` in `rsed.envvars` konfigurieren.
4. Der INETD-Service gibt die Port-Nummer an den Client zurück.
5. Der Client stellt eine Verbindung zum Host-Port her.



Wenn Sie den Port-Bereich für die Kommunikation des Clients mit z/OS angeben möchten, entfernen Sie das Kommentarzeichen aus der folgenden Zeile in `rsd.envvars`, und ändern Sie die Zeile:

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

**Anmerkung:** Überprüfen Sie vor Auswahl eines Port-Bereichs, ob der Bereich auf Ihrem System verfügbar ist. Verwenden Sie dazu die Befehle **NETSTAT** und **NETSTAT PORTL**. Weitere Informationen hierzu enthält der Abschnitt „Reservierte TCP/IP-Ports“ auf Seite 85.

PORTRANGE hat folgendes Format: `_RSE_PORTRANGE=min-max`. (Die Angabe `max` gilt nicht einschließlich. Die Einstellung `_RSE_PORTRANGE=8108-8118` bedeutet beispielsweise, dass die Port-Nummern von 8108 bis einschließlich 8117 verwendet werden können.) Die vom RSE-Server verwendete Port-Nummer wird wie folgt ermittelt:

1. Wenn in den Subsystemmerkmalen auf dem Client eine Port-Nummer ungleich null angegeben ist, wird diese Port-Nummer verwendet. Ist der Port nicht verfügbar, scheitert die Verbindung. Diese Konfiguration wird nicht empfohlen.
2. Wenn die Port-Nummer in den Subsystemmerkmalen null ist und in `rsd.envvars` `_RSE_PORTRANGE` enthalten ist, wird der von `_RSE_PORTRANGE` angegebene Port-Bereich verwendet. Falls kein Port aus dem Bereich verfügbar ist, scheitert die Verbindung.
3. Wenn die Port-Nummer in den Subsystemmerkmalen null ist und `_RSE_PORTRANGE` nicht in `rsd.envvars` enthalten ist, wird ein beliebiger verfügbarer Port verwendet.

**Anmerkung:** Wenn ein Webserver einen Port öffnet und auf den Empfang von Daten wartet, kann dieser Port nicht von einem anderen Server verwendet werden. Ist die Verbindung jedoch einmal hergestellt, können mehrere Server denselben Port verwenden. Die Port-Nummern im Port-Bereich bedeuten also keine Einschränkung hinsichtlich der Anzahl gleichzeitig verbundener Benutzer.

## Zusätzliche Java-Startparameter mit `_RSE_*OPTS` definieren (optional)

Mit den verschiedenen `_RSE_*OPTS`-Anweisungen bietet die Datei `rsd.envvars` die Möglichkeit, zusätzliche Parameter für Java anzugeben, wenn der RSE-Server gestartet wird.

Die in `rsd.envvars` enthaltenen Beispieloptionen können durch Entfernen des Kommentarzeichens aktiviert werden.

### `_RSE_JVAOPTS`

`_RSE_JVAOPTS` definiert RSE-spezifische Java-Optionen und Standard-Java-Optionen.

```
_RSE_JVAOPTS=""
```

Variableninitialisierung. Modifizieren Sie diese Einstellung nicht.

```
#_RSE_JVAOPTS="$_RSE_JVAOPTS -Xquickstart"
```

Verbessert die Startzeit durch eine Verzögerung der JIT-Kompilierung und -Optimierungen. Die Effizienz bei der Kompilierung ausführbarer Dateien geht dadurch jedoch leicht zurück, was sich bei Tasks mit langer Ausführungszeit bemerkbar macht. Entfernen Sie das Kommentarzeichen, um diese Option zu aktivieren.

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Xms128m -Xmx128m"**

Festlegen der anfänglichen Heap-Größe (Xms) und der maximalen Heap-Größe (Xmx). Die Systemstandardwerte sind jeweils 1M und 64M. Entfernen Sie das Kommentarzeichen, und ändern Sie diese Option, damit die angegebenen Heap-Größenangaben angewendet werden.

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

Auswahl der Host-Codepage. Entfernen Sie das Kommentarzeichen, und ändern Sie die Option, damit die angegebene Codepage verwendet wird.

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

Option für die Kennwortspeicherung. Entfernen Sie das Kommentarzeichen, wenn Benutzer nicht in der Lage sein sollen, ihr Hostkennwort auf dem Client zu speichern. Bereits gespeicherte Kennwörter werden dann entfernt. Diese Option funktioniert nur bei Clients ab der Version 7.1.

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS  
-DDSTORE\_TRACING\_ON=true"**

Starten des DSTORE-Trace. Verwenden Sie diese Option nur auf Anweisung des IBM Support Center.

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS  
-DDSTORE\_MEMLOGGING\_ON=true"**

Starten des DSTORE-Speicher-Trace. Verwenden Sie diese Option nur auf Anweisung des IBM Support Center.

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

Verwendung von APPC für den TSO Commands Service. Weitere Informationen hierzu enthält der Abschnitt „APPC-Transaktion für den TSO Commands Service definieren“ auf Seite 24.

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

Die Anweisung `_RSE_CLASS_OPTS` definiert die Optionen für Java ab Version 5.0, die für die gemeinsame Nutzung von Klassen durch mehrere RSE-Server erforderlich sind. Weitere Informationen hierzu enthält der Abschnitt „Gemeinsame Klassennutzung durch mehrere JVMs“ auf Seite 69.

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

Variableninitialisierung. Modifizieren Sie diese Einstellung nicht.

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

Aktivierung der gemeinsamen Klassennutzung; nur für Java ab Version 5.0. Entfernen Sie das Kommentarzeichen, wenn Klassen von mehreren RSE-Servern gemeinsam genutzt werden sollen.

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

Festlegen der Größe des Cache für gemeinsam genutzte Klassen; nur für Java ab Version 5.0. Der Systemstandardwert liegt bei 16M.

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

Die `_RSE_CMDSERV_OPTS`-Anweisungen sind RSE-spezifische Java-Optionen, die nur wirksam sind, wenn das SCLM Developer Toolkit als TSO Commands Server verwendet wird.

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

Variableninitialisierung. Modifizieren Sie diese Einstellung nicht.

```
#_RSE_CMDSERV_OPTS="$_RSE_CMDSERV_OPTS&ISPPROF=
&SYSUID..ISPPROF"
```

Verwendung eines vorhandenen ISPF-Profiles für den TSO Commands Server. Entfernen Sie das Kommentarzeichen, und ändern Sie den Dateigruppennamen, um das angegebene ISPF-Profil zu verwenden. &SYSUID. kann als Ersatz für die Benutzer-ID des Entwicklers verwendet werden.

---

## INETD-Dämon und -REXEC/SSH für RSE konfigurieren

Wenn ein Client eine Verbindung anfordert, greift Developer für System z auf den INETD-Service zurück, um den RSE-Server (Remote Systems Explorer) zu starten. INETD ist ein z/OS UNIX-Standarddämon, der andere Dämonen verwaltet, die die eigentliche Arbeit ausführen (in diesem Falle das Starten des RSE-Servers). INETD wird nicht im Rahmen der Anpassung von Developer für System z konfiguriert. Anhang D, „INETD konfigurieren“, auf Seite 103 enthält dennoch wertvolle Informationen hierzu.

Developer für System z unterstützt mehrere Methoden für den Start des RSE-Servers.

Sie müssen, je nach vorgesehener Arbeitsweise der Benutzer, mindestens eine dieser Methoden anpassen.

- Der RSE-Dämon wird durch das Herstellen einer Verbindung zu INETD am Port 4035 (Standard) gestartet. Dies ist die empfohlene Methode, da sie dem Benutzer mehr Steuerungsmöglichkeiten und mehr Sicherheit als REXEC gibt.
- Der REXEC-Befehlsserver (Remote Execution) führt am Standard-Port 512 ein Shell-Script aus, das RSE aufruft. Diese Verbindungsmethode kann zur Prüfung der Erfolgchancen verwendet werden, ist jedoch nicht für die Langzeitanwendung bestimmt. Diese Methode hat den Vorteil, dass der Installations- und Konfigurationsaufwand geringer ist, sofern REXEC bereits konfiguriert wurde. (In der Regel geschieht dies im Rahmen der TCP/IP-Konfiguration.)
- Der SSH-Befehlsserver (Secure Shell) führt am Standard-Port 22 ein Shell-Script aus, das RSE aufruft. Diese Methode ist mit REXEC vergleichbar, SSH kommuniziert jedoch sicher (verschlüsselt) mit dem Client.

**Anmerkung:** Ferne (hostbasierte) Aktionen für z/OS-UNIX-Unterprojekte erfordern, dass auf dem Host REXEC oder SSH aktiv ist.

Als berechtigter Benutzer können Sie den Befehl **ps -e** absetzen, um zu überprüfen, ob INETD aktiv ist. Die Ausgabe muss einen Verweis auf INETD enthalten. Beispiel (# ist die z/OS-UNIX-Eingabeaufforderung):

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

**Anmerkung:** Wenn z/OS-UNIX-Server (wie RSE-Dämon, REXEC und SSH) IPv6-Verbindungen unterstützten, muss in der Datei /etc/inetd.conf als Protokoll für den Servicenamen tcp6 angegeben sein. IPv4-Clients werden ebenfalls unterstützt, wenn tcp6 definiert ist. /etc/services unterstützt nur das Schlüsselwort tcp ohne ein Zahlensuffix.

## INETD-RSE-Dämon konfigurieren

1. Fügen Sie zu /etc/services die folgende Zeile hinzu:

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

**rse** Gibt den Servicenamen des Dämons an (standardmäßig rse in Kleinbuchstaben). Der Name muss mit dem in /etc/inetd.conf verwendeten Namen übereinstimmen.

#### **4035/tcp**

Gibt den verwendeten Port und das verwendete Protokoll an. Der Standard-Port ist 4035; das Protokoll muss tcp sein.

Der verwendete Port muss mit dem auf dem Client definierten Port übereinstimmen. Auf dem Client wird der Port während der Erstellung einer neuen z/OS-Verbindung festgelegt.

**Anmerkung:** Überprüfen Sie vor Auswahl eines Ports, ob der Port auf Ihrem System verfügbar ist. Verwenden Sie dazu die Befehle **NETSTAT** und **NETSTAT PORTL**. Weitere Informationen hierzu enthält der Abschnitt „Reservierte TCP/IP-Ports“ auf Seite 85.

#### **#Developer für System z RSE**

Kommentar, der mit einem Nummernzeichen (#) beginnen muss

2. Fügen Sie die beiden folgenden Zeilen zu /etc/inetd.conf hinzu. Informationen zu Fortsetzungsregeln finden Sie in Anhang D, „INETD konfigurieren“, auf Seite 103.

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

**rse** Servicename des Dämons. Die Standardeinstellung ist rse (in Kleinbuchstaben). Der Name muss mit dem in /etc/services verwendeten Namen übereinstimmen.

#### **stream tcp nowait**

INETD-spezifische Konfigurationsanweisungen (Socket-Typ, Protokoll, Option wait). Modifizieren Sie diese Einstellung nicht.

**Anmerkung:** Verwenden Sie an Stelle von tcp das Schlüsselwort tcp6, um IPV6-Verbindungen zu unterstützen.

#### **OMVSKERN**

Benutzer-ID für den RSE-Dämonprozess. Die Standardeinstellung ist OMVSKERN. Diese Benutzer-ID muss eine Benutzer-ID mit einem gültigen OMVS-Sicherheitssegment, der Berechtigung BPX.DAEMON sowie den Zugriffsberechtigungen READ und EXECUTE für das Installations- und Konfigurationsverzeichnis von Developer für System z sein. Weitere Einzelheiten zu den Anforderungen an Benutzer-IDs, die für System-services verwendet werden, finden Sie in Anhang D, „INETD konfigurieren“, auf Seite 103.

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

Serverprogramm (absolute Position von fekfrsed). Die Standardeinstellung ist /usr/lpp/wd4z/rse/lib/fekfrsed.

Alle Angaben nach diesem INETD-Argument sind Serverargumente. Das erste Serverargument ist der Servername.

**rsed** Servername. Modifizieren Sie diese Einstellung nicht.

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

Arbeitsverzeichnis (Position der RSE-Serverkonfigurationsdateien). Die Standardeinstellung ist /usr/lpp/wd4z/rse/lib.

**Anmerkung:** Sie sollten die angepassten RSE-Konfigurationsdateien in ein neues Verzeichnis kopieren (z. B. in /etc/wd4z/), damit sie im Falle einer Wartung nicht überschrieben werden. Das hier definierte Arbeitsverzeichnis muss diese Änderung widerspiegeln. Beispiel:

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

Folgende Definitionen sind optional. Wenn Sie diese Definitionen übergehen, werden Standardwerte verwendet.

**-t 60** Zeitlimitoption, mit der angegeben werden kann, wie viele Sekunden der RSE-Dämon auf eine Antwort des RSE-Servers warten soll. Die Standardeinstellung ist 60 (Sekunden). Das Zeitlimit für den RSE-Server, der auf den Client wartet, wird in `rsed.envvars` festgelegt und liegt standardmäßig bei zwei Minuten.

3. Zum Aktivieren der Änderungen an den /etc-Dateien muss INETD von einem berechtigten Benutzer neu gestartet werden. Diesbezügliche Informationen finden Sie in Anhang D, „INETD konfigurieren“, auf Seite 103. Schauen Sie sich die folgenden Beispielbefehle an (# ist die z/OS-UNIX-Eingabeaufforderung):

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

**Anmerkung:** Wenn das Profil BPX.DAEMON in der Klasse FACILITY Ihres Sicherheitsprodukts definiert ist und der Benutzer, der INETD (erneut) startet, keinen Zugriff auf diese Ressource hat, müssen Sie für jeden Client, der eine Verbindung zu RSE herstellt, mit der folgenden Sicherheitswarnung rechnen. Die Angabe IBMUSER steht hier für die Benutzer-ID, die für den Start von INETD verwendet wurde.

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

## INETD-REXEC (oder -SSH) konfigurieren

Es gibt keine spezifische Konfiguration für Developer für System z für die Verwendung des INETD-REXEC-Befehlsservers (oder -SSH-Befehlsservers). Der Client muss jedoch zwei Werte kennen, um über REXEC/SSH eine RSE-Verbindung herstellen zu können:

- Das Verzeichnis, in dem sich das Start-Skript `server.zseries` befindet.

Standardmäßig ist dies das Installationsverzeichnis (/usr/lpp/wd4z/rse/lib/). Die Datei `server.zseries` gehört allerdings zu den Dateien, die ebenfalls kopiert werden müssen, wenn Sie `rsed.envvars` in ein anderes Verzeichnis, z. B. in /etc/wd4z/, kopieren.

- Der verwendete Port.

Ein von REXEC verwendeter allgemeiner Port ist 512. Für eine schnelle Überprüfung können Sie den Befehl **NETSTAT** verwenden. Sehen Sie sich dazu das folgende Beispiel an (\$ ist die z/OS-UNIX-Eingabeaufforderung):

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

Zur Verifizierung können Sie in `/etc/inetd.conf` und `/etc/services` die verwendete Port-Nummer überprüfen.

1. Suchen Sie in der Datei `/etc/inetd.conf` den Servicenamen (erstes Wort; in diesem Beispiel `exec`) des `rexecd`-Servers (siebentes Wort).  
`exec stream tcp nowait OMVSKERN /usr/sbin/orexecd rexecd -LV`
2. Suchen Sie in der Datei `/etc/services` den Port (zweites Wort; in diesem Beispiel 512), der diesem Servicenamen (erstes Wort) zugeordnet ist.  
`exec 512/tcp #REXEC-Befehlsserver`

Dasselbe Prinzip gilt für SSH. Der allgemeine Port ist 22, und der Servername ist `sshd`.

**Anmerkung:** Ferne (hostbasierte) Aktionen für z/OS-UNIX-Unterprojekte erfordern, dass auf dem Host REXEC oder SSH aktiv ist. Wenn REXEC/SSH nicht für die Verwendung des Standard-Ports konfiguriert ist, muss die Clientkomponente von Developer für System z den korrekten Port für z/OS-UNIX-Unterprojekte definieren. Hierfür können Sie die Vorgabenseite **Fenster > Benutzervorgaben... > z/OS-Lösungen > USS-Unterprojekte > Optionen für ferne Aktionen** auswählen.

---

## ISPF-Konfigurationsdatei `ISPF.conf` anpassen

Dieser Schritt ist nur erforderlich, wenn für den TSO Commands Service das SCLM Developer Toolkit verwendet wird (Standardmethode). Er ist nicht erforderlich, wenn Sie für den TSO Commands Service APPC verwenden.

Das SCLM Developer Toolkit benötigt zum Erstellen einer für die Ausführung von ISPF-Services gültigen Umgebung Definitionen in `ISPF.conf`. Zu dieser ISPF-Umgebung muss der TSO Commands Service hinzugefügt werden.

`ISPF.conf` wird während der Anpassung des SCLM Developer Toolkit erstellt. Diese Anpassung ist in der Veröffentlichung *SCLM Developer Toolkit Installation und Anpassung (IBM Form SC12-4101)* beschrieben. Die Standardposition ist `/etc/SCLMDT/CONFIG`; an Ihrem Standort kann jedoch ein anderer Pfad verwendet worden sein.

Fügen Sie die folgenden Zeilen zu `ISPF.conf` hinzu. HLQ steht hier für den bei der Installation von Developer für System z verwendeten High Level Qualifier (standardmäßig FEK).

```
*****
* Developer für System z – TSO Commands Server
*****
sysexec=HLQ.SFEKPROC
```

Das Ergebnis sollte wie das Beispiel in Abb. 5 auf Seite 47 aussehen.



```

sysproc=ISP.SISPLIB
isplib=ISP.SISPMENU
isptlib=ISP.SISPTENU
ispplib=ISP.SISPPENU
ispslib=ISP.SISPSLIB
ispllib=BWB.SWBLOAD
*****
* Developer für System z – TSO Commands Server
*****
sysexec=FEK.SFEKPROC

```

Abbildung 5. ISPF-Konfigurationsdatei ISPF.conf

**Anmerkung:** Falls die Anweisung sysexec bereits definiert ist, fügen Sie die Dateigruppe HLQ.SFEKPROC am Ende der Anweisung hinzu. Trennen Sie die einzelnen Dateigruppenamen jeweils durch ein Komma (,).

## RSE-Serverkonfiguration prüfen

Die Installation von Developer für System z stellt mehrere Installationsprüfprogramme (IVP, Installation Verification Programs) für den RSE-Server bereit. Die IVP-Scripts befinden sich im Installationsverzeichnis (standardmäßig /usr/lpp/wd4z/rse/lib/).

- fekfivpa: siehe Abschnitt „TSO-Commands-Service-Verbindung (bei Verwendung von APPC)“ auf Seite 52
- fekfivpc: siehe Abschnitt „TSO-Commands-Service-Verbindung (bei Verwendung von SCLMDT)“ auf Seite 51
- fekfivpd: siehe Abschnitt „RSE-Dämonverbindung“ auf Seite 50
- fekfivpj: siehe Abschnitt „JES-Job-Monitor-Verbindung“ auf Seite 51
- fekfivpr: siehe Abschnitt „REXEC-Verbindung“ auf Seite 49
- fekfivps: siehe Abschnitt „REXEC/SSH-Shell-Script“ auf Seite 49

Bei allen Beispielbefehlen in diesem Abschnitt wird vorausgesetzt, dass die RSE-Umgebungsvariablen gesetzt sind. Wenn das der Fall ist, sind die IVP-Scripts über die Anweisung PATH verfügbar, und die Position von rsed.envvars ist bekannt. Verwenden Sie die Befehle **pwd** und **cd**, um Ihr aktuelles Verzeichnis zu prüfen und das Verzeichnis mit der angepassten Datei rsed.envvars aufzurufen. Danach können Sie mit dem Shell-Script setup.env.zseries die RSE-Umgebungsvariablen setzen. Sehen Sie sich hierzu das folgende Beispiel an (\$ ist die z/OS-UNIX-Eingabeaufforderung):

```

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

```

Das Shell-Script . ./setup.env.zseries befindet sich in demselben Verzeichnis wie rsed.envvars und exportiert die Umgebungsvariablen, damit sie von anderen Prozessen verwendet werden können. Der erste Punkt (".") in . ./setup.env.zseries ist ein z/OS-UNIX-Befehl zur Ausführung der Shell in der aktuellen Umgebung, damit die in der Shell gesetzten Umgebungsvariablen auch nach dem Beenden der Shell in Kraft bleiben. Der zweite Punkt bezieht sich auf das aktuelle Verzeichnis.

**Anmerkung:** Wenn . ./setup.env.zseries nicht vor den fekfiv\*-Scripts ausgeführt wird, muss der Pfad zu diesen Scripts angegeben werden, wenn sie aufgerufen werden. Sehen Sie sich dazu das folgende Beispiel an:

```

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

```

Die meisten fekfivp\*-Scripts fordern außerdem die Position der angepassten Datei rsed.envvars an, wenn . ./setup.env.zseries nicht zuerst ausgeführt wird.

**Anmerkung:** Einige IVP-Tests verwenden die TCP/IP-REXX-Socket-API, die erfordert, dass die TCP/IP-Ladebibliothek (standardmäßig TCPIP.SEZALOAD) in der LINKLIST oder STEPLIB enthalten ist. Für die Ausführung solcher IVP-Tests können die folgenden Befehle erforderlich sein (\$ ist die z/OS-UNIX-Eingabeaufforderung):

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

oder

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

Informationen zur Diagnostizierung von RSE-Verbindungsproblemen können Sie Anhang B, „Konfigurationsprobleme lösen“, auf Seite 79 oder den technischen Hinweisen auf der Supportseite zu Developer für System z (<http://www-306.ibm.com/software/awdtools/devzseries/support/>) entnehmen.

## Port-Verfügbarkeit

Die Port-Verfügbarkeit für JES Job Monitor, REXEC, SSH und den RSE-Dämon können Sie durch Absetzen des Befehls **netstat** prüfen. Das Ergebnis sollte die von diesen Services verwendeten Ports zeigen. Sehen Sie sich dazu die folgenden Beispiele an (\$ ist die z/OS-UNIX-Eingabeaufforderung):

### IPV4

```
$ netstat
MVS TCP/IP NETSTAT CS V1R7    TCPIP Name: TCPIP      13:57:36
User Id  Conn      Local Socket      Foreign Socket      State
-----  ----  -----
INETD4   00000014  0.0.0.0..22      0.0.0.0..0         Listen
INETD4   00000030  0.0.0.0..512     0.0.0.0..0         Listen
INETD4   0000004B  0.0.0.0..4035    0.0.0.0..0         Listen
JMON     00000037  0.0.0.0..6715    0.0.0.0..0         Listen
```

### IPV6

```
$ netstat
MVS TCP/IP NETSTAT CS V1R7    TCPIP Name: TCPIP      14:03:35
User Id  Conn      State
-----  ----  -----
INETD4   00000018 Listen
Local Socket: 0.0.0.0..22
Foreign Socket: 0.0.0.0..0
INETD4   00000046 Listen
Local Socket: 0.0.0.0..512
Foreign Socket: 0.0.0.0..0
INETD4   0000004B Listen
Local Socket: 0.0.0.0..4035
Foreign Socket: 0.0.0.0..0
JMON     00000037 Listen
Local Socket: 0.0.0.0..6715
Foreign Socket: 0.0.0.0..0
```



## REXEC-Verbindung

Führen Sie den folgenden Befehl aus, um die REXEC-Verbindung zu überprüfen. Ersetzen Sie 512 durch den von REXEC verwendeten Port und USERID durch eine gültige Benutzer-ID.

```
fekfivpr 512 USERID
```

Nachdem der Befehl Sie zur Eingabe eines Kennworts aufgefordert hat, sollte er den REXEC-Trace, eine Warnung zum Zeitlimit, die Java-Version und die RSE-Servernachricht zurückgeben. Sehen Sie sich hierzu das folgende Beispiel an (\$ ist die z/OS-UNIX-Eingabeaufforderung):

```
$ fekfivpr 512 USERID
Enter password:
$ EYZRC01I Calling function rexec_af with the following:
EYZRC02I Host: CDFMVS08, user USERID, cmd cd /etc/wd4z;export RSE_USER_ID=USERID;./server.zseries -ivp, port 512
EYZRC19I Data socket = 4, Control socket = 6.
```

expect to see time out messages after a successful IVP test

```
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
```

```
Server Started Successfully
1272
Server running on: CDFMVS08
```

**Anmerkung:** Falls Sie keine Java- und RSE-Serverausgabe empfangen, ist wahrscheinlich die INETD-Region zu klein. (Beim Start von einer TSO/OMVS-Shell-Sitzung aus muss die Regionsgröße bei mindestens 2096128 liegen und beim Start mit BPXBATCH bei 0.)

**Anmerkung:** Das von REXEC verwendete Shell-Script können Sie gesondert testen. Eine diesbezügliche Beschreibung enthält der nächste IVP-Test im Abschnitt „REXEC/SSH-Shell-Script“.

**Anmerkung:** Der Server wird gestartet, ohne dass ein Client versucht, eine Verbindung herzustellen, so dass das Zeitlimit (von fünf Sekunden) überschritten wird. Diese Überschreitung führt zu einem Java-Stack-Trace (mit 25 und mehr Zeilen), der so ähnlich wie das folgende Beispiel aussieht:

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

```

## REXEC/SSH-Shell-Script

Wenn Sie den vorherigen IVP-Test aus dem Abschnitt „REXEC-Verbindung“ erfolgreich beendet haben, können Sie diesen Test überspringen.

Führen Sie den folgenden Befehl aus, um das von der REXEC- und SSH-Verbindung verwendete Shell-Script zu überprüfen:

```
fekfivps
```

Der Befehl sollte eine Warnung zum Zeitlimit, die Java-Version und die RSE-Servernachricht zurückgeben. Sehen Sie sich hierzu das folgende Beispiel an (\$ ist die z/OS-UNIX-Eingabeaufforderung):

```
$ fekfivps
```

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

expect to see time out messages after a successful IVP test

```
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 j9vmz3123-20070201 (JIT
enabled)
J9VM - 20070131_11312_bHdSMr
JIT - 20070109_1805ifx1_r8
GC - 200701_09)
JCL - 20070126
```

```
Server Started Successfully
```

```
1751
```

```
Server running on: CDFMVS08$
```

**Anmerkung:** Falls keine Ausgabe angezeigt wird, ist Ihre TSO-Region wahrscheinlich zu klein. (Die Regionsgröße muss bei mindestens 2096128 liegen.)

**Anmerkung:** Der Server wird gestartet, ohne dass ein Client versucht, eine Verbindung herzustellen, so dass das Zeitlimit (von fünf Sekunden) überschritten wird. Diese Überschreitung führt zu einem Java-Stack-Trace (mit 25 und mehr Zeilen), der so ähnlich wie das folgende Beispiel aussieht:

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

```

## RSE-Dämonverbindung

Führen Sie den folgenden Befehl aus, um die RSE-Dämonverbindung zu überprüfen. Ersetzen Sie 4035 durch den vom RSE-Dämon verwendeten Port und USERID durch eine gültige Benutzer-ID.

```
fekfivpd 4035 USERID
```

Nach einer Aufforderung zur Kennworteingabe sollte der Befehl eine Ausgabe wie im folgenden Beispiel zurückgeben (\$ ist die z/OS-UNIX-Eingabeaufforderung):

```
$ fekfivpd 4035 USERID
```

```
Password:
```

```
SSL is disabled
```

```
connected
```

```
8108
```

```
570655399
```

```
Success
```

**Anmerkung:** Wenn Sie eine SSL-fähige Verbindung testen, überprüfen Sie beim Empfang der folgenden Fehlnachricht, ob Sie den richtigen Port angegeben haben: gsk\_secure\_socket\_init() failed: Socket closed by remote partner

## JES-Job-Monitor-Verbindung

Führen Sie den folgenden Befehl aus, um die JES-Job-Monitor-Verbindung zu überprüfen. Ersetzen Sie 6715 durch die Port-Nummer von JES Job Monitor.

```
fekfivpj 6715
```

Der Befehl sollte die Bestätigungsnachricht von JES Job Monitor zurückgeben. Vergleichen Sie hierzu das folgende Beispiel (\$ ist die z/OS-UNIX-Eingabeaufforderung):

```
$ fekfivpj 6715
Waiting for JES Job Monitor response...
ACKNOWLEDGE01v03

Success
```

## TSO-Commands-Service-Verbindung (bei Verwendung von SCLMDT)

Dieser IVP-Test ist nur notwendig, wenn Sie das SCLM Developer Toolkit für den TSO Commands Service oder das Client-Plug-in verwenden.

Überprüfen Sie bei Verwendung von SCLM Developer die Verbindung zum TSO Commands Server, indem Sie den folgenden Befehl ausführen.

```
fekfivpc
```

Der Befehl sollte die Ergebnisse der auf das SCLM Developer Toolkit bezogenen Überprüfungen zurückgeben (Variablen, HFS-Module, REXX-Laufzeit, Start und Stopp der TSO/ISPF-Sitzung). Vergleichen Sie hierzu das folgende Beispiel (\$ ist die z/OS-UNIX-Eingabeaufforderung):

```
$ fekfivpc
-----
Host install verification for RSE
Review IVP log messages from HOST below :
-----

RSE connection and base TSO/ISPF session initialization check only

*** CHECK : ENVIRONMENT VARIABLES - key variables displayed below :

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

STEPLIB              = BWB.SWBLOAD

_CMDSERV_BASE_HOME   = /usr/lpp/SCLMDT
_CMDSERV_CONF_HOME   = /etc/SCLMDT
_CMDSERV_WORK_HOME    = /var/SCLMDT
-----

*** CHECK : HFS MODULES
Checking install Directory : /usr/lpp/SCLMDT
Checking for BWB modules in /bin directory
RC=0
MSG: SUCCESSFUL

-----

*** CHECK : REXX RUNTIME ENVIRONMENT
```

```
RC=0
MSG: SUCCESSFUL
```

```
-----
*** CHECK : TSO/ISPF INITIALIZATION
( TSO/ISPF session will be initialized )
RC=0
MSG: SUCCESSFUL
```

```
-----
*** CHECK: Shutting down TSO/ISPF IVP session
RC=0
MSG: SUCCESSFUL
```

```
-----
Host installation verification completed successfully
-----
```

**Anmerkung:** Falls eine der SCLMDT-Überprüfungen fehlschlägt, werden detaillierte Informationen angezeigt.

Der Befehl `fekfivpc` kann mit mehreren optionalen, nicht positionsgebundenen Parametern verwendet werden:

**-file** Der Befehl `fekfivpc` kann umfangreiche Ausgaben (mit Hunderten von Zeilen) erzeugen. Der Parameter `-file` sendet diese Ausgabe an eine Datei `home/.eclipse/RSE/USERID/fekfivpc.log`. Hier steht `home` für den Ausgangspfad, der in Ihrem OMVS-Segment definiert ist (oder im OMVS-Standardsegment, falls Sie kein Segment haben), und `USERID` für Ihre Benutzer-ID (in Großbuchstaben).

**-plugin** Standardmäßig überprüft `fekfivpc` nur die für den TSO Commands Service benötigten Funktionen. Mit dem Parameter `-plugin` können Sie zusätzliche Tests für das SCLMDT-Client-Plug-in hinzufügen.

**-debug** Der Parameter `-debug` erstellt eine detaillierte Testausgabe. Verwenden Sie diese Option nur auf Anweisung des IBM Support Center.

## TSO-Commands-Service-Verbindung (bei Verwendung von APPC)

Führen Sie diese Prozedur nur aus, wenn Sie APPC für den TSO Commands Service konfiguriert haben.

Überprüfen Sie (bei Verwendung von APPC) die Verbindung zum TSO Commands Server, indem Sie den folgenden Befehl ausführen. Ersetzen Sie `USERID` durch eine gültige Benutzer-ID.

```
fekfivpa USERID
```

Nach einer Aufforderung zur Kennworteingabe sollte der Befehl den APPC-Dialog zurückgeben. Sehen Sie sich dazu das folgende Beispiel an (\$ ist die z/OS-UNIX-Eingabeaufforderung):

```
$ fekfivpa USERID
Enter password:
20070607 13:57:18.584060 /usr/lpp/wd4z/rse/lib/fekfscmd: version=Oct 2003
20070607 13:57:18.584326 Input parms: 1.2.3.4 * NOTRACE USERID *****
20070607 13:57:18.585132 TP_name set via envvar: FEKFRSRV
20070607 13:57:18.586800 APPC: Allocate succeeded
20070607 13:57:18.587022 Conversation id is 0DDBD3F800000000
```

```

20070607 13:57:18.587380 APPC: Set Send Type succeeded
20070607 13:57:26.736674 APPC: Confirm succeeded
20070607 13:57:26.737027 Req to send recd value is 0
20070607 13:57:26.737546 APPC: SEND_DATA return_code = 0
20070607 13:57:26.737726 request_to_send_received = 0
20070607 13:57:26.737893 Send Data succeeded
20070607 13:57:26.738169 APPC: Set Prepare to Receive type succeeded
20070607 13:57:26.738580 APPC: Prepare to Receive succeeded
20070607 13:57:26.808899 APPC: Receive data
20070607 13:57:26.809122 RCV return_code = 0
20070607 13:57:26.809270 RCV data_received= 2
20070607 13:57:26.809415 RCV received_length= 29
20070607 13:57:26.809556 RCV status_received= 4
20070607 13:57:26.809712 RCV req_to_send= 0
20070607 13:57:26.809868 Receive succeeded
:IP: 0 9.42.112.75 1674 50246
20070607 13:57:26.810533 APPC: CONFIRMED succeeded

```

Informationen zur Diagnostizierung von RSE-Verbindungsproblemen können Sie Anhang B, „Konfigurationsprobleme lösen“, auf Seite 79 oder den technischen Hinweisen auf der Supportseite zu Developer für System z (<http://www-306.ibm.com/software/awdtools/devzseries/support/>) entnehmen.

---

## RSE-SSL-Konfiguration in `ssl.properties` anpassen (optional)

Alle Verbindungsmethoden der Clientkomponente von Developer für System z verwenden die in der Datei `ssl.properties` gesetzten SSL-Variablen (Secure Sockets Layer). Standardmäßig befindet sich die Datei im Installationsverzeichnis `/usr/lpp/wd4z/rse/lib/`. Die Datei `ssl.properties` gehört allerdings zu den Dateien, die ebenfalls kopiert werden müssen, wenn Sie `rsed.envvars` in ein anderes Verzeichnis, z. B. in `/etc/wd4/`, kopieren. Die bereitgestellte Beispieldatei enthält die in Abb. 6 aufgelisteten Anweisungen. Kommentarzeilen beginnen mit einem Nummernzeichen (#).

```

# Setzen Sie dieses Merkmal zum Aktivieren von SSL auf true.
enable_ssl=false

#####
# Dämonmerkmale
# Die vom Dämon zu verwendende Schlüsseldatei und das
# zu verwendende Kennwort müssen angegeben werden.
# Der Schlüsselkennsatz muss angegeben werden, wenn
# kein Standardschlüssel verwendet wird.
#daemon_keydb_file=
#daemon_keydb_password=
#daemon_key_label=

#####
# Servermerkmale
# Die vom Server zu verwendende Keystore-Datei und das
# zu verwendende Kennwort müssen angegeben werden.
#server_keystore_file=
#server_keystore_password=

```

*Abbildung 6. SSL-Konfigurationsdatei `ssl.properties`*

Die Dämon- und Servermerkmale müssen nur gesetzt werden, wenn Sie SSL aktivieren. Weitere Informationen zur SSL-Konfiguration enthält Anhang E, „SSL konfigurieren“, auf Seite 111.

---

## RSE-Trace-Konfiguration in `rsecomm.properties` anpassen (optional)

Alle Verbindungsmethoden der Clientkomponente von Developer für System z verwenden die in der Datei `rsecomm.properties` gesetzten Variablen. Standardmäßig befindet sich die Datei im Installationsverzeichnis `/usr/lpp/wd4z/rse/lib/`. Die Datei `rsecomm.properties` gehört allerdings zu den Dateien, die ebenfalls kopiert werden müssen, wenn Sie `rsed.envvars` in ein anderes Verzeichnis, z. B. in `/etc/wd4z/`, kopieren. Die bereitgestellte Beispieldatei enthält die in Abb. 7 aufgelisteten Anweisungen. Kommentarzeilen beginnen mit einem Nummernzeichen (#).

```
# server.version - NICHT ÄNDERN!
server.version=5.0.0

# Protokollstufe
# 0 - Fehlnachrichten protokollieren
# 1 - Fehlnachrichten und Warnungen protokollieren
# 2 - Fehlnachrichten, Warnungen und Informationen protokollieren
# 3 - Fehlnachrichten, Warnungen, Informationen und Debug-Nachrichten protokollieren
debug_level=1

# Protokollposition
# Log_To_StdOut
# Log_To_File
log_location=Log_To_File
```

*Abbildung 7. Konfigurationsdatei für Protokollierung `rsecomm.properties`*

Wenn Sie `log_location=Log_To_File` (die Standardeinstellung) auswählen, werden die Protokollnachrichten in `home/.eclipse/RSE/USERID/rsecomm.log` geschrieben. Hier steht `home` für den im OMVS-Segment des Benutzers definierten Ausgangspfad (bzw. den im Standard-OMVS-Segment definierten Ausgangspfad, falls der Benutzer kein eigenes OMVS-Segment hat), und `USERID` ist die ID des angemeldeten Benutzers (in Großbuchstaben).

**Anmerkung:** Die Definition `debug_level` steuert auch die Protokollstufe der anderen Protokolldateien in diesem Verzeichnis.

**Achtung:** Änderungen an diesen Einstellungen können zu Leistungseinbußen führen und sollten nur auf Anweisung des IBM Support Center vorgenommen werden.

---

## Konfiguration von Hostprojekten in `projectcfg.properties` anpassen (optional)

z/OS-Projekte können in der Perspektive für z/OS-Projekte auf dem Client einzeln definiert werden. Sie können z/OS-Projekte aber auch zentral auf dem Host definieren und dann benutzerabhängig auf dem Client replizieren. Solche hostbasierten Projekte sind vom Aussehen und von der Funktionsweise her mit auf dem Client definierten Projekten identisch. Die Struktur, die Member und die Merkmale dieser Projekte können jedoch nicht vom Client geändert werden und sind nur bei bestehender Verbindung zum Host verfügbar.

Die Position der Projektdefinitionen ist in der Datei `projects.properties` festgelegt. Die Datei befindet sich standardmäßig im Installationsverzeichnis `/usr/lpp/`

wd4z/rse/lib/. Die Datei `projectcfg.properties` gehört allerdings zu den Dateien, die ebenfalls kopiert werden müssen, wenn Sie `rsed.envvars` in ein anderes Verzeichnis, z. B. in `/etc/wd4z/`, kopieren.

Die bereitgestellte Beispieldatei enthält die in Abb. 8 aufgelisteten Anweisungen. Kommentarzeilen beginnen mit einem Nummernzeichen (#).

```
#
# Stammkonfigurationsdatei für hostbasierte Projekte
#
# WSED-VERSION – nicht ändern!
WSED-VERSION=7.0.0.0
# Position der Definitionsdateien für das hostbasierte Projekt angeben
PROJECT-HOME=/var/wd4z/projects
```

Abbildung 8. Konfigurationsdatei für hostbasierte Projekte `projectcfg.properties`

Die einzige zu ändernde Variable ist `PROJECT-HOME`. Dieser Wert (standardmäßig `/var/wd4z/projects`) ist das Basisverzeichnis für Projektdefinitionen.

**Anmerkung:** Für die Aktivierung hostbasierter Projekte muss das Verzeichnis `/var/wd4z/projects/USERID` eine Datei `project.instance` enthalten. Hier gibt `/var/wd4z/projects` die Position der Projektdefinitionsdateien an, und `USERID` ist die Benutzer-ID, mit der sich der Entwickler anmeldet.

Weitere Informationen zu hostbasierten Projekten enthält das White Paper *Host Based Projects in WebSphere Developer for System z Version 7.0* in der Internetbibliothek zu Developer für System z (<http://www-306.ibm.com/software/awdtools/devzseries/library/>).

---

## File Manager Integration in `FMIEXT.properties` anpassen (optional)

Developer für System z unterstützt den direkten Zugriff von einem Client auf eine begrenzte Gruppe von Funktionen von IBM File Manager für z/OS. IBM File Manager für z/OS stellt umfassende Tools für die Arbeit mit MVS-Dateigruppen und z/OS-UNIX-Dateien sowie mit DB2-, IMS- und CICS-Daten bereit. Zu diesen Tools gehören die bekannten Anzeige-, Bearbeitungs-, Kopier- und Druckdienstprogramme von ISPF, die erweitert wurden, um den Anforderungen von Anwendungsentwicklern besser gerecht zu werden. In der aktuellen Version von Developer für System z wird nur das Anzeigen/Bearbeiten von MVS-Dateigruppen (einschließlich VSAM KSDS und ESDS) unterstützt.

Beachten Sie, dass Sie das Produkt IBM File Manager für z/OS gesondert bestellen, installieren und konfigurieren müssen. Welche Version des File Manager für Ihre Version von Developer für System z erforderlich ist, können Sie der Veröffentlichung *Rational Developer für System z Hostplanung* (IBM Form GI11-3123-00) entnehmen. Die Installation und Anpassung dieses Produkts ist nicht in diesem Handbuch beschrieben.

Die von Developer für System z benötigten File-Manager-Definitionen sind in `FMIEXT.properties` gespeichert. Diese Datei befindet sich standardmäßig im Installationsverzeichnis (`/usr/lpp/wd4z/rse/lib/`). Die Datei `FMIEXT.properties` gehört allerdings zu den Dateien, die ebenfalls kopiert werden müssen, wenn Sie `rsed.envvars` in ein anderes Verzeichnis, z. B. in `/etc/wd4z/`, kopieren.



Die bereitgestellte Beispieldatei enthält die in Abb. 9 aufgelisteten Anweisungen. Kommentarzeilen beginnen mit einem Nummernzeichen (#).

```
# Erweiterungsmerkmale für File Manager Integration (FMI)
#
startup.script=/usr/lpp/wd4z/rse/lib/fmiSub
startup.port=1957
startup.range=100
startup.fmload=FMN.SFMNMOD1
startup.jobcard1=//JOB CARD JOB <Jobparameter>
startup.jobcard2=//*
startup.jobcard3=//*
startup.sysout=*
```

Abbildung 9. File-Manager-Konfigurationsdatei *FMIEXT.properties*

#### **startup.script**

Absolute Position von fmiSub im FMI-Serverstart-Script. Der Standardwert ist /usr/lpp/wd4z/rse/lib/fmiSub.

#### **startup.port**

Der erste Port, der für die Kommunikation zwischen dem FMI-Server und dem RSE-Server, der die Informationen an den Client weiterleitet, verwendet wird. Der Standard-Port ist 1957. Die Kommunikation über diesen Port ist auf Ihre Hostmaschine beschränkt.

**Anmerkung:** Überprüfen Sie vor Auswahl eines Ports, ob der Port auf Ihrem System verfügbar ist. Verwenden Sie dazu die Befehle **NETSTAT** und **NETSTAT PORTL**. Weitere Informationen hierzu enthält der Abschnitt „Reservierte TCP/IP-Ports“ auf Seite 85.

#### **startup.range**

Port-Bereich, der mit startup.port beginnt und für die FMI-Serverkommunikation verwendet wird. Die Standardeinstellung ist 100. Wenn Sie die Standardeinstellungen verwenden, kann der FMI-Server beispielsweise die Ports 1957 bis einschließlich 2056 nutzen.

#### **startup.fmload**

Absolute Position der File-Manager-Ladebibliothek. Die Standardeinstellung ist FMN.SFMNMOD1. Verwenden Sie keine Hochkommata ('), um aus dem Dateigruppennamen einen absoluten Namen zu machen. Es wird kein Präfix hinzugefügt.

**Anmerkung:** Der File Manager hat mehrere Ladebibliotheken. In dieser Konfigurationsdatei muss die Bibliothek SFMNMOD1 referenziert werden.

#### **startup.jobcard1**

#### **startup.jobcard2**

#### **startup.jobcard3**

Jobkarteninformationen für den FMI-Server. Die Standardwerte sind //JOB CARD JOB <Jobparameter>, //\* und //. Der Jobname wird durch FEK<Port> ersetzt, um die Eindeutigkeit zu gewährleisten.

#### **startup.sysout**

Sysout-Klasse für den FMI-Server. Die Standardeinstellung ist \*.



---

## Kapitel 5. IBM Common Access Repository Manager (CARMA) aktivieren (optional)

Common Access Repository Manager (CARMA) (FMID HCMA710) ist eine Produktivitätshilfe für Entwickler, die APIs für Software Configuration Manager (SCM) erstellen. Mit diesen APIs können Anwendungen (z. B. Developer für System z) auf die SCM zugreifen.

Wenn Sie mit einer früheren CARMA-Version arbeiten, sollten Sie die zugehörigen Anpassungsdateien speichern, bevor Sie Version 7.1.1 installieren. Lesen Sie hierzu den Abschnitt „Bereits konfigurierte Dateien sichern“ auf Seite 11.

Nach der Installation von CARMA müssen Sie die folgenden Konfigurationsschritte ausführen:

1. Konfigurieren Sie den CARMA-Server auf Ihrem z/OS-Host (erfordert Aktionen in MVS und z/OS UNIX).
2. Konfigurieren Sie die Beispiel-RAM (optional).
3. Schränken Sie den Zugriff auf die Initialisierungsdateien und VSAM-Cluster ein (optional). In den meisten Fällen müssen nur Systemadministratoren und CARMA-RAM-Entwickler in diese Dateien schreiben. Andere Benutzer benötigen nur Lesezugriff.

**Anmerkung:** Repository Access Manager (RAM) sind vom Benutzer geschriebene APIs für die Anbindung an z/OS Software Configuration Manager (SCM).

In Tabelle 2 auf Seite 1 sind Handbücher aufgelistet, in denen Sie weitere Informationen zu CARMA und zur Verwendung von CARMA finden.

Den Umfang der von CARMA generierten Trace-Informationen kann der Benutzer steuern, indem er auf dem Client auf der Registerseite Merkmale der CARMA-Verbindung die Trace-Stufe definiert. Folgende Optionen sind für die Trace-Stufe verfügbar:

- Protokollierung inaktivieren
- Fehler
- Warnung
- Information
- Debug

Die Standardeinstellung ist  
Fehler

. Weitere Informationen zur Position der Protokolldateien enthält der Abschnitt „Position von Protokolldateien“ auf Seite 79.

---

## CARMA-MVS-Komponenten anpassen

Alle in diesem Abschnitt enthaltenen Verweise auf HLQ beziehen sich auf den während der Installation von CARMA verwendeten High Level Qualifier. Die Standardeinstellung für die Installation ist CRA, die jedoch nicht für Ihren Standort zutreffen muss.

**Anmerkung:** In Version 7.1 wurden neue Nachrichten zur VSAM für CARMA-Nachrichten CRAMSG hinzugefügt. Sie sollten ihre bisherige VSAM für Nachrichten aktualisieren. Außerdem hat sich in Version 7.1 der Name des Beispiel-Skeleton-RAM geändert, was eine Änderung in der VSAM für CARMA-Konfiguration CRADEF zur Folge hat. Diese VSAM muss nur aktualisiert werden, wenn Sie vorhaben, das Skeleton-RAM zu verwenden.

Führen Sie zum Konfigurieren Ihres MVS-Hosts die folgenden Schritte aus:

1. Kopieren Sie die anzupassenden Member aus dem Installationsverzeichnis in eine persönliche Bibliothek, und passen Sie die Kopien an, um das Überschreiben der Member bei einer Wartung zu verhindern.
  - HLQ.SCRACLST(CRASUBMT)
  - HLQ.SCRASAMP(CRA\$VDEF)
  - HLQ.SCRASAMP(CRA\$VMSG)
  - HLQ.SCRASAMP(CRA\$VSTR)
2. Passen Sie die CLIST HLQ.SCRACLST(CRASUBMT) an. Anpassungsanweisungen finden Sie in der in CRASUBMT enthaltenen Dokumentation. Die CLIST CRASUBMT übergibt einen CARMA-Server.

**Anmerkung:** Wenn Sie möchten, können Sie das CARMA-Zeitlimit ändern. Dazu müssen Sie die Zeile PROC 1 PORT TIMEOUT(420) in der CLIST HLQ.SCRACLST(CRASUBMT) ändern. Das Zeitlimit gibt die Zeit (in Sekunden) an, die CARMA auf den nächsten Befehl vom Client wartet. Wenn Sie den Wert 0 festlegen, wird das Standardzeitlimit verwendet, das derzeit bei 420 Sekunden (7 Minuten) liegt.

3. Passen Sie die JCL in HLQ.SCRASAMP(CRA\$VDEF) an, und übergeben Sie sie. Anpassungsanweisungen finden Sie in der in CRA\$VDEF enthaltenen Dokumentation.

**Anmerkung:** Mit der JCL in CRA\$VDEF können Sie später den VSAM-Cluster CRADEF (CARMA-Konfiguration) aktualisieren. Wenn Sie den Cluster aktualisieren möchten, muss die DD-Anweisung INPUT anstatt auf CRAINIT auf die von Ihnen ausgewählte sequenzielle Datei zeigen. Weitere Informationen zum Definieren dieser sequenziellen Datei enthält der *Rational Developer für System z Common Access Repository Manager Developer's Guide* (IBM Form SC23-7660).

4. Passen Sie die JCL in HLQ.SCRASAMP(CRA\$VMSG) an, und übergeben Sie sie. Anpassungsanweisungen finden Sie in der in CRA\$VMSG enthaltenen Dokumentation. CRA\$VMSG erstellt die VSAM für CARMA-Nachrichten CRAMSG und setzt die erforderlichen Daten ein.
5. Passen Sie die JCL in HLQ.SCRASAMP(CRA\$VSTR) an, und übergeben Sie sie. Anpassungsanweisungen finden Sie in der in CRA\$VSTR enthaltenen Dokumentation.

**Anmerkung:** Mit der JCL in CRA\$VSTR können Sie später den VSAM-Cluster CRASTRS (angepasste CARMA-Informationen) aktualisieren. Wenn Sie den Cluster aktualisieren möchten, muss die Anweisung INPUT DD anstatt auf CRASINIT auf die von Ihnen ausgewählte sequenzielle Datei zeigen. Weitere Informationen zum Definieren dieser sequenziellen Datei finden Sie im *Rational Developer für System z Common Access Repository Manager Developer's Guide* (IBM Form SC23-7660).

---

## CARMA-z/OS-UNIX-Komponenten anpassen

Falls Sie sich nicht mit z/OS UNIX auskennen, sollten Sie einen erfahrenen z/OS-UNIX-Administrator oder anderen UNIX-Administrator bitten, Sie bei der Ausführung der in diesem Abschnitt beschriebenen Tasks zu unterstützen.

Die für die aufgeführten Tasks erforderlichen z/OS-UNIX-Befehle werden hier kurz für Sie beschrieben. Weitere Informationen zu diesen Befehlen entnehmen Sie bitte der Veröffentlichung *UNIX System Services Command Reference* (IBM Form SA22-7802), sofern nichts anderes angegeben ist.

- Für die nachfolgenden Tasks wird vorausgesetzt, dass Sie aktivierter z/OS-UNIX-Benutzer sind. Zum Aktivieren können Sie den TSO-Befehl **OMVS** absetzen. Mit dem Befehl **exit** können Sie zu TSO zurückkehren.
- MVS bietet mit dem Befehl **OEDIT** die Möglichkeit, z/OS-UNIX-Dateien mit ISPF zu bearbeiten. Sie können diesen Befehl in TSO und OMVS verwenden.

Alle Pfadangaben in diesem Abschnitt, die `/usr/lpp/wd4z/` enthalten, beziehen sich auf den während der Installation von Developer für System z (nicht von CARMA) verwendeten Pfad. Der Standardpfad ist `/usr/lpp/wd4z/`; an Ihrem Standort kann jedoch ein anderer Pfad verwendet worden sein.

Führen Sie die folgenden Schritte aus, um die zusammen mit IBM Rational Developer für System z (FMID HHOP710) installierten CARMA-z/OS-UNIX-Komponenten zu konfigurieren:

1. Die Konfigurationsdatei `CRASRV.properties` muss sich in demselben Verzeichnis wie die angepasste Datei `rse.envvars` befinden. Standardmäßig sind beide Dateien im Installationsverzeichnis enthalten (Standardpfad `/usr/lpp/wd4z/rse/lib/`). Sie sollten sie jedoch in ein anderes Verzeichnis kopieren, damit Sie bei einer Wartung nicht überschrieben werden. Lesen Sie hierzu die Beschreibung im Abschnitt „Konfigurationsdatei `rse.envvars` in einem anderen Verzeichnis speichern“ auf Seite 34. In den Beispielen in diesem Handbuch hat dieses Verzeichnis den Namen `/etc/wd4z/`.

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

2. Die Beispielkonfigurationsdatei `CRASRV.properties` umfasst eine Reihe von Definitionen für Umgebungsvariablen. Die Beispielkonfigurationsdatei muss an die Standardwerte Ihres Standorts angepasst werden. Sie enthält die in Abb. 10 auf Seite 60 aufgelisteten Anweisungen. Kommentarzeilen beginnen mit einem Nummernzeichen (#).

```
# CARMA-Konfigurationsoptionen
#
port.start=5227
port.range=100
startup.script.name=/usr/lpp/wd4z/rse/lib/rexxsub
clist.dsname='HLQ.SCRACLST(CRASUBMT)'
```

Abbildung 10. CARMA-Konfigurationsdatei CRASRV.properties

#### port.start

Der erste für die Kommunikation zwischen den MVS- und z/OS-UNIX-Komponenten von CARMA verwendete Port. Der Standard-Port ist 5227. Die Kommunikation über diesen Port ist auf Ihre Hostmaschine beschränkt.

**Anmerkung:** Überprüfen Sie vor Auswahl eines Ports, ob der Port auf Ihrem System verfügbar ist. Verwenden Sie dazu die Befehle **NETSTAT** und **NETSTAT PORTL**. Weitere Informationen hierzu enthält der Abschnitt „Reservierte TCP/IP-Ports“ auf Seite 85.

#### port.range

Port-Bereich, der mit port.start beginnt und für die CARMA-Serverkommunikation verwendet wird. Die Standardeinstellung ist 100. Wenn Sie die Standardeinstellungen verwenden, kann CARMA beispielsweise die Ports 5227 bis einschließlich 5326 nutzen.

#### startup.script.name

Definiert den absoluten Pfad des REXX-Übergabe-Skripts rexxsub. Die Standardeinstellung ist /usr/lpp/wd4z/rse/lib/rexxsub. Diese REXX-Exec löst die Ausführung der CLIST CRASUBMT in MVS aus.

#### clist.dsname

Definiert die Position der CLIST CRASUBMT unter Verwendung der MVS-Referenzierungskonventionen. Eine mit Hochkommata (') angegebene Position ist eine absolute Position. Der Name der angegebenen Dateigruppe wird nicht mit dem Benutzerpräfix versehen. Die Standardeinstellung ist 'HLQ.SCRACLST(CRASUBMT)'. Die CARMA-SMP/E-Installation, die CRASUBMT erstellt, verwendet CRA als Standardwert für HLQ. Diese CLIST startet beim Öffnen einer Verbindung einen CARMA-Server.

**Anmerkung:** In Developer für System z Version 7.0 wurden der CLIST-Dateigruppenname und CLIST-Member-Name von rexxsub (Variable DSNAMES) in die Datei CRASRV.properties verschoben, so dass rexxsub nicht mehr angepasst werden muss. Sie können rexxsub im Installationsverzeichnis lassen, wenn die SMP/E-Wartung automatisch aktiviert werden soll.

---

## Beispiel-RAM (Repository Access Manager) aktivieren (optional)

Repository Access Manager (RAM) sind vom Benutzer geschriebene APIs für die Anbindung an z/OS Software Configuration Manager (SCM). Führen Sie für die Beispiel-RAM, die Sie aktivieren möchten, die Anweisungen in den folgenden Abschnitten aus.

**Anmerkung:** Die Beispiel-RAM werden zum Testen der Konfiguration Ihrer CARMA-Umgebung und für die Entwicklung Ihrer eigenen RAM bereitgestellt. Verwenden Sie die zur Verfügung gestellten Beispiel-RAM NICHT in einer Produktionsumgebung.

**Anmerkung:** Alle in diesem Abschnitt enthaltenen Verweise auf HLQ beziehen sich auf den während der Installation von CARMA verwendeten High Level Qualifier. Die Standardeinstellung für die Installation ist CRA, die jedoch nicht für Ihren Standort zutreffen muss.

Weitere Informationen zu den bereitgestellten Beispiel-RAM und zum bereitgestellten Beispiel Quellcode enthält der *Rational Developer für System z Common Access Repository Manager Developer's Guide (IBM Form SC23-7660)*.

## SCLM-RAM aktivieren

1. Kopieren Sie die anzupassenden Member aus dem Installationsverzeichnis in eine persönliche Bibliothek, und passen Sie die Kopien an, um das Überschreiben der Member bei einer Wartung zu verhindern.
  - HLQ.SCRASAMP(CRA#VSLM)
  - HLQ.SCRASAMP(CRA#ASLM)
  - HLQ.SCRACLST(CRASUBMT)
2. Passen Sie die JCL in HLQ.SCRASAMP(CRA#VSLM) an, und übergeben Sie sie. Anpassungsanweisungen finden Sie in der in CRA#VSLM enthaltenen Dokumentation. CRA#VSLM erstellt die VSAM für SCLM-RAM-Nachrichten und setzt die erforderlichen Daten ein.
3. Entfernen Sie das Kommentarzeichen vor der DD-Anweisung CRARAM2 in CRASUBMT, und geben Sie den Namen der Dateigruppe mit der VSAM für SCLM-RAM-Nachrichten an. Denken Sie daran, dass Sie CRASUBMT gemäß den Anweisungen im Abschnitt „CARMA-MVS-Komponenten anpassen“ auf Seite 58 angepasst haben.
4. Passen Sie die JCL in HLQ.SCRASAMP(CRA#ASLM) an. Anpassungsanweisungen finden Sie in der in CRA#ASLM enthaltenen Dokumentation. CRA#ASLM legt die für SCLM-RAM-Clients erforderlichen Dateigruppen an.

**Anmerkung:** Vor der Verwendung von CARMA mit dem SCLM-RAM muss jeder Benutzer CRA#ASLM einmal übergeben. Andernfalls kommt es zu einem Zuordnungsfehler.

## PDS-RAM aktivieren

1. Kopieren Sie die anzupassenden Member aus dem Installationsverzeichnis in eine persönliche Bibliothek, und passen Sie die Kopien an, um das Überschreiben der Member bei einer Wartung zu verhindern.
  - HLQ.SCRASAMP(CRA#VPDS)
  - HLQ.SCRACLST(CRASUBMT)
2. Passen Sie die JCL in HLQ.SCRASAMP(CRA#VPDS) an, und übergeben Sie sie. Anpassungsanweisungen finden Sie in der in CRA#VPDS enthaltenen Dokumentation. CRA#VPDS erstellt die VSAM für PDS-RAM-Nachrichten und setzt die erforderlichen Daten ein.
3. Entfernen Sie das Kommentarzeichen vor der DD-Anweisung CRARAM1 in CRASUBMT, und geben Sie den Namen der Dateigruppe mit der VSAM für PDS-RAM-Nachrichten an. Denken Sie daran, dass Sie CRASUBMT gemäß den Anweisungen im Abschnitt „CARMA-MVS-Komponenten anpassen“ auf Seite 58 angepasst haben.

## Skeleton-RAM aktivieren

1. Kopieren Sie die anzupassenden Member aus dem Installationsverzeichnis in eine persönliche Bibliothek und passen Sie die Kopien an, um das Überschreiben der Member bei einer Wartung zu verhindern.
  - HLQ.SCRASAMP(CRA#CRAM)
  - HLQ.SCRACLST(CRASUBMT)
2. Passen Sie die JCL in HLQ.SCRASAMP(CRA#CRAM) an, und übergeben Sie sie. Anpassungsanweisungen finden Sie in der in CRA#CRAM enthaltenen Dokumentation. CRA#CRAM kompiliert das Skeleton-RAM.
3. Fügen Sie die Ladebibliothek mit dem kompilierten Skeleton-RAM CRARAMSA zur DD-Anweisung STEPLIB von CRASUBMT hinzu. Denken Sie daran, dass Sie CRASUBMT gemäß den Anweisungen im Abschnitt „CARMA-MVS-Komponenten anpassen“ auf Seite 58 angepasst haben.

---

## Kapitel 6. IBM Software Configuration and Library Manager (SCLM) Developer Toolkit aktivieren (optional)

Das Developer Toolkit von IBM Software Configuration and Library Manager (SCLM)(FMID HSD3310) stellt die Tools bereit, mit denen das Leistungsspektrum von SCLM auch auf dem Client verfügbar gemacht werden kann. SCLM selbst ist ein hostbasierter Quellcodemanager, der im Lieferumfang von ISPF enthalten ist.

Das im Lieferumfang von Developer für System z enthaltene SCLM Developer Toolkit ist ein auf Eclipse basierendes Plug-in mit Anbindung an SCLM, das den Zugriff auf alle SCLM-Prozesse für die herkömmliche Codeentwicklung ermöglicht. Das Plug-in unterstützt auch die vollständige Java- und J2EE-Entwicklung auf der Workstation. Dazu gehört die Synchronisation mit SCLM auf dem Großrechner sowie die Build-Erstellung, die Assemblierung und das Deployment des J2EE-Codes vom Großrechner.

In Tabelle 2 auf Seite 1 sind Handbücher aufgelistet, in denen Sie weitere Informationen zum SCLM Developer Toolkit sowie zur Installation, Anpassung und Verwendung des Toolkit finden.





---

## Kapitel 7. Hinweise zur Clientkomponente von Developer für System z

Benutzer der Clientkomponente von Developer für System z müssen das Ergebnis bestimmter Hostanpassungen, z. B. der TCP/IP-Port-Nummern, kennen, damit der Client fehlerfrei funktioniert. Die erforderlichen Informationen können Sie der Prüfliste in Tabelle 12 entnehmen.

*Tabelle 12. Prüfliste für die Clientkomponente von Developer für System z*

Anpassung	Wert
Port-Nummer von JES Job Monitor Server (Standard: 6715)  Lesen Sie die Beschreibung für SERV_PORT im Abschnitt „Konfigurationsdatei für JES Job Monitor (FEJJCNCFG) anpassen“ auf Seite 16.	
Position der ELAXF*-Prozeduren, falls sie nicht in einer Prozedurenbibliothek des Systems enthalten sind  Lesen Sie die Anmerkung zu JCLLIB im Abschnitt „ELAXF*-Prozeduren für ferne Build-Erstellung anpassen“ auf Seite 22.	
Namen der ELAXF*-Prozeduren und/oder der zugehörigen Prozedurschritte, sofern sie geändert wurden  Lesen Sie die Anmerkung zur Änderung der Namen im Abschnitt „ELAXF*-Prozeduren für ferne Build-Erstellung anpassen“ auf Seite 22.	
Name der gespeicherten DB2-Prozedur (Standard: ELAXMSAM)  Lesen Sie die Informationen zu gespeicherten DB2-Prozeduren in Anhang A, „Mehrere Instanzen von Developer für System z ausführen“, auf Seite 75.	
Position der gespeicherten DB2-Prozedur, sofern sie nicht in einer Prozedurenbibliothek des Systems enthalten ist  Lesen Sie hierzu den Abschnitt „ELAXM*-Member für gespeicherte DB2-Prozedur anpassen (optional)“ auf Seite 27.	
Für RSE verwendete Verbindungsmethode (Dämon, REXEC oder SSH)  Lesen Sie hierzu den Abschnitt „INETD-Dämon und -RExec/SSH für RSE konfigurieren“ auf Seite 43.	
TCP/IP-Port-Nummer des RSE-Dämons (Standard: 4035)  Lesen Sie hierzu den Abschnitt „INETD-RSE-Dämon konfigurieren“ auf Seite 43.	
Pfad zum Shell-Script server.zseries für REXEC/SSH (Standard: /usr/lpp/wd4z/rse/lib; empfohlener Pfad: /etc/wd4z)  Lesen Sie hierzu den Abschnitt „INETD-RExec (oder -SSH) konfigurieren“ auf Seite 45.	
RExec- oder SSH-Port-Nummer (Standard 512 bzw. 22)  Lesen Sie hierzu den Abschnitt „INETD-RExec (oder -SSH) konfigurieren“ auf Seite 45. <b>Anmerkung:</b> Ferne (hostbasierte) Aktionen für z/OS-UNIX-Unterprojekte erfordern, dass auf dem Host REXEC oder SSH aktiv ist.	

*Tabelle 12. Prüfliste für die Clientkomponente von Developer für System z (Forts.)*

<b>Anpassung</b>	<b>Wert</b>
Position der CRA#ASLM-JCL für das Anlegen von CARMA-SCLM-RAM-Dateigruppen  Lesen Sie die Anmerkung zu CRA#ASLM im Abschnitt „SCLM-RAM aktivieren“ auf Seite 61.	

---

## Kapitel 8. Leistungsaspekte

z/OS ist ein sehr anpassungsfähiges Betriebssystem, bei dem (manchmal kleine) Systemänderungen eine enorme Auswirkung auf die Gesamtleistung haben können. Dieses Kapitel hebt einige der Änderungen hervor, die zu einer Verbesserung der Leistung von Developer für System z führen können.

Weitere Informationen zur Systemoptimierung finden Sie im *MVS Initialization and Tuning Guide (IBM Form SA22-7591)* und in der Veröffentlichung *UNIX System Services Planning (IBM Form GA22-7800)*.

---

### Dateisystem zFS verwenden

Das zFS (zSeries File System) und das HFS (Hierarchical File System) sind UNIX-Dateisysteme, die in einer z/OS-UNIX-Umgebung verwendet werden können. Das zFS bietet jedoch die folgenden Features und Vorteile:

- Leistungssteigerung in der Umgebung vieler Kunden beim Zugriff auf Dateien mit einer Größe von annähernd 8 K, wenn die Dateien häufig aufgerufen und aktualisiert werden. Die Zugriffszeit bei kleineren Dateien entspricht der des HFS.
- Erstellen eines schreibgeschützten Klons eines Dateisystems in derselben Dateigruppe. Das geklonte Dateisystem kann Benutzern als schreibgeschützte Zeitpunktkopie eines Dateisystems bereitgestellt werden. Dieses optionale Feature ist nur in einer Nicht-Sysplex-Umgebung verfügbar.
- Das zFS ist das strategische z/OS-UNIX-Dateisystem. Die Funktionalität des HFS wurde stabilisiert. Funktionale Erweiterungen werden jedoch nur für das zFS bereitgestellt.

Wenn Sie mehr über das zFS erfahren möchten, lesen Sie die Veröffentlichung *UNIX System Services Planning (IBM Form GA22-7800)*.

---

### Verwendung von STEPLIB vermeiden

Jeder z/OS-UNIX-Prozess mit einer STEPLIB, die vom übergeordneten Element zum untergeordneten Element oder über eine Exec weitergegeben wird, belegt einen erweiterten allgemeinen Speicherbereich (ECSA, Extended Common Storage Area) von ca. 200 Bytes. Wenn die Umgebungsvariable STEPLIB nicht oder mit STEPLIB=CURRENT definiert ist, gibt z/OS UNIX alle aktiven TASKLIB-, STEPLIB- und JOBLIB-Zuordnungen während einer Funktion fork(), spawn() oder exec() weiter. Der RSE-Server startet mehrere Prozesse, und für jede Clientverbindung gibt es einen privaten RSE-Server. Die Zahlen können sich daher schnell stark summieren.

In `rsed.envvars` ist die Standardeinstellung für Developer für System z mit STEPLIB=NONE codiert. Lesen Sie hierzu die Beschreibung im Abschnitt „RSE-Konfigurationsdatei `rsed.envvars` anpassen“ auf Seite 35. Aus den oben genannten Gründen sollten Sie diese Anweisung nicht ändern und die resultierenden Dateigruppen stattdessen in die LINKLIST oder den LPA (Link-Pack-Bereich) stellen.

Wenn Sie die Steueranweisung STEPLIB nicht verwenden, müssen Sie den Inhalt der Datei `rsed.envvars` überprüfen, um festzustellen, ob Ihre STEPLIB-Anweisung die erste Anweisung ist.

- Wenn die letzte STEPLIB-Anweisung, die zuvor in `rsed.envvars` definiert wurde, `STEPLIB=NONE` lautet:  
`STEPLIB=first.steplib.dataset:second.steplib.dataset`
- Wenn die letzte STEPLIB-Anweisung, die zuvor in `rsed.envvars` definiert wurde, nicht `STEPLIB=NONE` lautet:  
`STEPLIB=$STEPLIB:first.steplib.dataset:second.steplib.dataset`

---

## Zugriff auf Systembibliotheken verbessern

Bestimmte Systembibliotheken und Lademodule werden von z/OS UNIX und Aktivitäten der Anwendungsentwicklung besonders häufig verwendet. Wenn Sie den Zugriff auf diese Bibliotheken und Module verbessern, indem Sie sie beispielsweise zum Link-Pack-Bereich (LPA) hinzufügen, können Sie die Systemleistung steigern. Weitere Informationen zu den nachfolgend beschriebenen SYS1.PARMLIB-Memberrn enthält die Veröffentlichung *MVS Initialization and Tuning Reference* (IBM Form SA22-7592).

### LE-Laufzeitbibliotheken (Language Environment)

Wenn C-Programme (einschließlich der z/OS-UNIX-Shell) ausgeführt werden, verwenden sie häufig Routinen aus der LE-Laufzeitbibliothek (Language Environment). Für jeden Adressbereich, der ein LE-fähiges Programm ausführt, werden ungefähr 4 MB der Laufzeitbibliothek in den Speicher geladen und in jede Verzweigung kopiert.

Die Dateigruppe CEE.SCEELPA enthält eine Untergruppe der LE-Laufzeitroutinen, die besonders oft von z/OS UNIX verwendet werden. Sie sollten diese Dateigruppe zu SYS1.PARMLIB(LPALSTxx) hinzufügen, um einen maximalen Leistungsgewinn zu erzielen. Wenn Sie dieser Empfehlung folgen, werden die Module nur einmal von der Platte gelesen und an einer gemeinsam genutzten Position gespeichert.

**Anmerkung:** Fügen Sie die folgende Anweisung zu SYS1.PARMLIB(PROGxx) hinzu, wenn Sie die Lademodule lieber zum dynamischen LPA (Link-Pack-Bereich) hinzufügen möchten:

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

Außerdem sollten Sie die LE-Laufzeitbibliotheken CEE.SCEERUN und CEE.SCEERUN2 in die LINKLIST stellen, indem Sie die Dateigruppen zu SYS1.PARMLIB(LNKLSTxx) oder SYS1.PARMLIB(PROGxx) hinzufügen. Auf diese Weise entfällt der z/OS-UNIX-Systemaufwand für die STEPLIB, und das Ein-/Ausgabevolumen verringert sich infolge des Managements durch LLA und VLF oder ähnliche Produkte.

**Anmerkung:** Fügen Sie aus denselben Gründen ebenfalls die C/C++-DLL-Klassenbibliothek CBC.SCLBDLL zur LINKLIST hinzu.

Wenn Sie sich entschließen, diese Bibliotheken nicht in die LINKLIST zu stellen, müssen Sie in der Datei `rsed.envvars` die entsprechende STEPLIB-Anweisung konfigurieren. Lesen Sie hierzu die Beschreibung im Abschnitt „RSE-Konfigurationsdatei `rsed.envvars` anpassen“ auf Seite 35. Obwohl diese Methode immer zusätzlichen virtuellen Speicher verwendet, können Sie die Leistung verbessern, indem Sie die LE-Laufzeitbibliotheken für LLA oder ein ähnliches Produkt definieren. Dadurch werden die Ein-/Ausgaben reduziert, die für das Laden der Module erforderlich sind.

## Anwendungsentwicklung

Auf Systemen, deren primäre Aktivität die Anwendungsentwicklung ist, kann auch eine Leistungsverbesserung erreicht werden, wenn der Linkage-Editor in den dynamischen LPA gestellt wird. Hierfür müssen die folgenden Zeilen zu SYS1.PARMLIB(PROGxx) hinzugefügt werden:

```
LPA ADD MODNAME(CEEBINIT,CEEBLIB,CEEV003,EDCZV) DSNAME(CEE.SCEERUN)
LPA ADD MODNAME(IEFIB600,IEFXB603) DSNAME(SYS1.LINKLIB)
```

Für die C/C++-Entwicklung können Sie außerdem die Compilerdateigruppe CBC.SCCNMP zu SYS1.PARMLIB(LPALSTxx) hinzufügen.

Die obigen Anweisungen sind Beispiele für mögliche LPA-Kandidaten. Die Anforderungen an Ihrem Standort können jedoch andere Maßnahmen erfordern. Informationen zur Aufnahme anderer LE-Ladmodule in den dynamischen LPA enthält die Veröffentlichung *Language Environment Customization (IBM Form SA22-7564)*. Wie Ladmodule von C/C++-Compilern in den dynamischen LPA gestellt werden, erfahren Sie in der Veröffentlichung *UNIX System Services Planning (IBM Form GA22-7800)*.

---

## Durchsatz der Sicherheitsprüfung verbessern

Wenn Sie den Durchsatz der für z/OS UNIX durchgeführten Sicherheitsprüfung verbessern möchten, definieren Sie in der Klasse FACILITY Ihrer Sicherheitssoftware das Profil BPX.SAFFASTPATH. Dadurch wird für ein breites Spektrum von Operationen der Systemaufwand für die z/OS-UNIX-Sicherheitsprüfungen verringert, z. B. für die Überprüfung des Dateizugriffs und des IPC-Zugriffs sowie für die Überprüfung der Eigentumsrechte an Prozessen. Weitere Informationen zu diesem Profil finden Sie in der Veröffentlichung *UNIX System Services Planning (IBM Form GA22-7800)*.

**Anmerkung:** Benutzer benötigen keine Berechtigung für das Profil BPX.SAFFASTPATH.

---

## Gemeinsame Klassennutzung durch mehrere JVMs

Die IBM Java Virtual Machine (JVM) bietet ab Version 5 die Möglichkeit, dass JVMs die Bootstrap-Klassen und Anwendungsklassen gemeinsam nutzen können, indem sie sie in einem Cache innerhalb des gemeinsam genutzten Speichers ablegt. Bei der gemeinsamen Nutzung von Klassen verwenden mehrere JVMs einen Cache gemeinsam, so dass insgesamt weniger virtueller Speicher belegt wird. Die gemeinsame Klassennutzung verkürzt außerdem die Startzeit für eine JVM, nachdem der Cache erstellt wurde.

Der Cache für gemeinsam genutzte Klassen ist von den aktiven JVMs unabhängig und bleibt über die Lebensdauer der JVM hinweg bestehen, die den Cache erstellt hat. Da der Cache für gemeinsam genutzte Klassen länger bestehen bleibt als jede JVM, wird er durch dynamische Aktualisierungen an alle Änderungen angepasst, die ggf. an JARs oder Klassen im Dateisystem vorgenommen wurden.

Der Systemaufwand für das Erstellen eines neuen Cache und das Füllen des Cache mit Daten ist minimal. Das Starten einer einzelnen JVM dauert im Vergleich zur gemeinsamen Nutzung von Klassen in der Regel 0 bis 5 % länger. Der genaue Unterschied im Zeitaufwand hängt davon ab, wie viele Klassen geladen werden. Bei einem mit Daten gefüllten Cache verkürzt sich die Startzeit für eine JVM im Vergleich zu einem System ohne gemeinsame Klassennutzung normalerweise um

10 bis 40 %. Die tatsächliche Beschleunigung ist vom Betriebssystem und von der Anzahl der geladenen Klassen abhängig. Bei mehreren gleichzeitig aktiven JVMs macht sich die Reduzierung der Gesamtstartzeit deutlicher bemerkbar.

Wenn Sie mehr über die gemeinsame Nutzung von Klassen erfahren möchten, lesen Sie den *Java SDK and Runtime Environment User Guide*.

## Gemeinsame Klassennutzung aktivieren

Wenn Sie die gemeinsame Klassennutzung für den RSE-Server aktivieren möchten, entfernen Sie das Kommentarzeichen für die folgende Steueranweisung in `rse.d.envvars`. Die diesbezügliche Beschreibung finden Sie im Abschnitt „Zusätzliche Java-Startparameter mit `_RSE_*OPTS` definieren (optional)“ auf Seite 41. Die erste Anweisung definiert einen Cache mit dem Namen RSE und mit Gruppenzugriff. Sie ermöglicht den Start des RSE-Servers, auch wenn die gemeinsame Klassennutzung scheitert. Die zweite Anweisung ist optional und setzt die Cachegröße auf 6 Megabytes. (Der Systemstandardwert liegt bei 16 MB.)

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

**Anmerkung:** Wie im Abschnitt „Cachesicherheit“ erwähnt, müssen alle Benutzer, die die gemeinsam genutzte Klasse verwenden, dieselbe primäre Gruppen-ID (GID) haben. Das bedeutet, dass in der Sicherheitssoftware dieselbe Standardgruppe für die Benutzer definiert sein muss bzw. dass verschiedene Standardgruppen in den OMVS-Segmenten der Benutzer dieselbe GID haben.

## Begrenzung der Cachegröße

Die theoretische maximale Größe des gemeinsam genutzten Cache liegt bei 2 GB. Die Cachegröße, die Sie angeben können, wird durch den auf dem System verfügbaren physischen Hauptspeicher und den verfügbaren Auslagerungsspeicher begrenzt. Da der virtuelle Adressbereich eines Prozesses sowohl vom Cache für gemeinsam genutzte Klassen als auch vom Java-Heap-Speicher verwendet wird, führt eine Erhöhung der maximalen Java-Heap-Größe dazu, dass Sie einen entsprechend kleineren Cache für gemeinsam genutzte Klassen erstellen können.

## Cachesicherheit

Der Zugriff auf den Cache für gemeinsam genutzte Klassen wird durch Berechtigungen des Betriebssystems und Java-Sicherheitsberechtigungen beschränkt.

Standardmäßig wird für die Erstellung von Klassencaches die Sicherheit auf Benutzerebene verwendet, so dass nur der Benutzer, der den Cache erstellt hat, auf den Cache zugreifen kann. Unter z/OS UNIX gibt es die Option `groupAccess`, die allen Benutzern Zugriff gewährt, die zur Primärgruppe des Benutzers gehören, der den Cache erstellt hat. Zerstört werden kann ein Cache unabhängig von der verwendeten Zugriffsebene nur von dem Benutzer, der ihn erstellt hat, oder von einem Benutzer `root` (UID 0).

Wenn Sie mehr über zusätzliche Sicherheitsoptionen bei Verwendung eines Java-SecurityManager erfahren möchten, lesen Sie den *Java SDK and Runtime Environment User Guide*.

## SYS1.PARMLIB(BPXPRMxx)

Einige der Einstellungen von SYS1.PARMLIB(BPXPRMxx) wirken sich bei gemeinsam genutzten Klassen auf den Durchsatz aus. Falsche Einstellungen können dazu führen, dass die gemeinsam genutzten Klassen nicht funktionieren. Diese Einstellungen können sich auch auf die Leistung auswirken. Weitere Informationen zur Verwendung dieser Parameter und zu ihrer Auswirkung auf die Leistung enthalten die Veröffentlichungen *MVS Initialization and Tuning Reference* (IBM Form SA22-7592) und *UNIX System Services Planning* (IBM Form GA22-7800). Die hinsichtlich der Verarbeitung gemeinsam genutzter Klassen wichtigsten Parameter sind die BPXPRMxx-Parameter.

- MAXSHAREPAGES, IPCSHMPAGES, IPCSHMPAGES und IPCSHMSEGS

Diese Einstellungen beeinflussen, wie viele gemeinsam genutzte Speicherseiten der JVM zur Verfügung stehen. Für einen z/OS-UNIX-Systemservice (31 Bit) hat die gemeinsam genutzte Seite eine feste Größe von 4 KB. Gemeinsam genutzte Klassen versuchen standardmäßig, einen Cache mit einer Größe von 16 MB zu erstellen. Sie sollten IPCSHMPAGES deshalb auf einen Wert größer als 4096 setzen.

Wenn Sie die Cachegröße mit -Xscmx festlegen, rundet die JVM den Wert auf das nächste volle Megabyte auf. Berücksichtigen Sie dies, wenn Sie IPCSHMPAGES auf Ihrem System setzen.

- IPCSEMNIDS und IPCSEMNSEMS

Diese Einstellungen beeinflussen, wie viele Semaphore für UNIX-Prozesse zur Verfügung stehen. Gemeinsam genutzte Klassen verwenden für die Kommunikation zwischen JVMs IPC-Semaphore.

## Plattenspeicherplatz

Der Cache für gemeinsam genutzte Klassen benötigt zum Speichern von Kennungsdaten der auf dem System vorhandenen Caches Plattenspeicherplatz. Diese Daten werden unter /tmp/javasharedresources gespeichert. Wenn das Verzeichnis mit den Kennungsdaten gelöscht wird, kann die JVM nicht die gemeinsam genutzten Klassen auf dem System identifizieren und muss den Cache neu erstellen.

## Dienstprogramme für Cacheverwaltung

Der Java-Zeilenbefehl -Xshareclasses kann mit verschiedenen Optionen verwendet werden, zu denen auch Dienstprogramme für Cacheverwaltung gehören. Drei dieser Dienstprogramme sind im folgenden Beispiel enthalten. (\$ ist die z/OS-UNIX-Eingabeaufforderung.) Eine vollständige Übersicht über die unterstützten Befehlszeilenoptionen finden Sie im *Java SDK and Runtime Environment User Guide*.

```
$ java -Xshareclasses:listAllCaches
Shared Cache      OS shmid      in use      Last detach time
RSE               401412       0           Mon Jun 18 17:23:16 2007
```

Could not create the Java virtual machine.

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

Current statistics for cache "RSE":

```
base address      = 0x0F300058
end address       = 0x0F8FFFF8
allocation pointer = 0x0F4D2E28

cache size        = 6291368
free bytes        = 4355696
ROMClass bytes    = 1912272
```



```

Metadata bytes      = 23400
Metadata % used     = 1%

# ROMClasses        = 475
# Classpaths        = 4
# URLs              = 0
# Tokens            = 0
# Stale classes      = 0
% Stale classes      = 0%

```

Cache is 30% full

Could not create the Java virtual machine.

```

$ java -Xshareclasses:name=RSE,destroy
JVMSHRC010I Shared Cache "RSE" is destroyed
Could not create the Java virtual machine.

```

**Anmerkung:** Cachedienstprogramme führen die erforderliche Operation für den angegebenen Cache aus, ohne die JVM zu starten. Die Nachricht "Could not create the Java virtual machine." ist daher normal.

**Anmerkung:** Ein Cache kann nur zerstört werden, wenn alle JVMs, die den Cache benutzen, beendet sind und der Benutzer über ausreichende Berechtigungen verfügt.

---

## Feste Java-Heap-Größe

Bei einem Heap-Speicher fester Größe gibt es keine Erweiterung oder Verkleinerung, was in bestimmten Situationen zu einer deutlichen Leistungssteigerung führen kann. Generell ist die Verwendung eines Heap-Speichers mit fester Größe jedoch keine gute Idee, weil sie den Start der Garbage-Collection hinauszögert, bis der Heap-Speicher voll ist. Die dann ausgeführte Garbage-Collection ist dementsprechend umfangreich. Außerdem steigt das Fragmentierungsrisiko, so dass eine Heap-Komprimierung erforderlich ist. Heap-Speicher mit fester Größe sollten Sie daher nur nach gründlichen Tests bzw. unter Anleitung des IBM Support Center verwenden. Weitere Informationen zu Heap-Größen und Garbage-Collections enthält der *Java Diagnostics Guide (IBM Form SC34-6650)*.

Der Heap-Speicher einer z/OS-JVM hat standardmäßig eine Anfangsgröße von einem Megabyte. Die maximale Größe liegt bei 64 Megabytes. Die Grenzwerte können Sie mit den Java-Befehlszeilenoptionen `-Xms` (Anfangsgröße) und `-Xmx` (maximale Größe) setzen.

In Developer für System z sind Java-Befehlszeilenoptionen in der Steueranweisung `_RSE_JAVA_OPTS` der Datei `rsed.envvars` definiert. Eine diesbezügliche Beschreibung finden Sie im Abschnitt „Zusätzliche Java-Startparameter mit `_RSE_*OPTS` definieren (optional)“ auf Seite 41.

```
#_RSE_JAVA_OPTS="_RSE_JAVA_OPTS -Xms128m -Xmx128m"
```



---

## Workload-Management

An jedem Standort gelten ganz bestimmte Anforderungen. Das Betriebssystem z/OS kann so angepasst werden, dass die verfügbaren Ressourcen optimal genutzt werden, um diese Anforderungen zu erfüllen. Beim Workload-Management definieren Sie Leistungsziele und ordnen jedem dieser Ziele eine geschäftliche Bedeutung zu. Sie definieren Arbeitsziele mit Geschäftsbegriffen, und das System entscheidet, wie viele Ressourcen (z. B. CPU und Speicher) der Arbeit zugeordnet werden müssen, um das angestrebte Ziel zu erreichen.

Indem Sie für die Prozesse von Developer für System z die richtigen Ziele festlegen, können Sie für eine ausgeglichene Leistung des Produkts sorgen. Nachfolgend sind dazu einige allgemeine Richtlinien aufgelistet.

- Ordnen Sie die APPC-Transaktion einer TSO-Leistungsgruppe zu.
- Ordnen Sie den RSE-Server einer TSO-Leistungsgruppe zu oder - bei starker TSO-Nutzung - einer direkt untergeordneten Gruppe.

Weitere Informationen zu diesem Thema finden Sie in der Veröffentlichung *MVS Planning Workload Management (IBM Form SA22-7602)*.



---

## Anhang A. Mehrere Instanzen von Developer für System z ausführen

In bestimmten Situationen, z. B. beim Testen eines Upgrades, kann die Ausführung mehrerer aktiver Instanzen von Developer für System z auf demselben System erwünscht sein. Manche Ressourcen können jedoch nicht gemeinsam genutzt werden, z. B. TCP/IP-Ports, so dass die Standardeinstellungen nicht immer anwendbar sind. Anhand der Informationen in diesem Anhang können Sie die Koexistenz verschiedener Instanzen von Developer für System z planen, um sie dann gestützt auf dieses Konfigurationshandbuch anzupassen.

Die gemeinsame Nutzung bestimmter Komponenten von Developer für System z durch zwei (oder mehr) Instanzen ist zwar möglich, wird jedoch NUR empfohlen, wenn die Softwareversionen identisch sind und es außer Änderungen an Konfigurations-Memberrn keine weiteren Änderungen gibt. Developer für System z bietet genug Anpassungsspielraum für die Erstellung mehrerer Instanzen ohne Überschneidung. Wir raten Ihnen dringend, diese Anpassungsfeatures zu nutzen.

---

### Identische Softwareversionen mit unterschiedlichen Konfigurationsdateien

Unter ganz bestimmten Umständen können Sie (fast) alle anpassbaren Komponenten gemeinsam nutzen. Eines der Beispiele ermöglicht für die Nutzung vor Ort den Zugriff ohne SSL und für die Nutzung an einem anderen Standort die mit SSL verschlüsselte Kommunikation.

**Achtung:** Mit der gemeinsam genutzten Konfiguration ist es NICHT möglich, ein Wartungsrelease, eine technische Vorschau oder ein neues Release sicher zu testen.

Wenn Sie eine andere Instanz einer aktiven Installation von Developer für System z konfigurieren möchten, führen Sie erneut die Anpassungsschritte für die Komponenten aus, die verschieden sind. Verwenden Sie dazu verschiedene Dateigruppen/Verzeichnisse/Ports, um Überschneidungen mit der aktuellen Konfiguration zu vermeiden.

In dem oben erwähnten SSL-Beispiel können die aktuellen RSE-Serveranpassungen geklont werden. Im Anschluss daran können Sie die geklonte Datei `ssl.properties` aktualisieren. Die MVS-Anpassungen (JES Job Monitor usw.) können von SSL-Instanzen und Nicht-SSL-Instanzen gemeinsam genutzt werden. Dies würde die folgenden Aktionen erforderlich machen:

1. Erstellen Sie ein neues Verzeichnis für die an SSL angepassten Konfigurations-Member.
2. Kopieren Sie die aktiven Konfigurations-Member in dieses Verzeichnis.
3. Aktualisieren Sie die kopierte Datei `ssl.properties` mit den SSL-bezogenen Informationen.
4. Definieren Sie in `/etc/services` eine neue RSE-Dämon-Port-Nummer.
5. Definieren Sie in `/etc/inetd.conf` einen neuen RSE-Dämonprozess. Verwenden Sie für den Parameter `-d` das neue Verzeichnis.
6. Starten Sie INETD neu, um den neuen RSE-Dämon zu aktivieren.

---

## Alle anderen Situationen

Wenn Codeänderungen vorgenommen werden müssen (Wartungsrelease, technische Neuentwicklungen, neues Release) oder Ihre Änderungen ziemlich komplex sind, sollten Sie Developer für System z neu installieren. In diesem Abschnitt sind mögliche Konfliktpunkte zwischen den verschiedenen Installationen beschrieben.

Die folgende Liste gibt Ihnen einen kurzen Überblick über die Elemente, die bei den Instanzen von Developer für System z unbedingt verschieden sein sollten oder müssen:

- Installationsbibliotheken
- TCP/IP-Port von JES Job Monitor und die zugehörige Konfigurationsdatei FEJJC�FG
- Start-JCL für JES Job Monitor
- APPC-Transaktionsname
- RSE-Konfigurationsdatei rsed.envvars
- TCP/IP-Port des RSE-Dämons

Die einzelnen Elemente sind in der folgenden Übersicht detaillierter beschrieben.

- Installationsbibliotheken
  1. Installieren Sie jede Instanz von Developer für System z in gesonderten Dateigruppen und Verzeichnissen. Denken Sie daran, dass Sie den z/OS-UNIX-Pfad nur ändern können, indem Sie den IBM Standardpfad /usr/lpp/wd4z mit einem Präfix versehen. Ein gültiges Beispiel ist /service/usr/lpp/wd4z.
- Obligatorische Komponenten
  1. Die Konfigurationsdatei von JES Job Monitor, HLQ.SFEKSAMP(FEJJC�FG), enthält die TCP/IP-Port-Nummer von JES Job Monitor und kann deshalb nicht gemeinsam genutzt werden. Der Member selbst kann umbenannt werden (sofern die JCL ebenfalls aktualisiert wird). Sie können somit alle angepassten Versionen dieses Members in eine Dateigruppe stellen, wenn Sie die Updates nicht in der Installationsdateigruppe ausführen.
  2. Die Start-JCL für JES Job Monitor, HLQ.SFEKSAMP(FEJJJCL), verweist auf FEJJC�FG und kann daher auch nicht gemeinsam genutzt werden. Nach der Umbenennung des Members (und der JOB-Karte) können Sie die gesamte JCL in dieselbe Dateigruppe stellen.
  3. Die APPC-Transaktion enthält einen Verweis auf den TSO Commands Server, HLQ.SFEKPROC(FEKFRSRV). Dieser ist von der Softwareversion abhängig, so dass Sie pro Instanz eine APPC-Transaktion erstellen müssen. Ab Version 7.1 können Sie auch über eine Funktion des SCLM Developer Toolkit auf den TSO Commands Server zugreifen.
  4. Die Konfigurationsdatei ISPF.conf (des SCLM Developer Toolkit) enthält einen Verweis auf den TSO Commands Server, HLQ.SFEKPROC(FEKFRSRV). Dieser ist von der Softwareversion abhängig, so dass Sie pro Instanz eine Datei ISPF.conf erstellen müssen. Dies ist nur erforderlich, wenn Sie über das SCLM Developer Toolkit auf den TSO Commands Server zugreifen wollen.
  5. Die RSE-Konfigurationsdatei /usr/lpp/wd4z/rse/lib/rsed.envvars enthält Verweise auf den Installationspfad und den APPC-Transaktionsnamen und muss deshalb eindeutig sein. Der Dateiname ist obligatorisch, so dass Sie die verschiedenen Kopien nicht in demselben Verzeichnis speichern können. Denken Sie daran, dass die Variable \_FEKFSCMD\_TP\_NAME\_ in rsed.envvars definiert werden muss, weil sich der APPC-Transaktionsname geändert hat.

6. Alle obligatorischen Konfigurationsdateien und Shell-Scripts, die sich in demselben Verzeichnis wie die Datei `rsed.envvars` befinden müssen, können nicht gemeinsam genutzt werden, da sich die Datei `rsed.envvars` an einer nicht gemeinsam genutzten Position befinden muss.
  7. Der REXEC- und der SSH-TCP/IP-Port können ohne Einschränkungen gemeinsam genutzt werden.
  8. Der RSE-Dämon kann nicht gemeinsam genutzt werden, da jeder RSE-Dämon einen eigenen TCP/IP-Port hat. Sie müssen einen neuen Dämon mit einem anderen Servicenamen und einem anderen Port erstellen.
- Optionale Komponenten
    1. Einige ELAXF\*-Prozeduren verweisen auf die Ladebibliothek `HLQ.SFEKLOAD` von Developer für System z. Lesen Sie im Abschnitt „ELAXF\*-Prozeduren für ferne Build-Erstellung anpassen“ auf Seite 22 die Anmerkung zu `JCLLIB`, um eine Lösung für die Bereitstellung verschiedener Sets für den Benutzer zu finden.
    2. Zum Aktivieren von zwei Instanzen der gespeicherten DB2-Prozedur müssen die folgenden Tasks ausgeführt werden. Beachten Sie, dass die folgende Beschreibung ohne Unterstützung von IBM und ohne jede Gewährleistung bereitgestellt wird.
      - a. Kopieren Sie `HLQ.SFEKPROC(ELAXMREX)` in einen anders benannten Member, z. B. in `ELAXMRXX`.
      - b. Kopieren Sie den Beispiel-Member `HLQ.SFEKSAMP(ELAXMSAM)` in einen anders benannten Member, z. B. in `ELAXMWDZ`.
      - c. Ändern Sie den Beispiel-Member `HLQ.SFEKSAMP(ELAXMJCL)` so, dass er diese Namensänderungen widerspiegelt. Beispiel:
 

```
//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. Fahren Sie mit der im Abschnitt „ELAXM\*-Member für gespeicherte DB2-Prozedur anpassen (optional)“ auf Seite 27 beschriebenen Anpassung fort, verwenden Sie jetzt jedoch die neuen Member.
      - e. Im Assistenten für gespeicherte DB2-Prozeduren auf dem Client muss der neue WLM-Umgebungsname (z. B. `ELAXMWDZ`) verwendet werden.
    3. Die BIDI-Einstellungen basieren auf einem Member der Ladebibliothek und können daher nicht releaseübergreifend gemeinsam genutzt werden. Wenn der Name des Lademoduls jedoch für alle Instanzen derselbe ist, können Sie die neueste Version in allen Instanzen, ja sogar releaseübergreifend, nutzen. Die Abwärtskompatibilität ist nicht gegeben, wenn der Name des Lademoduls geändert wurde.
    4. Die ADM-Lademodule sind abwärtskompatibel, so dass die neueste Version releaseübergreifend gemeinsam genutzt werden kann.

5. ADM-CRD-VSAM ist abwärtskompatibel. Die neueste Version kann demzufolge releaseübergreifend gemeinsam genutzt werden.
6. Die ADM-CICS-Ressourcendefinitionen sind abwärtskompatibel, so dass die neueste Version releaseübergreifend gemeinsam genutzt werden kann.
7. Alle optionalen Konfigurationsdateien und Shell-Scripts, die sich in demselben Verzeichnis wie die Datei `rsed.envvars` befinden müssen, können nicht gemeinsam genutzt werden, da sich die Datei `rsed.envvars` an einer nicht gemeinsam genutzten Position befinden muss.
8. In `rsed.envvars` können Sie die gemeinsame Nutzung von Java-Klassen durch mehrere RSE-Server aktivieren, indem Sie die Anweisung `_RSE_JAVAOPTS` aktualisieren. Die Gruppe für gemeinsame Nutzung muss einen eindeutigen Namen haben (`-Xshareclasses:name=RSE`).
9. Die CARMA-VSAMs können sich in jeder Softwareversion ändern und sollten daher nicht gemeinsam genutzt werden.

---

## Anhang B. Konfigurationsprobleme lösen

Dieser Anhang soll Sie bei einigen allgemeinen Problemen unterstützen, die beim Konfigurieren von Developer für System z auftreten könnten.

Weitere Informationen sind im Bereich 'Support' der Website zu Developer für System z (<http://www-306.ibm.com/software/awdtools/devzseries/support/>) verfügbar. In diesem Bereich finden Sie die aktuellsten technischen Hinweise des Unterstützungsteams.

Die aktuellste Version der Dokumentation zu Developer für System z, einschließlich White Papers und anderer hilfreicher Informationen, finden Sie im Abschnitt 'Library' der Website.

Wertvolle Informationen enthält auch die z/OS-Internetbibliothek mit der Adresse <http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/>.

---

### Position von Protokolldateien

Developer für System z erstellt Protokolldateien, die Sie und das IBM Support Center bei der Feststellung und Lösung von Problemen unterstützen können. Nachfolgend sind die Protokolldateien, die erstellt werden können, übersichtlich aufgelistet. Überprüfen Sie neben diesen produktspezifischen Protokollen stets, ob das SYSLOG zugehörige Nachrichten enthält.

Nach MVS-basierten Protokollen kann über die entsprechende DD-Anweisung gesucht werden. z/OS-UNIX-basierte Protokolldateien befinden sich in folgenden Verzeichnissen:

- `home/.eclipse/RSE/USERID/`

Die meisten Protokolldateien befinden sich in `home/.eclipse/RSE/USERID/`. Home steht hier für den im OMVS-Segment des Benutzers definierten Ausgangspfad (bzw. den im Standard-OMVS-Segment definierten Ausgangspfad, falls der Benutzer kein eigenes OMVS-Segment hat). USERID ist die ID des angemeldeten Benutzers (in Großbuchstaben).

- `daemon.log` - Protokoll des Dämons
- `fa.log` - Protokoll von Fault Analyzer Integration
- `fekfivpc.log` - Protokoll des IVP-Tests `fekfivpc`
- `ffs.log` - Protokoll des FFS-Servers, der native Funktionen ausführt
- `ffsget.log` - Protokoll des Datei-Reader, der eine sequenzielle Datei oder einen PDS-Member liest
- `ffsput.log` - Protokoll des Datei-Writer, der eine sequenzielle Datei oder einen PDS-Member schreibt
- `lock.log` - Protokoll des Sperrenmanagers, der eine sequenzielle Datei oder einen PDS-Member sperrt bzw. freigibt
- `rmt_class_loader.cache.jar` - Cache der vom fernen RSE-Klassenlader geladenen Klassen
- `rsecomm.log` - Protokoll des RSE-Servers, der Befehle vom Client verarbeitet (Es kann den Java-Stack-Trace für Ausnahmen enthalten.)
- `stderr.log` - Umgeleitete Daten von der Standardfehlerausgabe `stderr`

- stdout.log - Umgeleitete Daten von der Standardausgabe stdout
- /tmp/  
/tmp/ enthält Dateien, die vor der Prüfung der Benutzer-ID erstellt wurden.
- rseDaemon.log - Protokoll des Dämons vor der Anmeldung

## Protokollierung von JES Job Monitor

- **SYSOUT DD**  
Es werden normale Operationen protokolliert. Der Standardwert in der Beispiel-JCL HLQ.SFEKSAMP(FEJJJCL) ist SYSOUT=\*.
- **SYSPRINT DD**  
Trace-Protokollierung. Der Standardwert in der Beispiel-JCL HLQ.SFEKSAMP(FEJJJCL) ist SYSOUT=\*. Der Trace wird mit dem Parameter -TV aktiviert. Weitere Details hierzu enthält der Abschnitt „Start-JCL für JES Job Monitor anpassen“ auf Seite 18.

## APPC-Transaktionsprotokollierung (TSO Commands Service)

- **SYSPRINT DD**  
Wenn das APPC-Administrationsdienstprogramm ein Profil für ein Transaktionsprogramm hinzufügt und modifiziert, wird sowohl das Profil als auch die JCL auf Syntaxfehler überprüft. Die Ausgaben dieser Phase umfassen Nachrichten zu Syntaxfehlern im TP-Profil, Verarbeitungsnachrichten des Dienstprogramms und JCL-Konvertierungsanweisungen. Die Protokollierung von Nachrichten dieser Phase wird von der Anweisung SYSPRINT DD für das Dienstprogramm ATBPDFMU gesteuert. Der Standardwert in der Beispiel-JCL HLQ.SFEKSAMP(FEAPPCC) ist SYSOUT=\*. Weitere Details hierzu enthält die Veröffentlichung *MVS Planning APPC/MVS Management (IBM Form SA22-7599)*.
- **&SYSUID.FEKFRSRV.&TPDATE.&TPTIME.LOG**  
Wenn ein TP ausgeführt wird, werden die TP-Laufzeitnachrichten, z. B. Zuordnungs- und Beendigungsnachrichten, in das Protokoll geschrieben, das vom Schlüsselwort MESSAGE\_DATA\_SET im TP-Profil genannt wird. Der Standardwert in der Beispiel-JCL HLQ.SFEKSAMP(FEAPPCC) ist &SYSUID.FEKFRSRV.&TPDATE.&TPTIME.LOG. Weitere Details hierzu enthält die Veröffentlichung *MVS Planning APPC/MVS Management (IBM Form SA22-7599)*.

**Anmerkung:** Diese Protokolldatei erscheint möglicherweise erst, wenn Sie das Schlüsselwort KEEP\_MESSAGE\_LOG(ALWAYS) zu den Transaktionsdefinitionen hinzugefügt haben. Dies ist von Ihren APPC-Transaktionsdefinitionen und den Standards an Ihrem Standort abhängig. Weitere Informationen hierzu enthält die Veröffentlichung *MVS Planning APPC/MVS Management (IBM Form SA22-7599)*.

## RSE-Protokollierung

- **home/.eclipse/RSE/USERID/**  
Die zu RSE gehörenden Komponenten erstellen mehrere Protokolldateien. Die meisten dieser Dateien befinden sich im Verzeichnis Home/.eclipse/RSE/USERID/. Home steht hier für den im OMVS-Segment des Benutzers definierten Ausgangspfad (bzw. den im Standard-OMVS-Segment definierten Ausgangspfad, falls der Benutzer kein eigenes OMVS-Segment hat). USERID ist die ID des angemeldeten Benutzers (in Großbuchstaben).
- daemon.log - Protokoll des Dämons



- `ffs.log` - Protokoll des FFS-Servers, der native MVS-Funktionen ausführt
- `ffsget.log` - Protokoll des Datei-Reader, der eine sequenzielle Datei oder einen PDS-Member liest
- `ffsput.log` - Protokoll des Datei-Writer, der eine sequenzielle Datei oder einen PDS-Member schreibt
- `lock.log` - Protokoll des Sperrenmanagers, der eine sequenzielle Datei oder einen PDS-Member sperrt bzw. freigibt
- `rmt_class_loader.cache.jar` - Cache der vom fernen RSE-Klassenlader geladenen Klassen
- `rsecomm.log` - Protokoll des RSE-Servers, der Befehle vom Client verarbeitet (Es kann den Java-Stack-Trace für Ausnahmen enthalten.)
- `stderr.log` - Umgeleitete Daten von der Standardfehlerausgabe `stderr`
- `stdout.log` - Umgeleitete Daten von der Standardausgabe `stdout`

Das in `ffs*.log`, `lock.log` und `rsecomm.log` geschriebene Datenvolumen wird von der Einstellung `debug_level` in `rsecomm.properties` gesteuert. Weitere Details hierzu enthält der Abschnitt „RSE-Trace-Konfiguration in `rsecomm.properties` anpassen (optional)“ auf Seite 54.

- **`/tmp/rsedaemon.log`**

Diese Datei enthält ein Protokoll des Dämons vor der Anmeldung.

## Protokollierung für IVP-Test `fekfivpc`

- **`home/.eclipse/RSE/USERID/fekfivpc.log`**

Ausgabe des Befehls `fekfivpc -file` (IVP-Test bezogen auf das SCLM Developer Toolkit). Home steht hier für den im OMVS-Segment des Benutzers definierten Ausgangspfad (bzw. den im Standard-OMVS-Segment definierten Ausgangspfad, falls der Benutzer kein eigenes OMVS-Segment hat). USERID ist die ID des angemeldeten Benutzers (in Großbuchstaben).

Weitere Einzelheiten hierzu enthält der Abschnitt „TSO-Commands-Service-Verbindung (bei Verwendung von SCLMDT)“ auf Seite 51.

## Protokollierung von Fault Analyzer Integration

- **`home/.eclipse/RSE/USERID/fa.log`**

Protokollierung für Fault Analyzer Integration. Home steht hier für den im OMVS-Segment des Benutzers definierten Ausgangspfad (bzw. den im Standard-OMVS-Segment definierten Ausgangspfad, falls der Benutzer kein eigenes OMVS-Segment hat). USERID ist die ID des angemeldeten Benutzers (in Großbuchstaben).

## Protokollierung von File Manager Integration

- **Serverjob von File Manager Integration**

Wenn Sie eine Verbindung zum File Manager öffnen, wird ein Serverjob mit der Benutzer-ID als Eigner gestartet. Der Name des Jobs ist FEKport. Die Angabe port steht hier für den verwendeten TCP/IP-Port.

- **SYSPRINT DD**

Die SYSPRINT-Karte des Serverjobs enthält die FMI-Serverprotokollierung.

- **`Home/.eclipse/RSE/USERID/rsecomm.log`**

Protokollierung der FMI-Kommunikation. Home steht hier für den im OMVS-Segment des Benutzers definierten Ausgangspfad (bzw. den im Standard-OMVS-

Segment definierten Ausgangspfad, falls der Benutzer kein eigenes OMVS-Segment hat). USERID ist die ID des angemeldeten Benutzers (in Großbuchstaben).

Das in diese Datei geschriebene Datenvolumen wird von der Einstellung `debug_level` in `rsecomm.properties` gesteuert. Weitere Details hierzu enthält der Abschnitt „RSE-Trace-Konfiguration in `rsecomm.properties` anpassen (optional)“ auf Seite 54.

## CARMA-Protokollierung

- CARMA-Serverjob

Wenn Sie eine Verbindung zu CARMA öffnen, startet `HLQ.SCRACLST(SCRASUBMT)` einen Serverjob `CRAport` (mit der Benutzer-ID als Eigner). Die Angabe `port` im Namen steht hier für den verwendeten TCP/IP-Port.

- CARMALOG DD

Wenn in `HLQ.SCRACLST(SCRASUBMT)` die DD-Anweisung `CARMALOG` angegeben ist, wird die CARMA-Protokollierung an diese DD-Anweisung im Serverjob umgeleitet. Andernfalls ist sie auf der `SYSPRINT`-Karte enthalten.

Der Protokollierungsumfang wird durch die Einstellung `Trace-Stufe` auf dem Client bestimmt. Weitere Details zum Festlegen der `Trace-Stufe` enthält Kapitel 5, „IBM Common Access Repository Manager (CARMA) aktivieren (optional)“, auf Seite 57.

- SYSPRINT DD

Die `SYSPRINT`-Karte des Serverjobs enthält die CARMA-Protokollierung, sofern nicht die DD-Anweisung `CARMALOG` definiert ist.

- **Home/.eclipse/RSE/USERID/rsecomm.log**

Protokollierung der CARMA-Kommunikation. `Home` steht hier für den im OMVS-Segment des Benutzers definierten Ausgangspfad (bzw. den im Standard-OMVS-Segment definierten Ausgangspfad, falls der Benutzer kein eigenes OMVS-Segment hat). `USERID` ist die ID des angemeldeten Benutzers (in Großbuchstaben).

Das in diese Datei geschriebene Datenvolumen wird von der Einstellung `debug_level` in `rsecomm.properties` gesteuert. Weitere Details hierzu enthält der Abschnitt „RSE-Trace-Konfiguration in `rsecomm.properties` anpassen (optional)“ auf Seite 54.

---

## Speicherauszugsdateien

Wenn ein Produkt anormal beendet wird, wird ein Speicherauszug zur Unterstützung der Fehlerbestimmung erstellt. Verfügbarkeit und Position dieser Speicherauszüge hängen in hohem Maße von standortspezifischen Einstellungen ab. Es ist möglich, dass sie gar nicht oder an anderen Positionen als unten angegeben erstellt werden.

## MVS-Speicherauszüge

Wenn das Programm unter MVS ausgeführt wird, überprüfen Sie die System-speicherauszugsdateien und Ihre JCL (je nach Produkt) auf die folgenden DD-Anweisungen:

- SYSABEND
- SYSMDUMP
- SYSUDUMP
- CEEDUMP
- SYSPRINT
- SYSOUT

Weitere Informationen zu diesen DD-Anweisungen sind in den Veröffentlichungen *MVS JCL Reference (IBM Form SA22-7597)* und *Language Environment Debugging Guide (IBM Form GA22-7560)* enthalten.

## Java-Speicherauszüge

Unter z/OS UNIX werden Speicherauszüge von Developer für System z durch die Java Virtual Machine (JVM) gesteuert.

Die JVM erstellt während ihrer Initialisierung eine Gruppe von Speicherauszugsagenten (SYSTDUMP und JAVADUMP). Sie können diese Speicherauszugsagenten mit der Umgebungsvariablen `JAVA_DUMP_OPTS` sowie in der Befehlszeile mit `-Xdump` außer Kraft setzen. JVM-Befehlszeilenoptionen sind in der Anweisung `_RSE_JAVAOPTS` der Datei `rsed.envvars` definiert. Ändern Sie die Speicherauszeugs-einstellungen nur auf Anweisung des IBM Support Center.

**Anmerkung:** Mit der Option `-Xdump:what` in der Befehlszeile können Sie feststellen, welche Speicherauszugsagenten nach Beendigung des Systemstarts vorhanden sind.

Folgende Arten von Speicherauszügen können erzeugt werden:

### SYSTDUMP

Java-Transaktionsspeicherauszug. Dies ist ein nicht formatierter, von z/OS generierter Speicherauszug.

Der Speicherauszug wird in eine sequenzielle MVS-Datei geschrieben, deren Name standardmäßig die Form `&userid.JVM.TDUMP.&jobname.D &date.T&time` hat oder von der Umgebungsvariablen `JAVA_DUMP_TDUMP_PATTERN` bestimmt wird. Falls Sie keine Transaktionsspeicherauszüge erstellen möchten, fügen Sie die Umgebungsvariable `IBM_JAVA_ZOS_TDUMP=NO` zur Datei `rsed.envvars` hinzu.

### CEEDUMP

LE-Speicherauszug (Language Environment). Dies ist ein Systemspeicherauszug in einer formatierten Zusammenfassung, die die Stack-Traces für jeden Thread im JVM-Prozess zusammen mit Registerinformationen und einem Kurzspeicherauszug für jedes Register anzeigt.

Der Speicherauszug wird in eine z/OS-UNIX-Datei mit dem Namen `CEEDUMP.jjjjmmdd.hhmmss.pid` geschrieben. Hier stehen `jjjjmmdd` für das aktuelle Datum, `hhmmss` für die aktuelle Uhrzeit und `pid` für die ID des aktuellen Prozesses. Die möglichen Positionen dieser Datei sind im Abschnitt „Positionen für z/OS UNIX-Speicherauszüge“ auf Seite 84 beschrieben.

### HEAPDUMP

Dies ist ein formatierter Speicherauszug (Liste) der Objekte im Java-Heap-Speicher.

Der Speicherauszug wird in eine z/OS-UNIX-Datei mit dem Namen `HEAPDUMP.jjjjmmdd.hhmmss.pid.TXT` geschrieben. Hier stehen `jjjjmmdd` für das aktuelle Datum, `hhmmss` für die aktuelle Uhrzeit und `pid` für die ID des aktuellen Prozesses. Die möglichen Positionen dieser Datei sind im Abschnitt „Positionen für z/OS UNIX-Speicherauszüge“ auf Seite 84 beschrieben.

## JAVADUMP

Dies ist eine formatierte Analyse der JVM. Sie enthält Diagnoseinformationen zur JVM und zur Java-Anwendung, z. B. Angaben zur Anwendungsumgebung, zu Threads, zum nativen Stack, zu Sperren und zum Hauptspeicher.

Der Speicherauszug wird in eine z/OS-UNIX-Datei mit dem Namen JAVADUMP.jjjjmmdd.hhmmss.pid.TXT geschrieben. Hier stehen jjjjmmdd für das aktuelle Datum, hhmmss für die aktuelle Uhrzeit und pid für die ID des aktuellen Prozesses. Die möglichen Positionen dieser Datei sind im Abschnitt „Positionen für z/OS UNIX-Speicherauszüge“ beschrieben.

Weitere Informationen zu JVM-Speicherauszügen finden Sie im *Java Diagnostics Guide (IBM Form SC34-6650)*. LE-spezifische Informationen sind im *Language Environment Debugging Guide (IBM Form GA22-7560)* enthalten.

## Positionen für z/OS UNIX-Speicherauszüge

Die JVM überprüft alle nachfolgend angegebenen Positionen auf ihr Vorhandensein und auf die Schreibberechtigungen. An der ersten verfügbaren Position werden die CEEDUMP-, HEAPDUMP- und JAVADUMP-Dateien gespeichert. Denken Sie daran, dass genug freier Platten Speicherplatz vorhanden sein muss, damit die Speicherauszugsdatei ordnungsgemäß geschrieben werden kann.

1. In der Umgebungsvariablen \_CEE\_DMPTARG angegebenes Verzeichnis, sofern ein Wert gefunden wird. Diese Variable wird in rsed.envvars gesetzt, z. B. home/.eclipse/RSE/USERID/. Home steht hier für den im OMVS-Segment des Benutzers definierten Ausgangspfad (bzw. den im Standard-OMVS-Segment definierten Ausgangspfad, falls der Benutzer kein eigenes OMVS-Segment hat). USERID ist die ID des angemeldeten Benutzers (in Großbuchstaben).
2. Das aktuelle Arbeitsverzeichnis, sofern es sich nicht um das Stammverzeichnis (/) handelt und in das Verzeichnis geschrieben werden kann
3. In der Umgebungsvariablen TMPDIR angegebenes Verzeichnis (Wenn diese Umgebungsvariable gefunden wird, gibt sie die Position eines temporären Verzeichnisses an, sofern nicht /tmp verwendet wird.)
4. Das Verzeichnis /tmp
5. Falls der Speicherauszug in keinem der oben genannten Verzeichnisse gespeichert werden kann, wird er an stderr gesendet.

---

## Programmsteuerung für RSE-Programme autorisieren

Remote Systems Explorer (RSE) ist die Komponente von Developer für System z, die Kernservices wie die Verbindung vom Client zum Host bereitstellt. Für die Ausführung von Tasks, z. B. die Umschaltung auf die Benutzer-ID des Clients, muss die Komponente programmgesteuert ausgeführt werden.

Während der SMP/E-Installation wird das Programmsteuerungsbit dort, wo es erforderlich ist, gesetzt.

Sie können dieses Ergebnis mit dem Befehl `ls -lE fekf*` überprüfen, dessen Ausgabe in etwa wie das folgende Beispiel aussieht. (\$) ist die z/OS-UNIX-Eingabeaufforderung.)

```
$ ls -lE /usr/lpp/wd4z/rse/lib/fekf*
-rwxr-xr-x -ps- 2 user group 94208 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfdir.dll
-rwxr-xr-x -ps- 2 user group 143360 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfdivp
-rwxr-xr-x --s- 2 user group 480 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfivpa
-rwxr-xr-x --s- 2 user group 342 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfivpc
-rwxr-xr-x --s- 2 user group 445 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfivpd
```

```

-rwxr-xr-x --s- 2 user group 1491 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfivpj
-rwxr-xr-x --s- 2 user group 883 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfivpr
-rwxr-xr-x --s- 2 user group 307 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfivps
-rwxr-xr-x -ps- 2 user group 139264 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekflock
-rwxr-xr-x -ps- 2 user group 196608 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfrsed
-rwxr-xr-x --s- 2 user group 42443 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfscmd

```

**Anmerkung:** Für fekfivp\* und fekfscmd müssen Sie nicht das Attribut +p angeben.

Setzen Sie die folgenden Befehle ab, wenn das Programmsteuerungsbit manuell gesetzt werden muss. Hier wird vorausgesetzt, dass Sie das Standardinstallationsverzeichnis (/usr/lpp/wd4z/rse/lib/) verwendet haben.

1. cd /usr/lpp/wd4z/rse/lib
2. extattr +p fekfdir.dll fekfivp fekflock fekfrsed

**Anmerkung:** Für die Verwendung des Befehls **extattr +p** benötigen Sie mindestens die Zugriffsberechtigung READ für das Profil BPX.FILEATTR.PROGCTL in der Klasse FACILITY Ihrer Sicherheitssoftware. Weitere Informationen hierzu finden Sie in der Veröffentlichung *UNIX System Services Planning (IBM Form GA22-7800)*.

---

## Reservierte TCP/IP-Ports

Mit dem Befehl **NETSTAT** (TSO oder z/OS UNIX) können Sie eine Übersicht der zurzeit verwendeten Ports aufrufen. Die Ausgabe dieses Befehls sieht in etwa wie eines der folgenden Beispiele aus. Die letzte Zahl in der Zeile bzw. Spalte 'Local Socket' (nach '.') gibt die verwendeten Ports an. Da diese Ports bereits genutzt werden, können sie nicht für die Konfiguration von Developer für System z verwendet werden.

### IPV4

MVS TCP/IP NETSTAT CS V1R7			TCPIP Name: TCPIP	16:36:42
User Id	Conn	Local Socket	Foreign Socket	State
-----	-----	-----	-----	-----
BPX0INIT	00000014	0.0.0.0..10007	0.0.0.0..0	Listen
INETD4	0000004D	0.0.0.0..512	0.0.0.0..0	Listen
INETD4	0000004B	0.0.0.0..4035	0.0.0.0..0	Listen
JMON	00000038	0.0.0.0..6715	0.0.0.0..0	Listen

### IPV6

MVS TCP/IP NETSTAT CS V1R7			TCPIP Name: TCPIP	12:46:25
User Id	Conn	State		
-----	----	-----		
BPX0INIT	00000018	Listen		
	Local Socket:	0.0.0.0..10007		
	Foreign Socket:	0.0.0.0..0		
INETD4	00000046	Listen		
	Local Socket:	0.0.0.0..512		
	Foreign Socket:	0.0.0.0..0		
INETD4	0000004B	Listen		
	Local Socket:	0.0.0.0..4035		
	Foreign Socket:	0.0.0.0..0		
JMON	00000037	Listen		
	Local Socket:	0.0.0.0..6715		
	Foreign Socket:	0.0.0.0..0		

Eine andere bestehende Einschränkung sind reservierte TCP/IP-Ports. Es gibt zwei allgemeine Bereiche, in denen TCP/IP-Ports reserviert werden:

- **PROFILE.TCPIP**

Auf diese Dateigruppe verweist die DD-Anweisung PROFILE der gestarteten TCP/IP-Task, die oft den Namen SYS1.TCPPARMS(TCPPROF) hat.

- PORT: Reserviert einen Port für angegebene Jobnamen.
- PORTRANGE: Reserviert einen Port-Bereich für angegebene Jobnamen.

Weitere Informationen zu diesen Anweisungen finden Sie im *Communications Server IP Configuration Guide (IBM Form SC31-8775)*.

- **SYS1.PARMLIB(BPXPRMxx)**

- INADDRANYPORT: Gibt die Nummer des ersten Ports für den Port-Bereich an, den das System für die Bindung an PORT 0 mit INADDR\_ANY reserviert. Dieser Wert wird nur für CINET benötigt.
- INADDRANYCOUNT: Gibt die Anzahl der vom System reservierten Ports an, einschließlich des mit dem Parameter INADDRANYPORT angegebenen Ports. Dieser Wert wird nur für CINET benötigt.

Weitere Informationen zu diesen Anweisungen können Sie den Veröffentlichungen *UNIX System Services Planning (IBM Form GA22-7800)* und *MVS Initialization and Tuning Reference (IBM Form SA22-7592)* entnehmen.

Diese reservierten Ports können mit dem Befehl **NETSTAT PORTL** (TSO oder z/OS UNIX) aufgelistet werden. Die erstellte Ausgabe sieht in etwa wie das folgende Beispiel aus:

```

MVS TCP/IP NETSTAT CS V1R7          TCP/IP Name: TCP/IP          17:08:32
Port# Prot User  Flags      Range      IP Address
-----
00007 TCP  MISCSERV DA
00009 TCP  MISCSERV DA
00019 TCP  MISCSERV DA
00020 TCP  OMVS     D
00021 TCP  FTPD1    DA
00025 TCP  SMTP     DA
00053 TCP  NAMESRV  DA
00080 TCP  OMVS     DA
03500 TCP  OMVS     DAR      03500-03519
03501 TCP  OMVS     DAR      03500-03519

```

Weitere Informationen zum Befehl **NETSTAT** enthält die Veröffentlichung *Communications Server IP System Administrator's Commands (IBM Form SC31-8781)*.

**Anmerkung:** Der Befehl **NETSTAT** zeigt nur die in PROFILE.TCPIP definierten Informationen an, die sich mit den Definitionen in BPXPRMxx überschneiden müssten. Überprüfen Sie im Zweifelsfall, welche Ports im PARMLIB-Member BPXPRMxx reserviert sind.

**Anmerkung:** Falls Sie Probleme bei der Bindung von INETD an reservierte Ports haben, lesen Sie den Abschnitt „Port-Definitionen in PROFILE.TCPIP“ auf Seite 106.

---

## Größe des Adressbereichs

Der RSE-Server ist ein Java-UNIX-Prozess und erfordert für seine Funktionen eine große Regionsgröße. Deshalb ist es wichtig, dass für OMVS-Adressbereiche große Speichergrenzen festgelegt werden.



## INETD-Anforderungen

INETD startet einen RSE-Server, wenn ein Client eine Verbindung zum RSE-Dämon-Port herstellt oder mit REXEC/SSH das Start-Script aufruft. Hierfür werden die Speichergrenzen von INETD verwendet, die demzufolge entsprechend groß festgelegt sein müssen.

- Wenn der UNIX-Serverprozess INETD über JCL mit BPXBATCH gestartet wird, muss die Regionsgröße gleich null sein.
- Wird INETD in einer TSO/OMVS-Sitzung gestartet, muss die Regionsgröße bei mindestens 2096128 liegen.

## In SYS1.PARMLIB(BPXPRMxx) festgelegte Begrenzungen

Setzen Sie MAXASSIZE in SYS1.PARMLIB(BPXPRMxx) (zum Definieren der Standardregionsgröße bzw. -prozessgröße für den OMVS-Adressbereich) auf mindestens 2147483647.

Dieser Wert kann mit folgenden Konsolbefehlen überprüft und dynamisch (bis zum nächsten IPL) gesetzt werden. Lesen Sie hierzu die Beschreibung in der Veröffentlichung *MVS System Commands (IBM Form SA22-7627)*.

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

## Im Sicherheitsprofil gespeicherte Begrenzungen

Überprüfen Sie ASSIZEMAX im OMVS-Segment der Clientbenutzer-ID, und setzen Sie das Feld auf 2147483647 oder vorzugsweise auf NONE, damit der Wert SYS1.PARMLIB(BPXPRMxx) verwendet wird.

Dieser Wert kann in RACF mit den folgenden TSO-Befehlen überprüft und gesetzt werden. Lesen Sie hierzu die Beschreibung in der Veröffentlichung *Security Server RACF Command Language Reference (IBM Form SA22-7687)*.

1. LISTUSER userid NORACF OMVS
2. ALTUSER userid OMVS(NOASSIZEMAX)

## Von System-Exits erzwungene Begrenzungen

Stellen Sie sicher, dass Regionsgrößen des OMVS-Adressbereichs nicht von den System-Exits IEFUSI oder IEALIMIT gesteuert werden. Eine Möglichkeit, dies zu erreichen, ist die Verwendung des Codes SUBSYS(OMVS,NOEXITS) in SYS1.PARMLIB(SMFPRMxx).

SYS1.PARMLIB(SMFPRMxx)-Werte können mit folgenden Konsolbefehlen überprüft und aktiviert werden. Lesen Sie hierzu die Beschreibung in der Veröffentlichung *MVS System Commands (IBM Form SA22-7627)*.

1. DISPLAY SMF,0
2. SET SMF=xx

---

## Trace für Fehlerrückmeldungen

Mit der folgenden Prozedur können Informationen zusammengestellt werden, die notwendig sind, um Probleme bei Fehlerrückmeldungen für ferne Build-Prozeduren zu diagnostizieren. Dieser Trace bringt Leistungseinbußen mit sich und sollte nur auf Anweisung des IBM Support Center durchgeführt werden. Alle in diesem Abschnitt enthaltenen Verweise auf HLQ beziehen sich auf den während der Instal-

lation von Developer für System z verwendeten High Level Qualifier. Die Standardeinstellung für die Installation ist FEK, die jedoch nicht für Ihren Standort zutreffen muss.

1. Erstellen Sie eine Sicherungskopie Ihrer aktiven ELAXFC0C-Kompilierungsprozedur. Standardmäßig ist diese Prozedur in der Dateigruppe HLQ.SFEKSAMP enthalten. Möglicherweise wurde sie jedoch an eine andere Position kopiert, z. B. nach SYS1.PROCLIB. Lesen Sie hierzu den Abschnitt „ELAXF\*-Prozeduren für ferne Build-Erstellung anpassen“ auf Seite 22.
2. Ändern Sie die aktive ELAXFC0C-Prozedur so, dass sie in der Kompilierungsoption EXIT(ADEXIT(ELAXMGUX)) die Zeichenfolge 'MAXTRACE' enthält.

```
//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)
```

**Anmerkung:** Sie müssen MAXTRACE in doppelte Hochkommata setzen. Die Option sieht jetzt wie folgt aus: EXIT(ADEXIT('MAXTRACE',ELAXMGUX))

3. Führen Sie eine ferne Syntaxprüfung für ein COBOL-Programm durch. Im Lieferumfang von Developer für System z ist ein Beispiel-COBOL-Programm in HLQ.SFEKSAMP(ADNSMSGH) enthalten.
4. Der Abschnitt der JES-Ausgabe beginnt mit einer Auflistung der Dateigruppennamen für SIDEFILE1, SIDEFILE2, SIDEFILE3 und 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'
```

**Anmerkung:** SIDEFILE1 und SIDEFILE2 können, abhängig von Ihren Einstellungen, auf eine DD-Anweisung zeigen (SUCCESSFUL OPEN SIDEFILE1 - NAME = DD:WSEDSF1). Den tatsächlichen Namen der Dateigruppe finden Sie im Abschnitt JESJCL der Ausgabe (der sich vor dem Abschnitt SYSOUT befindet).

```
22 //COBOL.WSEDSF1 DD DISP=MOD,
// DSN=uid.ERRCOB.member.SF1.Z682746.XML
23 //COBOL.WSEDSF2 DD DISP=MOD,
// DSN=uid.ERRCOB.member.SF1.Z682747.XML
```

5. Kopieren Sie diese vier Dateigruppen auf Ihren PC, indem Sie beispielsweise in Developer für System z ein lokales COBOL-Projekt erstellen und die Dateigruppen SIDEFILE1 bis SIDEFILE4 zu diesem Projekt hinzufügen.
6. Kopieren Sie das vollständige JES-Jobprotokoll auf Ihren PC, indem Sie beispielsweise die Jobausgabe in Developer für System z öffnen und **Datei > Speichern unter...** auswählen, um das Protokoll im lokalen Projekt zu speichern.
7. Stellen Sie den ursprünglichen Zustand der Prozedur ELAXFC0C wieder her, indem Sie die Änderung rückgängig machen (die Zeichenfolge "MAXTRACE" aus den Kompilierungsoptionen entfernen) oder die Sicherungskopie zurückschreiben.
8. Senden Sie die gesammelten Dateien (SIDEFILE1 bis SIDEFILE4 sowie das Jobprotokoll) an das IBM Support Center.



---

## APPC-Transaktion und TSO Commands Service

Falls Sie die APPC-Version des TSO Commands Service nicht nutzen können, können in zwei Bereichen Probleme auftauchen, beim Starten der APPC-Servertransaktion und beim Herstellen einer Verbindung zu RSE.

- Falls die Nachrichten zum Konfigurieren von APPC nicht angezeigt werden, überprüfen Sie, ob das Systemprotokoll RACF-Nachrichten (mit der Nachrichten-ID ICHxxxxx) oder andere Nachrichten zu dem abgesetzten Befehl enthält oder zu der Benutzer-ID, die den Befehl abgesetzt hat. Einige allgemeine Ursachen für Probleme:
  - Sie haben keine Leseberechtigung für die Dateigruppe HLQ.SFEKPROC.
  - TCP/IP ist nicht aktiv, TCP/IP wurde ein falscher DNS-Name zugeordnet, oder das System ist aufgrund von Netzproblemen, einer falschen IP-Adresse oder aus anderen Gründen nicht (mit ping) erreichbar.
- Wenn die Nachrichten zum Konfigurieren von APPC angezeigt werden, jedoch keine Bestätigung für den erfolgreichen Abschluss der Konfiguration, konnte wahrscheinlich die APPC-Servertransaktion nicht gestartet werden. Überprüfen Sie das Transaktionsfehlerprotokoll (USERID.FEKFRRV.&TPDATE.&TPTIME.LOG). Einige wahrscheinliche Problemursachen:
  - Der TCP/IP-Stack verwendet nicht den Standardnamen TCPIP, und die DD-Karte SYSTCPD wurde nicht gesetzt oder zeigt auf die falsche Dateigruppe.
  - Der Server konnte SYSPROC oder SYSTSPRT nicht zuordnen.
  - Die JCL zeigt auf die falsche SYSPROC. (SYSPROC muss HLQ.SFEKPROC enthalten.)
  - Der Server konnte die von MESSAGE\_DATA\_SET referenzierte Nachrichtendateigruppe (das Protokoll) nicht öffnen oder nicht auf die Dateigruppe zugreifen.
  - Es sind nicht genug APPC-Scheduler-Initiatoren verfügbar.
  - Der APPC- oder ASCH-Adressbereich ist nicht aktiv.
  - Die verwendete Klasse (standardmäßig die Klasse "A") ist nicht für den APPC-Scheduler ASCH definiert.
  - Es ist kein Standard-OMVS-Segment für das System vorhanden, und der Benutzer hat kein persönliches OMVS-Segment, oder im Segment gibt es einen Definitionsfehler.
  - Die Standardgruppe des Standard-OMVS-Segments oder die Standardgruppe des Benutzers hat keine GID-Nummer.

Die im Abschnitt „APPC-Transaktion für den TSO Commands Service definieren“ auf Seite 24 angegebene REXX kann Ihnen bei der Lösung von APPC-Problemen helfen, denn sie ermöglicht Ihnen, APPC in ISPF-Anzeigen im Dialogbetrieb zu verwalten. Mit diesem Tool können Sie die Transaktion inaktivieren. Die Transaktion ist jedoch unverändert vorhanden, akzeptiert dann allerdings keine Verbindungen mehr.

Nachfolgend sind technische Hinweise aufgelistet, die derzeit auf der Unterstützungswebsite (<http://www-306.ibm.com/software/awdtools/devzseries/support/>) verfügbar sind. Zusätzliche Informationen entnehmen Sie bitte direkt der Unterstützungswebsite.

- APPC-Prüfung scheitert mit Rückkehrcode 2016 - EHOSTNOTFOUND
- APPC-Prüfung scheitert mit Rückkehrcode 1004 - EIBMIUCVERR
- APPC-Prüfung scheitert mit Rückkehrcode 9 - TP-Name nicht erkannt
- APPC-Prüfung scheitert mit Rückkehrcode 10 - TP kann nicht erneut gestartet werden

- APPC-Prüfung scheitert mit Rückkehrcode 19 - Parameterfehler
- APPC-Prüfung scheitert mit Rückkehrcode 20 - Produktspezifischer Fehler
- APPC-Prüfung scheitert mit Rückkehrcode 26 - Ressourcenfehler
- CEE3501S: The module IOSTREAM was not found
- Der Server konnte nicht gestartet werden: EDC5129I No such file or directory
- exec/tcp: bind: EDC5111I Permission denied, rsn=744C7246
- Der Server antwortet nicht, und es erscheint eine der folgenden Nachrichten:
  - IEA995I SYMPTOM DUMP OUTPUT 473 USER COMPLETION CODE=4093 REASON CODE=0000001C (in SDSF LOG)
  - CEE3512S An HFS load of module libicudata32.0.dll failed. The system return code was 0000000157; the reason code was 0BDF019B. (in CEEDUMP)
  - Get Space failed (in client .log)
- Befehl C\_CONNECT nicht verfügbar
- Fehlernachricht "FFS server initialization failed" beim Herstellen der Verbindung zum Host
- Nachricht "EDC5139I Operation not permitted" beim Herstellen der Verbindung zum Host
- Nachricht "RSEG1056U FFS server initialization failed" beim Öffnen einer MVS-Datei

**Anmerkung:** Diese Liste ist vorläufig. Zusätzliche technische Hinweise entnehmen Sie bitte der Unterstützungswebsite.

---

## Weitere Informationen

### Systemgrenzwerte

SYS1.PARMLIB(BPXPRMxx) definiert viele z/OS-UNIX-bezogene Begrenzungen, die erreicht werden können, wenn mehrere Clientkomponenten von Developer für System z aktiv sind. Die meisten BPXPRMxx-Werte können mit den Konsolbefehlen **SETOMVS** und **SET OMVS** dynamisch geändert werden.

#### Verbindung verweigert

Jede RSE-Verbindung startet mehrere Prozesse, die permanent aktiv sind. Neue Verbindungen können durch den in SYS1.PARMLIB(BPXPRMxx) gesetzten Grenzwert für die Anzahl der Prozesse zurückgewiesen werden. Dies gilt insbesondere, wenn Benutzer dieselbe UID gemeinsam benutzen (wie es z. B. bei Verwendung des Standard-OMVS-Segments der Fall ist).

- Der Grenzwert pro UID wird durch das Schlüsselwort MAXPROCUSER festgelegt und liegt standardmäßig bei 25.
- Der systemweite Grenzwert wird durch das Schlüsselwort MAXPROCSYS festgelegt und liegt standardmäßig bei 200.

Eine weitere Ursache für zurückgewiesene Verbindungen ist der Grenzwert für die Menge aktiver z/OS-Adressbereiche und z/OS-UNIX-Benutzer.

- Die maximale Anzahl von Adressbereichs-IDs (ASID) wird in SYS1.PARMLIB(IEASYSxx) mit dem Schlüsselwort MAXUSER definiert. Der Standardwert liegt bei 255.
- Die maximale Anzahl von z/OS-UNIX-Benutzer-IDs (UID) wird in SYS1.PARMLIB(BPXPRMxx) mit dem Schlüsselwort MAXUIDS definiert. Der Standardwert liegt bei 200.

## Bekannte Probleme

### MVS-Dateigruppen können nicht geöffnet werden

Für das Lesen und Schreiben einer MVS-Dateigruppe muss eine Dateisystem-domäne mit physischen Sockets verwendet werden. Wenn das Dateisystem nicht ordnungsgemäß definiert ist oder nicht genug Sockets hat, verhindert der Sperrenmanager (FFS) Lese-/Schreibanforderungen. Die Dateien ffs\*.log enthalten dann Nachrichten wie die folgenden:

- Error 127 getting socket pair - setting port to 0.
- Unable to create socket in the UNIX domain. Error is: "The address family is not supported"

Prüfen Sie, ob der Member SYS1.PARMLIB(BPXPRMxx) die folgenden Anweisungen enthält:

```
FILESYSTYPE TYPE(UDS) ENTRYPPOINT(BPXTUINT)
NETWORK DOMAINNAME(AF_UNIX)
          DOMAINNUMBER(1)
          MAXSOCKETS(200)
          TYPE(UDS)
```

### DVIPA-Bindungen scheitern

Wenn Sie eine dynamische virtuelle IP-Adresse (VIPA) für mehrere TCPIP-Stacks verwenden, muss die Zuordnung ephemerer Ports für den gesamten Sysplex so koordiniert werden, dass das 4-Tupel (die Kombination aus Quellen- und Ziel-IP-Adressen und Ports) für jede Verbindung eindeutig bleibt. Zu diesem Zweck können Sie den optionalen Parameter SYSPLEXPORTS zur ersten Anweisung VIPADISTRIBUTE hinzufügen. Eine diesbezügliche Beschreibung finden Sie im *Communications Server IP Configuration Guide (IBM Form SC31-8775)*.

Wenn Sie dieses Verfahren anwenden, vergewissern Sie sich, dass die Coupling-Facility-Struktur EZBEPOR (mit Informationen zur Sysplex-Port-Zuordnung) definiert wurde. Informationen hierzu finden Sie im *Communications Server SNA Network Implementation Guide (IBM Form SC31-8777)*.

## Host-Connect-Emulator

- Der Host-Connect-Emulator verwendet für Verbindungen zum Host TN3270-Telnet und nicht den RSE-Server.
- Wenn Sie mit sicherem Telnet (SSL) arbeiten und Zertifikate verwenden, die nicht von einer anerkannten Zertifizierungsstelle signiert sind, muss jeder Client das Zertifikat der Zertifizierungsstelle zu seiner Host-Connect-Emulator-Liste anerkannter Zertifizierungsstellen hinzufügen.
- Zum Inaktivieren der funktionalen SNA-Erweiterungen benötigen Sie möglicherweise die Option NOSNAEXT für die TELNETPARMS von TCP/IP. Wenn NOSNAEXT angegeben ist, führt der TN3270-Telnet-Server keine Verhandlungen zu einer Konfliktlösung oder zu SNA-Prüffunktionen.

---

## Kontakt zur IBM Unterstützungsfunktion aufnehmen

Sollten Sie nach dem Durchlesen dieses Handbuchs und der technischen Hinweise auf der Unterstützungswebsite (<http://www-306.ibm.com/software/awdtools/devzseries/support/>) noch immer Schwierigkeiten haben und Hilfestellung von der IBM Unterstützungsfunktion benötigen, stellen Sie die folgenden Informationen in einem Problemsatz für IBM zusammen:

- Ihre aktuelle Softwareversion, die an folgenden Stellen dokumentiert ist:
  - HLQ.SFEKJCL(FEK#LVL#) für Developer für System z. Der Standardwert für HLQ ist FEK.
  - HLQ.SCRAJCL(CRA#LVL#) für CARMA. Der Standardwert für HLQ ist CRA.

Die folgenden Informationen sind für die Lösung von Verbindungsproblemen hilfreich.

- Ausgabe der folgenden Konsolbefehle:
  - D OMVS,0
- Ausgabe der folgenden TSO-Befehle:
  - HOMETEST
  - LISTUSER userid NORACF OMVS
  - LISTGROUP group NORACF OMVS
- Datei rsed.envvars, die sich standardmäßig in /usr/lpp/wd4z/rse/lib/ befindet, der Empfehlung nach jedoch in das Verzeichnis /etc/wd4z/ gestellt werden sollte
- Datei /etc/inetd.conf
- Alle Dateien \*.log in home/.eclipse/RSE/USERID/. Home steht hier für den im OMVS-Segment des Benutzers definierten Ausgangspfad (bzw. den im Standard-OMVS-Segment definierten Ausgangspfad, falls der Benutzer kein eigenes OMVS-Segment hat). USERID ist die ID des angemeldeten Benutzers (in Großbuchstaben).
- Zugehörige SYSLOG-Nachrichten
- Vollständige Ausgabe der zugehörigen IVP-Tests (siehe Abschnitt „RSE-Serverkonfiguration prüfen“ auf Seite 47).
  - netstat: siehe Abschnitt „Port-Verfügbarkeit“ auf Seite 48
  - fekfivps: siehe Abschnitt „REXEC/SSH-Shell-Script“ auf Seite 49
  - fekfivpr: siehe Abschnitt „REXEC-Verbindung“ auf Seite 49
  - fekfivpd: siehe Abschnitt „RSE-Dämonverbindung“ auf Seite 50
  - fekfivpj: siehe Abschnitt „JES-Job-Monitor-Verbindung“ auf Seite 51
  - fekfivpc: siehe Abschnitt „TSO-Commands-Service-Verbindung (bei Verwendung von SCLMDT)“ auf Seite 51
  - fekfivpa: siehe Abschnitt „TSO-Commands-Service-Verbindung (bei Verwendung von APPC)“ auf Seite 52

Bei Verwendung des SCLM Developer Toolkit für den TSO Commands Service (Standardmethode)

- Vollständige Ausgabe des SCLMDT-IVP-Tests (fekfivpc) wie im Abschnitt „TSO-Commands-Service-Verbindung (bei Verwendung von SCLMDT)“ auf Seite 51 beschrieben.
- Datei /etc/SCLMDT/CONFIG/ISPF.conf

Bei Verwendung von APPC für den TSO Commands Service

- Vollständige Ausgabe des APPC-IVP-Tests (fekfi vpa) wie im Abschnitt „TSO-Commands-Service-Verbindung (bei Verwendung von APPC)“ auf Seite 52 beschrieben.
- JCL, die die APPC-Servertransaktion definiert (bei Verwendung des Beispieljobs ist dies HLQ.SFEKSAMP(FEKAPPCC))
- Member SYS1.PARMLIB(APPCPMxx)
- Member SYS1.PARMLIB(ASCHPMxx)
- Dateigruppe userid.FEKFSERV.&TPDATE.&TPTIME.LOG
- Ausgabe der folgenden Konsolbefehle:
  - D APPC,TP,ALL
  - D APPC,LU,ALL
  - D ASCH,ALL

Stellen Sie alle Informationen bereit, die für funktionale Probleme relevant erscheinen, z. B.:

- Jobprotokollierung
- Zugehörige SYSLOG-Nachrichten
- Protokolldateien in home/.eclipse/RSE/USERID/. Home steht hier für den im OMVS-Segment des Benutzers definierten Ausgangspfad (bzw. den im Standard-OMVS-Segment definierten Ausgangspfad, falls der Benutzer kein eigenes OMVS-Segment hat). USERID ist die ID des angemeldeten Benutzers (in Großbuchstaben).



---

## Anhang C. TCP/IP konfigurieren

Dieser Anhang soll Sie bei einigen allgemeinen Problemen unterstützen, die beim Konfigurieren von TCP/IP oder beim Überprüfen und/oder Modifizieren einer vorhandenen Konfiguration auftreten könnten.

Zusätzliche Informationen zur TCP/IP-Konfiguration finden Sie im *Communications Server IP Configuration Guide (IBM Form SC31-8775)* und in der Veröffentlichung *Communications Server IP Configuration Reference (IBM Form SC31-8776)*.

---

### Abhängigkeit vom Hostnamen

Developer für System z ist bei der Initialisierung davon abhängig, dass TCP/IP mit dem richtigen Hostnamen konfiguriert ist. Dies impliziert, dass die verschiedenen TCP/IP- und Resolver-Konfigurationsdateien ordnungsgemäß definiert sein müssen.

Sie können Ihre TCP/IP-Konfiguration mit dem TSO-Befehl **HOMETEST** testen. Weitere Informationen zum Befehl **HOMETEST** enthält die Veröffentlichung *Communications Server IP System Administrator's Commands (IBM Form SC31-8781)*.

Beispielausgabe des Befehls **HOMETEST**:

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

The FTP configuration parameter file used will be "SYS1.TCPPARMS(FTPDATA)"

TCP Host Name is: CDFMVS08

Using Name Server to Resolve CDFMVS08

The following IP addresses correspond to TCP Host Name: CDFMVS08  
9.42.112.75

The following IP addresses are the HOME IP addresses defined in PROFILE.TCPIP:  
9.42.112.75  
127.0.0.1

All IP addresses for CDFMVS08 are in the HOME list!

Hometest was successful – all Tests Passed!

---

### Wissenswertes zu Auflösern

Der Auflöser arbeitet für Programme als ein Client, der für die Auflösung von Namen in Adressen oder von Adressen in Namen auf Namensserver zugreift. Für die Anforderung eines aufrufenden Programms kann der Auflöser auf verfügbare Namensserver zugreifen, lokale Definitionen verwenden (z. B. `/etc/resolv.conf`, `/etc/hosts`, `/etc/ipnodes`, `HOSTS.SITEINFO`, `HOSTS.ADDRINFO` oder `ETC.IPNODES`) oder eine Kombination aus beiden Möglichkeiten anwenden.

Beim Starten des Adressbereichs des Auflösers wird eine optionale Auflöserkonfigurationsdateigruppe gelesen, auf die die DD-Karte `SETUP` in der JCL-Prozedur des Auflösers zeigt. Wenn die Konfigurationsdaten nicht zur Verfügung stehen, greift der Auflöser auf die anwendbare native MVS- oder z/OS-UNIX-Suchreihenfolge ohne Angaben von `GLOBALTCIPDATA`, `DEFAULTTCIPDATA`, `GLOBALIPNODES`, `DEFAULTIPNODES` oder `COMMONSEARCH` zurück.

**Anmerkung:** Wenn in der Prozedurenbibliothek (PROCLIB) des Systems keine JCL-Prozedur des Auflösers vorhanden ist, beginnt der Adressbereich mit den Systemstandardwerten (keine DD-Anweisung SETUP).

---

## Wissenswertes zur Suchreihenfolge für Konfigurationsdaten

Es ist wichtig, dass Sie die von TCP/IP-Funktionen verwendete Suchreihenfolge für Konfigurationsdateien verstehen und wissen, wann Sie die Standardsuchreihenfolge mit Umgebungsvariablen, JCL oder anderen von Ihnen angegebenen Variablen außer Kraft setzen können. Ausgehend von diesen Kenntnissen können Sie Ihre Benennungsstandards für lokale Dateigruppen und HFS-Dateien anpassen. Außerdem ist es bei der Fehlerdiagnose hilfreich zu wissen, welche Konfigurationsdateigruppe oder HFS-Datei verwendet wird.

Ein anderer wichtiger Punkt ist, dass die Suche bei Anwendung einer Suchreihenfolge für Konfigurationsdateien bei der ersten gefundenen Datei beendet wird. Wenn Sie Konfigurationsdaten in eine Datei stellen, die nie gefunden wird, weil es in der Suchreihenfolge vorher eine andere Datei gibt oder die Datei nicht von der Suchreihenfolge, die die Anwendung gewählt hat, erfasst wird, kann es daher zu unerwarteten Ergebnissen kommen.

Bei der Suche nach Konfigurationsdateien können Sie TCP/IP mit DD-Anweisungen in den JCL-Prozeduren oder durch das Setzen von Umgebungsvariablen explizit mitteilen, wo sich die meisten Konfigurationsdateien befinden. Sie können TCP/IP die Position der Konfigurationsdateien aber auch dynamisch auf der Grundlage der Suchreihenfolgen ermitteln lassen. Diese Suchreihenfolgen sind im *Communications Server IP Configuration Guide (IBM Form SC31-8775)* dokumentiert.

Während der Initialisierung des TCP/IP-Stack verwendet die Konfigurationskomponente des TCP/IP-Stack TCPIP.DATA, um den HOSTNAME des Stack zu ermitteln. Zum Abrufen des Wertes wird die Suchreihenfolge für die z/OS-UNIX-Umgebung verwendet.

**Anmerkung:** Mit dem Trace-Auflöser können Sie bestimmen, welche TCPIP.DATA-Werte der Auflöser verwendet und woher sie stammen. Informationen zum dynamischen Starten des Trace enthält der *Communications Server IP Diagnosis Guide (IBM Form GC31-8782)*. Setzen Sie nach Aktivierung des Trace einen TSO-Befehl **NETSTAT HOME** und einen z/OS-UNIX-Shell-Befehl **netstat -h** ab, um die Werte anzuzeigen. Wenn Sie von TSO und von der z/OS-UNIX-Shell ein PING für einen Hostnamen absetzen, werden auch die Aktivitäten in Richtung aller DNS-Server angezeigt, die möglicherweise konfiguriert sind.

---

## Suchreihenfolgen in der z/OS-UNIX-Umgebung

Die Datei oder Tabelle, nach der gesucht wird, kann eine MVS-Dateigruppe oder eine HFS-Datei sein. Dies hängt von den Einstellungen in der Auflöserkonfiguration und dem Vorhandensein bestimmter Dateien im System ab.

### Basiskonfigurationsdateien des Auflösers

Die Basiskonfigurationsdatei des Auflösers enthält TCPIP.DATA-Anweisungen. Diese Datei wird wegen der enthaltenen Auflöseranweisungen referenziert, aber auch, weil sie unter anderem das Dateigruppenpräfix (Wert der Anweisung DATASETPREFIX) für den Zugriff auf die in diesem Abschnitt genannten Konfigurationsdateien enthält.



Für den Zugriff auf die Basiskonfigurationsdatei des Auflösers wird diese Suchreihenfolge verwendet:

1. **GLOBALTCPIPDATA**

Wenn der Wert der Konfigurationsanweisung GLOBALTCPIPDATA für den Auflöser definiert ist, wird er verwendet. (Lesen Sie hierzu auch den Abschnitt „Wissenswertes zu Auflösern“ auf Seite 95.) Es wird weiter nach einer zusätzlichen Konfigurationsdatei gesucht. Die Suche endet mit der nächsten gefundenen Datei.

2. Wert der Umgebungsvariablen **RESOLVER\_CONFIG**

Der Wert der Umgebungsvariablen wird verwendet. Die Suche scheitert, wenn die Datei nicht vorhanden ist oder anderweitig exklusiv zugeordnet ist.

3. **/etc/resolv.conf**

4. Karte **//SYSTCPD DD**

Die dem DD-Namen in SYSTCPD zugeordnete Dateigruppe wird verwendet. In der z/OS-UNIX-Umgebung hat ein untergeordneter Prozess keinen Zugriff auf die DD-Anweisung SYSTCPD. Dies ist darauf zurückzuführen, dass bei fork()- oder exec-Funktionsaufrufen die SYSTCPD-Zuordnung nicht vom übergeordneten Prozess übernommen wird.

5. **USERID.TCPIP.DATA**

USERID ist die Benutzer-ID, die der aktuellen Sicherheitsumgebung (Adressbereich oder Task/Thread) zugeordnet ist.

6. **JOBNAME.TCPIP.DATA**

JOBNAME ist der in der JCL-Anweisung JOB angegebene Name für Batch-Jobs oder der Prozedurname für eine gestartete Prozedur.

7. **SYS1.TCPPARMS(TCPDATA)**

8. **DEFAULTTCPIPDATA**

Wenn der Wert der Konfigurationsanweisung DEFAULTTCPIPDATA für den Auflöser definiert ist, wird er verwendet. (Lesen Sie hierzu auch den Abschnitt „Wissenswertes zu Auflösern“ auf Seite 95.)

9. **TCPIP.TCPIP.DATA**

## Umsetztabellen

Die Umsetztabellen (EBCDIC-ASCII und ASCII-EBCDIC) werden referenziert, um die zu verwendenden Umsetzungsdateigruppen zu ermitteln. Für den Zugriff auf diese Konfigurationsdatei wird die folgende Suchreihenfolge verwendet: (Die Suche endet mit der ersten gefundenen Datei.)

1. Der Wert der Umgebungsvariablen **X\_XLATE**

Dies ist der Name der mit dem TSO-Befehl CONVXLAT erzeugten Umsetztabelle.

2. **USERID.STANDARD.TCPXLBIN**

USERID ist die Benutzer-ID, die der aktuellen Sicherheitsumgebung (Adressbereich oder Task/Thread) zugeordnet ist.

3. **JOBNAME.STANDARD.TCPXLBIN**

JOBNAME ist der in der JCL-Anweisung JOB angegebene Name für Batch-Jobs oder der Prozedurname für eine gestartete Prozedur.

4. **HLQ.STANDARD.TCPXLBIN**

HLQ repräsentiert den Wert der Anweisung DATASETPREFIX in der Basiskonfigurationsdatei des Auflösers, der verwendet wird, wenn er angegeben ist. Andernfalls wird der Standard-HLQ TCPIP verwendet.

5. Wenn keine Tabelle gefunden wird, verwendet der Auflöser eine fest codierte Standardtabelle, die mit der im Dateigruppen-Member SEZATCPX(STANDARD) aufgelisteten Tabelle identisch ist.

## Lokale Hosttabellen

Standardmäßig versucht der Auflöser zuerst, konfigurierte Domännennamensserver für Auflösungsanforderungen zu verwenden. Falls die Auflösungsanforderung nicht erfüllt werden kann, werden lokale Hosttabellen genutzt. Das Verhalten des Auflösers wird von TCPIP.DATA-Anweisungen gesteuert.

Die TCPIP.DATA-Auflöseranweisungen definieren, ob und ggf. wie Domännennamensserver zu verwenden sind. Außerdem kann mit der Anweisung LOOKUP TCPIP.DATA gesteuert werden, wie Domännennamensserver und lokale Hosttabellen verwendet werden sollen. Weitere Informationen zu TCPIP.DATA-Anweisungen finden Sie in der Veröffentlichung *Communications Server IP Configuration Reference* (IBM Form SC31-8776).

Der Auflöser verwendet die spezifische Suchreihenfolge für Sitenamen von IPV4 uneingeschränkt für getnetbyname-API-Aufrufe. Spezifische Suchreihenfolge für Sitenamen von IPV4 (die Suche endet mit der ersten gefundenen Datei):

1. Der Wert der Umgebungsvariablen **X\_SITE**  
Dies ist der Name der mit dem TSO-Befehl **MAKESITE** erstellten Informationsdatei HOSTS.SITEINFO.
2. Wert der Umgebungsvariablen **X\_ADDR**  
Dies ist der Name der mit dem TSO-Befehl **MAKESITE** erstellten Informationsdatei HOSTS.ADDRINFO.
3. **/etc/hosts**
4. **USERID.HOSTS.SITEINFO**  
USERID ist die Benutzer-ID, die der aktuellen Sicherheitsumgebung (Adressbereich oder Task/Thread) zugeordnet ist.
5. **JOBNAME.HOSTS.SITEINFO**  
JOBNAME ist der in der JCL-Anweisung JOB angegebene Name für Batch-Jobs oder der Prozedurname für eine gestartete Prozedur.
6. **HLQ.HOSTS.SITEINFO**  
HLQ repräsentiert den Wert der Anweisung DATASETPREFIX in der Basiskonfigurationsdatei des Auflösers, der verwendet wird, wenn er angegeben ist. Andernfalls wird der Standard-HLQ TCPIP verwendet.

---

## Anwendung in Developer für System z

Wie bereits erwähnt, ist Developer für System z bei der Initialisierung davon abhängig, dass TCP/IP mit dem richtigen Hostnamen konfiguriert ist. Dies impliziert, dass die verschiedenen TCP/IP- und Resolver-Konfigurationsdateien ordnungsgemäß definiert sein müssen.

Im folgenden Beispiel geht es hauptsächlich um einige Konfigurations-Tasks für TCP/IP und den Auflöser (Resolver). Beachten Sie, dass es sich nicht um eine komplette Konfiguration für TCP/IP oder den Auflöser handelt. Das Beispiel hebt nur einige wichtige Aspekte hervor, die auf Ihren Standort anwendbar sein könnten.

1. In der folgenden JCL sehen Sie, dass TCP/IP den Stack-Hostnamen mit Hilfe von SYS1.TCPPARMS(TCPDATA) bestimmt.

```
//TCPIP    PROC  PARMS='CTRACE(CTIEZB00)',PROF=TCPPROF,DATA=TCPDATA
//*
//* TCP/IP-NETZ
//*
//TCPIP    EXEC  PGM=EZBTCPIP,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. Aus SYS1.TCPPARMS(TCPDATA) können wir entnehmen, dass der Systemname als Hostname verwendet werden soll und dass kein Domänen Namensserver (DNS) verwendet wird. Alle Namen werden durch eine Suche in der Standorttabelle aufgelöst.

```
; HOSTNAME gibt den TCP-Hostnamen dieses Systems an. Wenn kein
; Wert angegeben ist, wird für HOSTNAME standardmäßig der im PARMLIB-Member
; IEFSSNxx angegebene Knotenname verwendet.
;
; HOSTNAME
;
; DOMAINORIGIN gibt den Domänenursprung an, der an Hostnamen angehängt
; wird, die an den Auflöser übergeben werden. Enthält ein Hostname
; Punkte, wird der Wert von DOMAINORIGIN nicht an den Hostnamen
; angehängt.
;
DOMAINORIGIN  RALEIGH.IBM.COM
;
; NSINTERADDR gibt die IP-Adresse des Namensservers an.
; LOOPBACK (14.0.0.0) gibt Ihren lokalen Namensserver an. Wenn kein
; Namensserver verwendet wird, codieren Sie keine Anweisung NSINTERADDR.
; (Setzen Sie die folgende Zeile NSINTERADDR auf Kommentar, wenn alle
; Namen durch eine Suche in der Standorttabelle aufgelöst werden sollen.)
;
; NSINTERADDR  14.0.0.0
;
; TRACE RESOLVER bewirkt, dass ein vollständiger Trace für alle Abfragen an den
; Namensserver oder an Standorttabellen und alle entsprechenden Antworten auf die
; Benutzerkonsole geschrieben werden. Der Befehl ist nur für Debug-Zwecke bestimmt.
;
; TRACE RESOLVER
```

3. In der Resolver-JCL sehen wir, dass die DD-Anweisung SETUP nicht verwendet wird. Wie Sie aus dem Abschnitt „Wissenswertes zu Auflösern“ auf Seite 95 wissen, bedeutet dies, dass GLOBALTCPDATA und andere Variablen nicht verwendet werden.

```
//RESOLVER PROC  PARMS='CTRACE(CTIRES00)'
//*
//* RESOLVER FÜR IP-NAMEN – BEGINN MIT SUB=MSTR
//*
//RESOLVER EXEC  PGM=EZBREINI,REGION=0M,TIME=1440,PARM=&PARMS
//*SETUP DD  DISP=SHR,DSN=USER.PROCLIB(RESSETUP),FREE=CLOSE
```

4. Wenn wir davon ausgehen, dass die Umgebungsvariable RESOLVER\_CONFIG nicht gesetzt ist, können wir aus Tabelle 13 auf Seite 100 entnehmen, dass der Auflöser (Resolver) versuchen wird, /etc/resolv.conf als Basiskonfigurationsdatei zu verwenden.

```
TCPIPJOBNAME TCPIP
DomainOrigin RALEIGH.IBM.COM
HostName CDFMVS08
```

Wie im Abschnitt „Suchreihenfolgen in der z/OS-UNIX-Umgebung“ auf Seite 96 erwähnt, enthält die Basiskonfigurationsdatei TCPIP.DATA-Anweisungen.

Wenn der Systemname CDFMVS08 lautet, sehen wir, dass /etc/resolv.conf mit SYS1.TCPPARMS(TCPDATA) synchron ist. (TCPDATA gibt an, dass der Systemname als Hostname verwendet werden soll.) Es liegen keine DNS-Definitionen vor, so dass die Standorttabellen durchsucht werden.

5. Aus Tabelle 13 können wir außerdem entnehmen, dass in Ermangelung anderer Angaben standardmäßig die ASCII-EBCDIC-Umsetztabelle verwendet wird.
6. Unter der Voraussetzung, dass der TSO-Befehl **MAKESITE** nicht verwendet wird (um die Variablen X\_SITE und X\_ADDR zu erstellen), wird /etc/hosts als Standorttabelle für die Namenssuche verwendet.

```
# Resolver-/etc/hosts-Datei cdfmvs08
9.42.112.75 cdfmvs08 # CDFMVS08 Host
9.42.112.75 cdfmvs08.raleigh.ibm.com # CDFMVS08 Host
127.0.0.1 localhost
```

Der minimale Inhalt dieser Datei bezieht sich auf das aktuelle System. Im obigen Beispiel sind cdfmvs08 und cdfmvs08.raleigh.ibm.com als gültige Namen für die IP-Adresse des z/OS-Systems definiert.

Wenn ein Domänennamensserver (DNS) verwendet werden würde, würde der DNS die /etc/hosts-Informationen enthalten, und /etc/resolv.conf und SYS1.TCPPARMS(TCPDATA) würden Anweisungen enthalten, die den DNS für das System identifizieren.

Um Unklarheiten zu vermeiden, sollten die Konfigurationsdateien für TCP/IP und den Auflöser (Resolver) synchron sein.

*Tabelle 13. Für den Auflöser verfügbare lokale Definitionen*

Beschreibung des Dateityps	Betroffene APIs	Mögliche Dateien
Basis-konfigurationsdatei des Auflösers	Alle APIs	<ol style="list-style-type: none"> <li>1. GLOBALTCPIPDATA</li> <li>2. Umgebungsvariable RESOLVER_CONFIG</li> <li>3. /etc/resolv.conf</li> <li>4. DD-Name in SYSTCPD</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>
Umsetztabellen	Alle APIs	<ol style="list-style-type: none"> <li>1. Umgebungsvariable X_XLATE</li> <li>2. USERID.STANDARD.TCPXLBIN</li> <li>3. JOBNAME.STANDARD.TCPXLBIN</li> <li>4. HLQ.STANDARD.TCPXLBIN</li> <li>5. Vom Auflöser bereitgestellte Umsetztabelle (Member STANDARD in SEZATCPX)</li> </ol>

Tabelle 13. Für den Auflöser verfügbare lokale Definitionen (Forts.)

Beschreibung des Dateityps	Betroffene APIs	Mögliche Dateien
Lokale Hosttabellen	endhostent endnetent getaddrinfo gethostbyaddr gethostbyname gethostent GetHostNumber GetHostResol GetHostString getnameinfo getnetbyaddr getnetbyname getnetent IsLocalHost Resolve sethostent setnetent	IPV4 1. Umgebungsvariable X_SITE 2. Umgebungsvariable X_ADDR 3. /etc/hosts 4. USERID.HOSTS.xxxxINFO 5. JOBNAME.HOSTS.xxxxINFO 6. HLQ.HOSTS.xxxxINFO  IPV6 1. GLOBALIPNODES 2. Umgebungsvariable RESOLVER_IPNODES 3. USERID.ETC.IPNODES 4. JOBNAME.ETC.IPNODES 5. HLQ.ETC.IPNODES 6. DEFAULTIPNODES 7. /etc/ipnodes

**Anmerkung:** Die obige Tabelle ist ein Auszug aus einer Tabelle im *Communications Server IP Configuration Guide (IBM Form SC31-8775)*. Die vollständige Tabelle können Sie sich im genannten Handbuch ansehen.



---

## Anhang D. INETD konfigurieren

Dieser Anhang soll Sie bei einigen allgemeinen Problemen unterstützen, die beim Konfigurieren von INETD oder beim Überprüfen und/oder Modifizieren einer vorhandenen Konfiguration auftreten könnten.

Der Dämon INETD führt das Servicemanagement für ein IP-Netz durch. Er verringert die Systembelastung, indem er andere Dämonen nur bei Bedarf aufruft und intern mehrere einfache Internetservices (wie echo) bereitstellt. INETD liest die Konfigurationsdatei `inetd.conf`, um zu bestimmen, welche zusätzlichen Services bereitgestellt werden müssen. ETC.SERVICES wird für die Verknüpfung von Services mit Ports verwendet.

---

### inetd.conf

Die Services, die auf INETD zurückgreifen, sind in `inetd.conf` definiert. Diese Datei wird während der Startzeit von INETD gelesen. Die Standardposition und der Standardname für `inetd.conf` lauten `/etc/inetd.conf`. Ein Beispiel für eine Datei `inetd.conf` finden Sie unter `/samples/inetd.conf`.

Für Einträge in `inetd.conf` gelten die folgenden Syntaxregeln:

- Kommentare beginnen mit einem Nummernzeichen (#) oder Semikolon (;) und gehen bis zum Ende der Zeile.
- Bei Einträgen wird die Groß-/Kleinschreibung unterschieden.
- Die Einträge sind feldabhängig, jedoch nicht spaltenabhängig.
- Die Felder sind durch ein Leerzeichen oder Tabulatorzeichen voneinander getrennt.
- Einträge können mehrere Zeilen umfassen, wobei die folgenden zusätzlichen Syntaxregeln zu beachten sind:
  - Die Teilung muss zwischen zwei separaten Wörtern erfolgen (die durch ein Leerzeichen oder Tabulatorzeichen voneinander getrennt sind).
  - Die Folgezeile muss mit einem Leerzeichen oder Tabulatorzeichen beginnen.
  - In die Fortsetzung dürfen keine Kommentare eingebettet werden.

Jeder Eintrag umfasst sieben positionsgebundene Felder im folgenden Format:

```
Servicename Socket-Typ Protokoll Option_wait Benutzer-ID Serverprogramm  
Serverprogrammargumente
```

#### [IP-Adresse:]Servicename

IP-Adresse ist eine lokale IP-Adresse, gefolgt von einem Doppelpunkt. Wenn die Adresse angegeben ist, wird sie an Stelle von `INADDR_ANY` oder der aktuellen Standardadresse verwendet. Wenn Sie speziell `INADDR_ANY` anfordern möchten, verwenden Sie `"*."`. Wenn IP-Adresse (oder ein Doppelpunkt) ohne weitere Einträge in der Zeile angegeben ist, wird diese Adresse zum Standard für die folgenden Zeilen, bis eine neue Standardadresse angegeben ist. Servicename ist ein anerkannter oder benutzerdefinierter Servicename. Der angegebene Name muss mit einem der in ETC.SERVICES definierten Servernamen übereinstimmen.

#### Socket-Typ

Der Typ `stream` oder `dgram` gibt an, ob für den Service ein Datenstrom- oder Datagramm-Socket verwendet wird.

### **Protokoll[,sndbuf=n[,rcvbuf=n]]**

Für Protokoll kann der Wert tcp[4|6] oder udp[4|6] als weitere Qualifizierung des Servicenamens angegeben werden. Der Servicenamen und das Protokoll müssen mit einem Eintrag in ETC.SERVICES übereinstimmen, bis auf die "4" oder "6", die im Eintrag in ETC.SERVICES nicht enthalten sein sollte.

Die Werte sndbuf und rcvbuf geben die Größe des Sendepuffers und des Empfangspuffers an. Die von n repräsentierte Größe kann in Bytes angegeben werden. Sie können aber auch ein "k" oder "m" hinzufügen, wenn Sie Kilobytes oder Megabytes angeben möchten. Die Reihenfolge, in der Sie sndbuf und rcvbuf angeben, ist beliebig.

### **Option\_wait[.max]**

Mit der Option wait oder nowait.wait wird angegeben, dass der Dämon ein Einzel-Thread-Dämon ist und die nächste Anforderung erst bedienen kann, wenn die erste abgeschlossen ist. Wenn nowait angegeben ist, setzt INETD bei Empfang einer Verbindungsanforderung an einem Datenstrom-Socket ein 'accept' ab. Wenn wait angegeben ist, muss der Server das 'accept' absetzen, sofern es sich um einen Datenstrom-Socket handelt.

Der Wert max gibt die maximal zulässige Anzahl von Benutzern an, die innerhalb eines Intervalls von 60 Sekunden einen Service anfordern dürfen. Der Standardwert liegt bei 40. Wird der Maximalwert überschritten, wird der Service-Port geschlossen.

### **Benutzer-ID[.Gruppe]**

Benutzer-ID ist die Benutzer-ID, unter der der verzweigte Dämon ausgeführt werden muss. Diese Benutzer-ID kann von der INETD-Benutzer-ID verschieden sein. Welche Berechtigungen dieser Benutzer-ID erteilt werden, hängt von den Anforderungen des Service ab. Die INETD-Benutzer-ID benötigt die Berechtigung BPX.DAEMON, um den verzweigten Prozess auf diese Benutzer-ID umzuschalten.

Der optionale Wert Gruppe ist durch einen Punkt (.) von der Benutzer-ID getrennt und ermöglicht die Ausführung des Servers mit einer Gruppen-ID, die von der Standardgruppen-ID für diese Benutzer-ID abweicht.

### **Serverprogramm**

Serverprogramm ist der vollständige Pfadname des Service. Beispiel:  
/usr/sbin/rlogind ist der vollständige Pfadname für den Befehl rlogind.

### **Serverprogrammargumente**

Es können maximal 20 Argumente angegeben werden. Das erste Argument ist der Servername.

---

## **ETC.SERVICES**

INETD verwendet ETC.SERVICES, um den Services, die von INETD unterstützt werden müssen, Port-Nummern und Protokolle zuzuordnen. Diese Datei kann eine MVS-Dateigruppe oder eine z/OS-UNIX-Datei sein. Ein Beispiel ist in SEZAINST(SERVICES) enthalten, das auch unter /usr/lpp/tcpip/samples/services verfügbar ist. Die Suchreihenfolge für ETC.SERVICES hängt davon ab, ob INETD mit einer z/OS-UNIX-Methode oder einer nativen MVS-Methode gestartet wird.

Für die Spezifikation der Serviceinformationen gelten die folgenden Syntaxregeln:



- Wenn ETC.SERVICES eine MVS-Dateigruppe ist, muss es sich um eine Datei im Fixed-Format oder Fixed-Block-Format mit LRECL zwischen 56 und 256 handeln.
- Wenn ETC.SERVICES eine HFS-Datei ist, kann sie eine maximale Länge von 256 haben.
- Einträge in einer Zeile sind durch Leerzeichen oder Tabulatorzeichen voneinander getrennt.
- Jeder Service ist in einer gesonderten Zeile angegeben.
- Jeder Servicename muss an der ersten Position einer Zeile beginnen.
- Die maximale Länge für Servicenamen und Aliasnamen liegt bei 32 Zeichen.
- Es werden maximal 35 Aliasnamen erkannt.
- Bei Servicenamen und Aliasnamen wird die Groß-/Kleinschreibung unterschieden.
- Kommentare beginnen mit einem Nummernzeichen (#) oder Semikolon (;) und gehen bis zum Ende der Zeile.

Jeder Eintrag umfasst vier positionsgebundene Felder im folgenden Format:

Servicename    Port-Nummer/Protokoll    Aliasnamen

#### **Servicename**

Gibt einen anerkannten oder benutzerdefinierten Servicenamen an.

#### **Port-Nummer**

Gibt die für den Service verwendete Socket-Port-Nummer an.

#### **Protokoll**

Gibt das für den Service verwendete Transportprotokoll an. Gültige Werte sind tcp und udp.

#### **Aliasnamen**

Gibt eine Liste nicht offizieller Servicenamen an.

## **Suchreihenfolge in der z/OS-UNIX-Umgebung**

Für den Zugriff auf ETC.SERVICES unter z/OS UNIX wird diese Suchreihenfolge verwendet. Die Suche endet mit der ersten gefundenen Datei:

1. **/etc/services**
2. **USERID.ETC.SERVICES**  
USERID ist die Benutzer-ID, die zum Starten von INETD verwendet wird..
3. **HLQ.ETC.SERVICES**  
HLQ repräsentiert den Wert der Anweisung DATASETPREFIX in der Basiskonfigurationsdatei des Auflösers, der verwendet wird, wenn er angegeben ist. Andernfalls wird der Standard-HLQ TCPIP verwendet.

## **Suchreihenfolge in der nativen MVS-Umgebung**

Für den Zugriff auf ETC.SERVICES in der nativen MVS-Umgebung wird diese Suchreihenfolge verwendet. Die Suche endet mit der ersten gefundenen Dateigruppe:

1. **//DD-Karte SERVICES**  
Die der DD-Anweisung SERVICES zugeordnete Dateigruppe wird verwendet.
2. **USERID.ETC.SERVICES**  
USERID ist die Benutzer-ID, die zum Starten von INETD verwendet wird..
3. **JOBNAME.ETC.SERVICES**

JOBNAME ist der in der JCL-Anweisung JOB angegebene Name für Batch-Jobs oder der Prozedurname für eine gestartete Prozedur.

#### 4. HLQ.ETC.SERVICES

HLQ repräsentiert den Wert der Anweisung DATASETPREFIX in der Basiskonfigurationsdatei des Auflösers, der verwendet wird, wenn er angegeben ist. Andernfalls wird der Standard-HLQ TCPIP verwendet.

**Anmerkung:** Wenn INETD mit BPXPATCH gestartet wird, wird nicht die native MVS-Suchreihenfolge verwendet, weil BPXBATCH den Startbefehl in der z/OS-UNIX-Umgebung ausführt. Die native MVS-Suchreihenfolge wird nur verwendet, wenn ein MVS-Lademodul wie SEZALOAD(FTP) gestartet wird.

---

## Port-Definitionen in PROFILE.TCPIP

Verwechseln Sie nicht die PORT-Definitionen (oder PORTRANGE-Definitionen) in PROFILE.TCPIP mit Ports, die in ETC.SERVICES definiert sind. Diese Ports werden jeweils für verschiedene Zwecke verwendet. Anhand der in PROFILE.TCPIP definierten Ports stellt TCPIP fest, ob ein Port für einen bestimmten Service reserviert ist. ETC.SERVICES wird von INETD verwendet, um einem Service einen Port zuzuordnen.

Wenn INETD eine Anforderung an einem überwachten Port empfängt, richtet er eine untergeordnete Prozessverzweigung (mit dem angeforderten Service) ein, die die Bezeichnung inetdx hat. Hier steht inetd für den Jobnamen für INETD (der von der Startmethode abhängig ist) und x für eine einstellige Zahl.

Dies kompliziert die Port-Reservierung. Wenn ein überwachter INETD-Port in PROFILE.TCPIP reserviert ist, sollte der Name der gestarteten JCL-Prozedur für den z/OS-UNIX-Kernel-Adressbereich verwendet werden, damit fast jeder Prozess an den Port gebunden werden kann. Dieser Name ist normalerweise OMVS, sofern im Parameter STARTUP\_PROC des PARMLIB-Members BPXPRMxx nicht explizit ein anderer Name angegeben ist.

In der nachfolgenden Auflistung ist erläutert, wie der Jobname ausgehend von der Umgebung, in der die Anwendung ausgeführt wird, bestimmt werden kann:

- Im Stapelbetrieb ausgeführte Anwendungen verwenden den Batch-Jobnamen.
- Von der MVS-Bedienerkonsole aus gestartete Anwendungen verwenden den Namen der gestarteten Prozedur (STC) als Jobnamen.
- Mit einer TSO-Benutzer-ID ausgeführte Anwendungen verwenden die TSO-Benutzer-ID als Jobnamen.
- Anwendungen, die von der z/OS-Shell ausgeführt werden, haben normalerweise einen Jobnamen, der sich aus der Benutzer-ID des angemeldeten Benutzers und einem Suffix (ein Zeichen) zusammensetzt.
- Berechtigte Benutzer können Anwendungen von der z/OS-Shell ausführen und den Jobnamen mit der Umgebungsvariablen \_BPX\_JOBNAME setzen. In diesem Fall ist der für die Umgebungsvariable angegebene Wert der Jobname.
- Der Name der gestarteten JCL-Prozedur für den USS-Kernel-Adressbereich kann verwendet werden, wenn die Anbindung an den Port für fast alle Aufrufer der Socket-API bind() (mit Ausnahme von Benutzern der Pascal-API) möglich sein soll. Dieser Name ist normalerweise OMVS, sofern im Parameter STARTUP\_PROC des PARMLIB-Members BPXPRMxx nicht explizit ein anderer Name angegeben ist.
- Von INETD gestartete z/OS-UNIX-Anwendungen verwenden den Jobnamen des INETD-Servers.

**Anmerkung:** Die in ETC.SERVICES definierten Ports können sich von der in PROFILE.TCPIP für den Service reservierten Port-Nummer unterscheiden, obwohl dies nicht empfohlen wird.

---

## /etc/inetd.pid

Der INETD-Prozess erstellt eine temporäre Datei /etc/inetd.pid, die die Prozess-ID (PID) des derzeit ausgeführten Dämons INETD enthält. Mit Hilfe dieses PID-Wertes werden syslog-Einträge identifiziert, die von dem INETD-Prozess stammen. Dieser PID-Wert wird außerdem für Befehle bereitgestellt, die einen solchen Wert benötigen, z. B. kill. Darüber hinaus wird die PID als Sperrmechanismus verwendet, um zu verhindern, dass mehr als ein INETD-Prozess aktiv ist.

---

## Start

Die z/OS-UNIX-Implementierung von INETD befindet sich standardmäßig in /usr/sbin/inetd und unterstützt zwei optionale, nicht positionsgebundene Startparameter:

```
/usr/sbin/inetd [-d] [inetd.conf]
```

**-d** Debug-Option. Die Debug-Ausgabe wird an stderr gesendet und kann dann vom Dämon syslogd an eine Datei weitergeleitet werden. Weitere Informationen zu syslogd finden Sie im *Communications Server IP Configuration Guide (IBM Form SC31-8775)*. Wenn INETD auf diese Weise gestartet wurde, wird keine Verzweigung für einen untergeordneten Prozess zum Starten eines Service erstellt.

### inetd.conf

Konfigurationsdatei. Der Standardwert ist /etc/inetd.conf

Es ist ratsam, INETD während des IPL zu starten. Am häufigsten geschieht dies von /etc/rc oder /etc/inittab aus (nur unter z/OS ab Version 1.8). Sie können den Dämon auch von einem Job oder einer gestarteten Task aus starten, indem Sie BPXBATCH verwenden, oder in einer Shell-Sitzung eines Benutzers mit entsprechender Berechtigung.

## /etc/rc

Wenn INETD unter z/OS UNIX vom Initialisierungs-Shell-Skript /etc/rc gestartet wird, sucht der Dämon in der z/OS-UNIX-Suchreihenfolge nach ETC.SERVICES. Eine Beispieldatei /etc/rc ist im Lieferumfang als /samples/rc enthalten. Zum Starten von INETD können die folgenden Beispielbefehle verwendet werden.

```
# Start INETD
_BPX_JOBNAME='INETD' /usr/sbin/inetd /etc/inetd.conf &
sleep 5
```

## /etc/inittab

Unter z/OS ab Version 1.8 gibt es eine alternative Methode (/etc/inittab), um während der Initialisierung von z/OS UNIX Befehle abzusetzen. Bei Verwendung von /etc/inittab haben Sie die Möglichkeit, den Parameter respawn zu definieren, der den Prozess automatisch neu startet, wenn er beendet ist. (Für einen zweiten Neustart innerhalb von 15 Minuten wird ein WTOR an den Bediener gesendet.) Wenn INETD unter Verwendung von /etc/inittab gestartet wird, sucht der Dämon in der z/OS-UNIX-Suchreihenfolge nach ETC.SERVICES. Eine Beispieldatei /etc/inittab ist im Lieferumfang als /samples/inittab enthalten. Zum Starten von INETD kann der folgende Beispielbefehl verwendet werden.

```
# Start INETD
inetd::respfrk:/usr/sbin/inetd /etc/inetd.conf
```

**Anmerkung:** Beachten Sie, dass der im Beispiel verwendete Parameter respfrk das Attribut respawn an alle verzweigten Prozesse, einschließlich RSE, sendet. Wenn der Client die Verbindung schließt, wird sie von respawn neu gestartet. Der RSE-Server wird später wieder durch eine Zeitlimitüberschreitung beendet. Wenn Sie mehr über inittab erfahren möchten, lesen Sie die Veröffentlichung *UNIX System Services Planning* (IBM Form GA22-7800).

## BPXBATCH

Die Startmethode BPXBATCH funktioniert für STCs und für Benutzerjobs. Vergessen Sie nicht, dass INETD ein Hintergrundprozess ist, so dass der BPXBATCH-Schritt, der INETD startet, innerhalb weniger Sekunden nach dem Start beendet ist. Wenn INETD von BPXBATCH gestartet wird, sucht der Dämon in der z/OS-UNIX-Suchreihenfolge nach ETC.SERVICES. Die in Abb. 11 aufgelistete JCL ist eine Beispielprozedur zum Starten von INETD. (Der Schritt KILL entfernt einen ggf. vorhandenen aktiven INETD-Prozess.)

```
//INETD    PROC PRM=
//*
//KILL      EXEC PGM=BPXBATCH,
//          PARM='SH ps -e | grep inetd | cut -c 1-10 | xargs -n 1 kill'
//*
//INETD     EXEC PGM=BPXBATCH,TIME=NOLIMIT,REGION=0M,
//          PARM='PGM /usr/sbin/inetd &PRM'
//STDERR    DD PATH='/tmp/bpxbatch.start.inetd.stderr',
//          PATHOPTS=(OWRONLY,OCREAT,OTRUNC),
//          PATHMODE=SIRWXU
//* STDIN, STDOUT und STDENV nehmen standardmäßig den Wert /dev/null an.
//* Die Ausgabe des Dämons INETD kann durch syslogd-Einstellungen gesteuert werden.
//*
```

Abbildung 11. Start-JCL für INETD

**Anmerkung:** STDIN, STDOUT und STDERR müssen beim Anlegen z/OS-UNIX-Dateien sein. STDENV kann eine MVS-Dateigruppe oder eine z/OS-UNIX-Datei sein. Wenn Sie mehr über BPXBATCH erfahren möchten, lesen Sie die Veröffentlichung *UNIX System Services Command Reference* (IBM Form SA22-7802).

**Anmerkung:** Wenn INETD von BPXBATCH gestartet wird, kann inetd.conf eine MVS-Dateigruppe oder ein MVS-Dateigruppen-Member sein. Codieren Sie hierfür die Anweisung PARM wie im folgenden Beispiel. (Verwenden Sie nur einfache Hochkommata (').)

```
// PARM='PGM /usr/sbin/inetd //'SYS1.TCPPARMS(INETCONF)'' &PRM'
```

## Shell-Sitzung

Wenn INETD von einer Shell-Sitzung aus gestartet wird, sucht der Dämon in der z/OS-UNIX-Suchreihenfolge nach ETC.SERVICES. Die folgenden Beispielbefehle können (von einer Person mit ausreichender Berechtigung) zum Stoppen und Starten von INETD verwendet werden. (# ist die z/OS-UNIX-Eingabeaufforderung.)

```
1. # ps -e | grep inetd
   7 ?          0:00 /usr/sbin/inetd
2. # kill 7
3. # _BPX_JOBNAME='INETD' /usr/sbin/inetd
```

**Anmerkung:** Diese Methode wird nicht für den Erststart empfohlen. Dafür ist /etc/rc oder /etc/inittab besser geeignet, da die Scripts während der Initialisierung von z/OS UNIX ausgeführt werden.

---

## Sicherheit

INETD ist ein z/OS-UNIX-Prozess und erfordert daher, dass die Sicherheitssoftware gültige OMVS-Definitionen für die INETD zugeordnete Benutzer-ID enthält. Für die Benutzer-ID müssen UID, HOME und PROGRAM gesetzt sein. Außerdem muss GID für die Standardgruppe des Benutzers gesetzt sein. Wenn INETD von /etc/rc oder /etc/inittab gestartet wird, wird die Benutzer-ID vom z/OS-UNIX-Kernel übernommen. Standardmäßig lautet sie OMVSKERN.

```
ADDGROUP OMVSGRP OMVS(GID(1))
ADDUSER OMVSKERN DFLTGRP(OMVSGRP) NOPASSWORD +
      OMVS(UID(0) HOME('/') PROGRAM('/bin/sh'))
```

INETD ist ein Dämon, der Zugriff auf Funktionen wie setuid() haben muss. Die Benutzer-ID, mit der INETD gestartet wird, benötigt daher für das Profil BPX.DAEMON in der Klasse FACILITY die Zugriffsberechtigung READ. Wenn das Profil nicht definiert ist, muss zwingend UID 0 verwendet werden.

```
PERMIT BPX.DAEMON CLASS(FACILITY) ACCESS(READ) ID(OMVSKERN)
```

Die INETD-Benutzer-ID benötigt außerdem die Zugriffsberechtigung EXECUTE für das Programm inetd (/usr/sbin/inetd), die Berechtigung READ für den Zugriff auf die Dateien inetd.conf und ETC.SERVICES sowie die Zugriffsberechtigung WRITE für /etc/inetd.pid. Wenn Sie INETD ohne UID 0 ausführen möchten, können Sie für das Profil SUPERUSER.FILESYS in der Klasse UNIXPRIV die Zugriffsberechtigung CONTROL einrichten, damit die erforderlichen Rechte für z/OS-UNIX-Dateien vorhanden sind.

Programme, die eine Dämonberechtigung benötigen, müssen programmgesteuert sein, wenn BPX.DAEMON in der Klasse FACILITY definiert ist. Bei Verwendung des INETD-Standardprogramms (/usr/sbin/inetd) ist dies bereits der Fall. Für Kopien oder eine angepasste Version müssen Sie die Programmsteuerung jedoch manuell definieren. Mit dem Befehl **extattr +p** können Sie eine z/OS-UNIX-Datei zu einer programmgesteuerten Datei machen. Mit der RACF-Klasse PROGRAM können Sie aus einem MVS-Lademodul ein programmgesteuertes Modul machen.

Systemprogrammierer, die INETD von ihrer Shell-Sitzung aus neu starten müssen, verwenden ihre Berechtigungen für den Start von INETD. Deshalb müssen sie dieselben Rechte wie die reguläre INETD-Benutzer-ID haben. Darüber hinaus benötigen sie die Berechtigung, den INETD-Prozess aufzulisten und zu stoppen. Diese Berechtigung kann auf verschiedenen Wegen erteilt werden.

- UID 0  
Für Benutzer-IDs von Personen wird dies nicht empfohlen, weil es keine z/OS-UNIX-bezogenen Einschränkungen gibt.
- Zugriffsberechtigung READ für das Profil BPX.SUPERUSER in der Klasse FACILITY  
Über den Befehl **su** kann der Benutzer ein Benutzer mit der UID 0 werden. Diese Konfiguration wird empfohlen.
- Zugriff auf Einzelprofile, die die erforderlichen Berechtigungen abdecken
  - Zugriffsberechtigung READ für SUPERUSER.PROCESS.GETPSENT in der Klasse UNIXPRIV (für den Befehl **ps**)
  - Zugriffsberechtigung READ für SUPERUSER.PROCESS.KILL in der Klasse UNIXPRIV (für den Befehl **kill**)

- Zugriffsberechtigung READ für BPX.JOBNAME in der Klasse FACILITY (für die Umgebungsvariable \_BPX\_JOBNAME)

Wenn Sie mehr über die Befehle **extattr** und **su** erfahren möchten, lesen Sie die Veröffentlichung *UNIX System Services Command Reference (IBM Form SA22-7802)*. Weitere Informationen zur Klasse UNIXPRIV und zu BPX.\*-Profilen in der Klasse FACILITY finden Sie in der Veröffentlichung *UNIX System Services Planning (IBM Form GA22-7800)*. Weitere Informationen zu den OMVS-Segmentdefinitionen und zur Klasse PROGRAM enthält der *Security Server RACF Security Administrator's Guide (IBM Form SA22-7683)*.

---

## Anforderungen von Developer für System z

Developer für System z ist bei der Einrichtung der Verbindung zwischen Client und Host auf INETD angewiesen. Das Produkt stellt zusätzliche Anforderungen, die über die oben beschriebenen Anforderungen an die INETD-Konfiguration hinausgehen.

### INETD

Die Umgebungseinstellungen von INETD werden übergeben, wenn ein Prozess gestartet wird. Die Berechtigungen für die INETD-Benutzer-ID müssen ordnungsgemäß gesetzt sein, damit INETD den RSE-Server starten kann.

- Wenn INETD von JCL mit BPXBATCH gestartet wird, muss die Regionsgröße gleich null sein.
- Wird INETD in einer TSO/OMVS-Shell-Sitzung gestartet, muss die Regionsgröße bei mindestens 2096128 liegen.
- Wenn INETD von /etc/rc oder /etc/inittab gestartet wird, wird die Regionsgröße von SYS1.PROCLIB(BPX0INIT) verwendet, die standardmäßig bei 0 liegt.
- Zugriffsrechte READ und EXECUTE für das Installationsverzeichnis von Developer für System z (standardmäßig /usr/lpp/wd4z/\*)

### RSE-Dämon

Der RSE-Dämon, der von INETD gestartet wird, wenn ein Client eine Verbindung zum Port 4035 herstellt, führt die Authentifizierung durch, startet den RSE-Server und gibt die Port-Nummer für die weitere Kommunikation an den Client zurück. Die dem RSE-Dämon zugeordnete Benutzer-ID benötigt hierfür die folgenden Berechtigungen:

- Gültige OMVS-Definitionen in der Sicherheitssoftware; UID, HOME und PROGRAM müssen gesetzt sein. Außerdem muss GID für die Standardgruppe des Benutzers gesetzt sein.
- Zugriffsberechtigung READ für das Profil BPX.DAEMON in der Klasse FACILITY
- Zugriffsrechte READ und EXECUTE für das Installationsverzeichnis von Developer für System z (standardmäßig /usr/lpp/wd4z/\*)
- Zugriffsrechte READ und EXECUTE für das Konfigurationsverzeichnis von Developer für System z. Standardmäßig ist dies /usr/lpp/wd4z/rse/lib/; Sie sollten jedoch ein anderes Verzeichnis verwenden, z. B. /etc/wd4z/.



---

## Anhang E. SSL konfigurieren

Dieser Anhang soll Sie bei einigen allgemeinen Problemen unterstützen, die beim Konfigurieren von SSL (Secure Sockets Layer) oder beim Überprüfen und/oder Modifizieren einer vorhandenen Konfiguration auftreten könnten.

Sichere Kommunikation bedeutet, dass Ihr DFV-Partner derjenige ist, der er zu sein vorgibt, und dass Informationen in einer Weise übertragen werden, die es anderen erschwert, die Daten abzufangen und zu lesen. Secure Sockets Layer (SSL) bietet diese Fähigkeiten für ein TCP/IP-Netz an. SSL verwendet digitale Zertifikate für Ihre Identifikation und ein Protokoll mit öffentlichen Schlüsseln, um die Kommunikation zu verschlüsseln. Weitere Informationen zu digitalen Zertifikaten und zu dem von SSL verwendeten Protokoll mit öffentlichen Schlüsseln enthält der *Security Server RACF Security Administrator's Guide (IBM Form SA22-7683)*.

Welche Aktionen erforderlich sind, um die SSL-Kommunikation für Developer für System z zu konfigurieren, hängt von den genauen Anforderungen am jeweiligen Standort, vom verwendeten RSE-Kommunikationsverfahren und von den am Standort verfügbaren Ressourcen ab.

In diesem Anhang werden Sie die aktuellen RSE-Definitionen klonen, damit Sie eine zweite Verbindung haben, die SSL verwendet. Sie werden sowohl die REXEC-Methode als auch die Methode mit der Dämonverbindung für SSL konfigurieren. Außerdem werden Sie Ihre eigenen Sicherheitszertifikate erstellen, die von den verschiedenen Teilnehmern der RSE-Verbindung verwendet werden. Für diese gesamte Prozedur werden Sie die folgenden Schritte ausführen:

1. Vorhandene RSE-Konfiguration klonen
2. Bestimmen, welche Dateien zu verwenden sind
3. Keystore mit keytool erstellen
4. Schlüsseldatenbank mit RACF oder gskkyman erstellen (nur für den Dämon)
5. Datei ssl.properties aktualisieren, um SSL zu aktivieren
6. Verbindung testen

**Anmerkung:** Weitere Informationen zur Verwendung eines von einer anerkannten Zertifizierungsstelle signierten Zertifikats bzw. zum Konfigurieren einer eigenen Zertifizierungsstelle enthält das White Paper *Setting up SSL for RSE Communication (IBM Form SC23-5816)* in der Internetbibliothek zu Developer für System z unter <http://www-306.ibm.com/software/awdtools/devzseries/library/>.

In diesem Anhang wird die folgende einheitliche Namenskonvention verwendet:

- Zertifikat: wd4zrse
- Schlüssel- und Zertifikatspeicher: wd4zssl.\*
- Kennwort: rssl

Für die meisten nachfolgenden Tasks wird vorausgesetzt, dass Sie aktivierter zOS-UNIX-Benutzer sind. Zum Aktivieren können Sie den TSO-Befehl **OMVS** absetzen. Mit dem Befehl **exit** können Sie zu TSO zurückkehren.

---

## Vorhandene RSE-Konfiguration klonen

In diesem Schritt wird eine neue Instanz des RSE-Servers und des RSE-Dämons erstellt. Die neuen Instanzen werden parallel zu den vorhandenen Instanzen ausgeführt. So ist sichergestellt, dass die SSL-Tests nicht den normalen Betrieb behindern. Bei den folgenden Beispielbefehlen wird vorausgesetzt, dass sich die Konfigurationsdateien in /etc/wd4z/ befinden. Lesen Sie hierzu die Empfehlung im Abschnitt „Konfigurationsdatei rsed.envvars in einem anderen Verzeichnis speichern“ auf Seite 34.

```
$ cd /etc/wd4z
$ mkdir ssl
$ cp * ssl
cp: FSUM6254 "ssl" is a directory (not copied)
$ ls ssl
rsecomm.properties  server.zseries      ssl.properties
rsed.envvars         setup.env.zseries
$ cd ssl
$ su
# oedit /etc/services

rsessl          4047/tcp          #Developer für System z RSE mit SSL
```

Hinzufügen des Service, der den Port 4047 verwendet

```
# oedit /etc/inetd.conf
```

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

Hinzufügen des rsessl-Dämons, der das Konfigurationsverzeichnis /etc/wd4z/ssl verwendet

```
# ps -e | grep inetd
7 ?          0:00 /usr/sbin/inetd
# kill 7
# _BPX_JOBNAME='INETD' /usr/sbin/inetd
# exit
$ netstat | grep 4047
INETD4  00016619 0.0.0.0..4047          0.0.0.0..0          Listen
```

Die oben aufgeführten Befehle erstellen ein Unterverzeichnis mit dem Namen ssl und füllen es mit den obligatorischen Konfigurationsdateien. Es müssen (noch) keine Konfigurationsänderungen vorgenommen werden. Das Installationsverzeichnis und die MVS-Komponenten können gemeinsam genutzt werden, da sie nicht SSL-spezifisch sind. Sie müssen jedoch einen neuen Dämon (rsessl) erstellen, der die neuen Konfigurationsdateien verwendet. Dem neuen Dämon wird der Port 4047 zugeordnet.

Weitere Informationen zu den oben gezeigten Aktionen enthält Kapitel 4, „z/OS-UNIX-Komponenten von Developer für System z aktivieren“, auf Seite 33.

---

## Bestimmen, welche Dateien zu verwenden sind

Die von SSL verwendeten Identitätszertifikate und Schlüssel für die Verschlüsselung/Entschlüsselung werden in einer Schlüsseldatei gespeichert. Die jeweiligen Implementierungen dieser Schlüsseldatei sind vom Anwendungstyp abhängig.

Der RSE-Dämon ist eine System-SSL-Anwendung und verwendet eine Schlüsseldatenbankdatei. Diese Schlüsseldatenbank kann eine von gskkyman erstellte physi-



sche Datei oder eine von Ihrer Sicherheitssoftware (z. B. RACF) verwaltete Schlüsseldatei sein. Der (vom Dämon oder von REXEC gestartete) RSE-Server ist eine Java-SSL-Anwendung und verwendet eine von keytool erstellte Keystore-Datei. RACF bietet derzeit keine vordefinierte Unterstützung für Java SSL.

Für die Verbindung über REXEC benötigen Sie somit nur die Keystore-Datei:

- Keystore (keytool)

Für die Verbindung über den Dämon benötigen Sie den Keystore und die Schlüsseldatenbankdatei:

- Keystore (keytool)
- Schlüsseldatenbank (RACF oder gskkyman)

Weitere Informationen zu RACF und digitalen Zertifikaten enthält der *Security Server RACF Security Administrator's Guide (IBM Form SA22-7683)*. Die Dokumentation zu gskkyman ist in der Veröffentlichung *Cryptographic Services System SSL Programming (IBM Form SC24-5901)* enthalten. Die Dokumentation zu keytool finden Sie unter <http://java.sun.com/j2se/1.4.2/docs/tooldocs/solaris/keytool.html>.

---

## Keystore mit keytool erstellen

Der Befehl "keytool -genkey" generiert ein Schlüsselpaar (einen öffentlichen Schlüssel und einen zugehörigen privaten Schlüssel). Anschließend wird der öffentliche Schlüssel in ein selbst signiertes Zertifikat (X.509 V1) eingeschlossen, das als Zertifikatkette mit einem Element gespeichert wird. Diese Zertifikatkette und der private Schlüssel werden als ein (mit einem Aliasnamen bezeichneten) Eintrag in einer (neuen) Keystore-Datei gespeichert.

**Anmerkung:** Sie müssen Java in Ihre Suchverzeichnisse für Befehle aufnehmen. Für die Ausführung von keytool ist möglicherweise die folgende Anweisung notwendig (/usr/lpp/java/J1.4 steht hier für das Verzeichnis, in dem Java installiert ist): PATH=\$PATH:/usr/lpp/java/J1.4/bin

Alle Informationen können als ein Parameter übergeben werden. Durch die Längenbeschränkung der Befehlszeile sind jedoch einige Interaktionen erforderlich.

```
$ keytool -genkey -alias wd4zrse -validity 3650 -keystore wd4zssl.jks -storepass
rsessl -keypass rsessl
What is your first and last name?
[Unknown]: wd4z rse ssl
What is the name of your organizational unit?
[Unknown]: wd4z
What is the name of your organization?
[Unknown]: IBM
What is the name of your City or Locality?
[Unknown]: Raleigh
What is the name of your State or Province?
[Unknown]: NC
What is the two-letter country code for this unit?
[Unknown]: US
Is CN=wd4z rse ssl, OU=wd4z, O=IBM, L=Raleigh, ST=NC, C=US correct? (type "yes"
or "no")
[no]: yes
$ ls
rsecomm.properties      server.zseries          ssl.properties
rsed.envvars            setup.env.zseries       wd4zssl.jks
```

Das oben erstellte selbst signierte Zertifikat ist für ca. 10 Jahre gültig (ohne Berücksichtigung des zusätzlichen Tages in Schaltjahren). Es wird in /etc/wd4z/ssl/wd4zssl.jks mit dem Aliasnamen wd4zrse gespeichert. Das Kennwort (rsessl) stimmt mit dem Keystore-Kennwort überein. Dies ist eine RSE-Anforderung.

Das Ergebnis können Sie mit der Option -list überprüfen:

```
$ keytool -list -alias wd4zrse -keystore wd4zssl.jks -storepass rsessl -v
Alias name: wd4zrse
Creation date: May 24, 2007
Entry type: keyEntry
Certificate chain length: 1
Certificate 1:
Owner: CN=wd4z rse ssl, OU=wd4z, O=IBM, L=Raleigh, ST=NC, C=US
Issuer: CN=wd4z rse ssl, OU=wd4z, O=IBM, L=Raleigh, ST=NC, C=US
Serial number: 46562b2b
Valid from: 5/24/07 2:17 PM until: 5/21/17 2:17 PM
Certificate fingerprints:
    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
```

---

## Schlüsseldatenbank erstellen (nur für den Dämon)

Der Dämon ist, wie bereits erwähnt, eine System-SSL-Anwendung, die eine Schlüsseldatenbankdatei verwendet. Diese Datei kann eine von gskkyman erstellte physische Datei oder eine RACF-Schlüsseldatei sein. Für die Verwaltung privater Schlüssel und Zertifikate für System SSL sind RACF-Schlüsseldateien die bevorzugte Methode.

**Anmerkung:** System SSL verwendet ICSF (Cryptographic Service Facility), sofern diese Serviceeinrichtung verfügbar ist. ICSF stellt Unterstützung für Hardwareverschlüsselung bereit und wird an Stelle der System-SSL-Softwarealgorithmen verwendet. Weitere Informationen hierzu enthält die Veröffentlichung *Cryptographic Services System SSL Programming (IBM Form SC24-5901)*.

## Schlüsseldatei mit RACF erstellen

Führen Sie diesen Schritt nicht aus, wenn Sie gskkyman für System SSL verwenden.

Der Befehl **RACDCERT** installiert und verwaltet private Schlüssel und Zertifikate in RACF. RACF unterstützt die Verwaltung mehrerer privater Schlüssel und Zertifikate in einer Gruppe. Diese Gruppen werden als Schlüsseldateien bezeichnet.

Ausführliche Informationen zum Befehl **RACDCERT** können Sie der Veröffentlichung *Security Server RACF Command Language Reference (IBM Form SA22-7687)* entnehmen.

```
RDEFINE FACILITY IRR.DIGTCERT.LIST UACC(NONE)
RDEFINE FACILITY IRR.DIGTCERT.LISTRING UACC(NONE)
PERMIT IRR.DIGTCERT.LIST CLASS(FACILITY) ACCESS(READ) ID(omvskern)
PERMIT IRR.DIGTCERT.LISTRING CLASS(FACILITY) ACCESS(READ) ID(omvskern)
SETROPTS RACLIST(FACILITY) REFRESH
```

```
RACDCERT ID(omvskern) GENCERT SUBJECTSDN(CN('wd4z rse ssl') +
OU('wd4z') O('IBM') L('Raleigh') SP('NC') C('US')) +
NOTAFTER(DATE(2017-05-21)) WITHLABEL('wd4zrse') KEYUSAGE(HANDSHAKE)
```

```
RACDCERT ID(omvskern) ADDRING(wd4zssl.racf)
RACDCERT ID(omvskern) CONNECT(LABEL('wd4zrse') RING(wd4zssl.racf) +
DEFAULT USAGE(PERSONAL))
```

Das obige Beispiel beginnt mit der Erstellung der notwendigen Profile und dem Berechtigen der Benutzer-ID OMVSKERN für den Zugriff auf die Schlüsseldateien. Die Benutzer-ID muss mit der in `/etc/inetd.conf` für den SSL-RSE-Dämon codierten Benutzer-ID übereinstimmen. Der nächste Schritt ist die Erstellung eines neuen selbst signierten Zertifikats mit der Bezeichnung `wd4zrse`. Es ist kein Kennwort erforderlich. Dieses Zertifikat wird dann zu einer neu erstellten Schlüsseldatei (`wd4zssl.racf`) hinzugefügt. Für die Schlüsseldatei ist ebenso wie für das Zertifikat kein Kennwort erforderlich. Die Lebensdauer des Zertifikats entspricht der des mit `keytool` erstellten Zertifikats.

Das Ergebnis können Sie mit der Option `list` überprüfen:

```
RACDCERT ID(omvskern) LIST
Digital certificate information for user OMVSKERN:

Label: wd4zrse
Certificate ID: 2QjW10Xi0sXZ1aaEqZmihUBA
Status: TRUST
Start Date: 2007/05/24 00:00:00
End Date: 2017/05/21 23:59:59
Serial Number:
    >00<
Issuer's Name:
    >CN=wd4z rse ssl.OU=wd4z.O=IBM.L=Raleigh.SP=NC.C=US<
Subject's Name:
    >CN=wd4z rse ssl.OU=wd4z.O=IBM.L=Raleigh.SP=NC.C=US<
Private Key Type: Non-ICSF
Private Key Size: 1024
Ring Associations:
    Ring Owner: OMVSKERN
Ring:
    >wd4zssl.racf<
```

## Schlüsseldatenbank mit `gskkyman` erstellen

Führen Sie diesen Schritt nicht aus, wenn Sie RACF für System SSL verwenden.

`gskkyman` ist ein Shell-basiertes, menügeführtes `z/OS-UNIX`-Programm, das eine `z/OS-UNIX`-Datei erstellt, mit Daten füllt und verwaltet. Diese Datei enthält private Schlüssel, Zertifikatanforderungen und Zertifikate und wird als Schlüsseldatenbank bezeichnet.

**Anmerkung:** Die Umgebung für `gskkyman` muss möglicherweise mit den folgenden Anweisungen konfiguriert werden. Weitere Informationen hierzu enthält die Veröffentlichung *System SSL Programming (IBM Form SC24-5901)*.

```
PATH=$PATH:/usr/lpp/gskssl/bin
export NLSPATH=/usr/lpp/gskssl/lib/nls/msg/En_US.IBM-1047/%N:$NLSPATH
export STEPLIB=$STEPLIB:SYS1.SIEALNKE
```

```
$ gskkyman
```

```
Database Menu
```

- 1 - Create new database
- 2 - Open database
- 3 - Change database password
- 4 - Change database record length
- 5 - Delete database
- 6 - Create key parameter file
- 0 - Exit program

Enter option number: 1  
Enter key database name (press ENTER to return to menu): wd4zssl.kdb  
Enter database password (press ENTER to return to menu): rsessl  
Re-enter database password: rsessl  
Enter password expiration in days (press ENTER for no expiration):  
Enter database record length (press ENTER to use 2500):

Key database /etc/wd4z/ssl/wd4zssl.kdb created.

Press ENTER to continue.

#### Key Management Menu

Database: /etc/wd4z/ssl/wd4zssl.kdb

- 1 - Manage keys and certificates
- 2 - Manage certificates
- 3 - Manage certificate requests
- 4 - Create new certificate request
- 5 - Receive requested certificate or a renewal certificate
- 6 - Create a self-signed certificate
- 7 - Import a certificate
- 8 - Import a certificate and a private key
- 9 - Show the default key
- 10 - Store database password
- 11 - Show database record length
  
- 0 - Exit program

Enter option number (press ENTER to return to previous menu): 6

#### Certificate Type

- 1 - CA certificate with 1024-bit RSA key
- 2 - CA certificate with 2048-bit RSA key
- 3 - CA certificate with 4096-bit RSA key
- 4 - CA certificate with 1024-bit DSA key
- 5 - User or server certificate with 1024-bit RSA key
- 6 - User or server certificate with 2048-bit RSA key
- 7 - User or server certificate with 4096-bit RSA key
- 8 - User or server certificate with 1024-bit DSA key

Select certificate type (press ENTER to return to menu): 5

Enter label (press ENTER to return to menu): wd4zrse

Enter subject name for certificate

Common name (required): wd4z rse ssl

Organizational unit (optional): wd4z

Organization (required): IBM

City/Locality (optional): Raleigh

State/Province (optional): NC

Country/Region (2 characters - required): US

Enter number of days certificate will be valid (default 365): 3650

Enter 1 to specify subject alternate names or 0 to continue: 0

Please wait .....

Certificate created.

Press ENTER to continue.

#### Key Management Menu

Database: /etc/wd4z/ssl/wd4zssl.kdb

- 1 - Manage keys and certificates
- 2 - Manage certificates

- 3 - Manage certificate requests
- 4 - Create new certificate request
- 5 - Receive requested certificate or a renewal certificate
- 6 - Create a self-signed certificate
- 7 - Import a certificate
- 8 - Import a certificate and a private key
- 9 - Show the default key
- 10 - Store database password
- 11 - Show database record length

0 - Exit program

Enter option number (press ENTER to return to previous menu): 0

```
$ ls -l
total 152
-rwxr-xr-x 1 IBMUSER SYS1      333 May 24 13:52 rsecomm.properties
-rwxr-xr-x 1 IBMUSER SYS1    6067 May 24 13:52 rsed.envvars
-rwxr-xr-x 1 IBMUSER SYS1     332 May 24 13:52 server.zseries
-rwxr-xr-x 1 IBMUSER SYS1     645 May 24 13:52 setup.env.zseries
-rwxr-xr-x 1 IBMUSER SYS1     638 May 24 13:52 ssl.properties
-rw-r--r-- 1 IBMUSER SYS1    1224 May 24 14:17 wd4zssl.jks
-rw----- 1 IBMUSER SYS1   35080 May 24 14:24 wd4zssl.kdb
-rw----- 1 IBMUSER SYS1      80 May 24 14:24 wd4zssl.rdb
$ chmod 644 wd4zssl.kdb
$ chmod 644 wd4zssl.rdb
$ ls -l
total 152
-rwxr-xr-x 1 IBMUSER SYS1      333 May 24 13:52 rsecomm.properties
-rwxr-xr-x 1 IBMUSER SYS1    6067 May 24 13:52 rsed.envvars
-rwxr-xr-x 1 IBMUSER SYS1     332 May 24 13:52 server.zseries
-rwxr-xr-x 1 IBMUSER SYS1     645 May 24 13:52 setup.env.zseries
-rwxr-xr-x 1 IBMUSER SYS1     638 May 24 13:52 ssl.properties
-rw-r--r-- 1 IBMUSER SYS1    1224 May 24 14:17 wd4zssl.jks
-rw-r--r-- 1 IBMUSER SYS1   35080 May 24 14:24 wd4zssl.kdb
-rw-r--r-- 1 IBMUSER SYS1      80 May 24 14:24 wd4zssl.rdb
```

Das obige Beispiel beginnt mit der Erstellung einer Schlüsseldatenbank mit der Bezeichnung wd4zssl.kdb mit dem Kennwort rsessl. Wenn die Datenbank vorhanden ist, wird sie mit Daten gefüllt. Dazu wird ein neues selbst signiertes Zertifikat erstellt, das unter der Bezeichnung wd4zrse und mit dem bereits für die Schlüsseldatenbank verwendeten Kennwort (rsessl) gespeichert. (Dies ist eine RSE-Anforderung.)

gskkyman legt die Schlüsseldatenbank mit einer (sehr sicheren) Bitmaske (600 Berechtigungsbits) an, die nur dem Eigner Zugriff gewährt. Die Berechtigungen müssen weniger restriktiv gesetzt werden, sofern der Dämon nicht dieselbe Benutzer-ID wie der Ersteller der Schlüsseldatenbank verwendet. Masken mit 640 Bits (Eigner mit Lese-/Schreibzugriff, Gruppe des Eigners mit Lesezugriff) oder 644 Bits (Eigner mit Lese-/Schreibzugriff; Lesezugriff für alle übrigen Benutzer) sind verwendbare Masken für den Befehl **chmod**.

Das Ergebnis können Sie überprüfen, indem Sie im Untermenü **Manage keys and certificates** die Option **Show certificate information** auswählen:

```
$ gskkyman
```

Database Menu

- 1 - Create new database
- 2 - Open database
- 3 - Change database password
- 4 - Change database record length
- 5 - Delete database
- 6 - Create key parameter file

```

0 - Exit program

Enter option number: 2
Enter key database name (press ENTER to return to menu): wd4zssl.kdb
Enter database password (press ENTER to return to menu): rsessl

    Key Management Menu

    Database: /etc/wd4z/ssl/wd4zssl.kdb

    1 - Manage keys and certificates
    2 - Manage certificates
    3 - Manage certificate requests
    4 - Create new certificate request
    5 - Receive requested certificate or a renewal certificate
    6 - Create a self-signed certificate
    7 - Import a certificate
    8 - Import a certificate and a private key
    9 - Show the default key
    10 - Store database password
    11 - Show database record length

    0 - Exit program

Enter option number (press ENTER to return to previous menu): 1

    Key and Certificate List

    Database: /etc/wd4z/ssl/wd4zssl.kdb

    1 - wd4zrse

    0 - Return to selection menu

Enter label number (ENTER to return to selection menu, p for previous list): 1

    Key and Certificate Menu

    Label: wd4zrse

    1 - Show certificate information
    2 - Show key information
    3 - Set key as default
    4 - Set certificate trust status
    5 - Copy certificate and key to another database
    6 - Export certificate to a file
    7 - Export certificate and key to a file
    8 - Delete certificate and key
    9 - Change label
    10 - Create a signed certificate and key
    11 - Create a certificate renewal request

    0 - Exit program

Enter option number (press ENTER to return to previous menu): 1

```

### Certificate Information

Label: wd4zrse  
Record ID: 14  
Issuer Record ID: 14  
Trusted: Yes  
Version: 3  
Serial number: 45356379000ac997  
Issuer name: wd4z rse ssl  
wd4z  
IBM  
Raleigh  
NC  
US  
Subject name: wd4z rse ssl  
wd4z  
IBM  
Raleigh  
NC  
US  
Effective date: 2007/05/24  
Expiration date: 2017/05/21  
Public key algorithm: rsaEncryption  
Public key size: 1024  
Signature algorithm: sha1WithRsaEncryption  
Issuer unique ID: None  
Subject unique ID: None  
Number of extensions: 3

Enter 1 to display extensions, 0 to return to menu: 0

### Key and Certificate Menu

Label: wd4zrse

- 1 - Show certificate information
- 2 - Show key information
- 3 - Set key as default
- 4 - Set certificate trust status
- 5 - Copy certificate and key to another database
- 6 - Export certificate to a file
- 7 - Export certificate and key to a file
- 8 - Delete certificate and key
- 9 - Change label
- 10 - Create a signed certificate and key
- 11 - Create a certificate renewal request
- 0 - Exit program

Enter option number (press ENTER to return to previous menu): 0

---

## Datei `ssl.properties` aktualisieren, um SSL zu aktivieren

Die Zertifikate liegen jetzt vor, so dass RSE beginnen kann, SSL zu verwenden. Welche Werte in `ssl.properties` angegeben werden müssen, hängt von den Definitionen ab, die Sie in den obigen Schritten ausgewählt haben.

```
$ oedit ssl.properties
```

```
enable_ssl=true
```

Gültige Werte sind `true` und `false` (Standardwert).

```
daemon_keydb_file=wd4zssl.racf
```

Name der gskkyman-Schlüsseldatenbank oder Name der RACF-Schlüssel-datei (nur bei Verwendung des Dämons erforderlich)

```
daemon_keydb_password=
```

Kennwort für die gskkyman-Schlüsseldatenbank oder Leereintrag für die RACF-Schlüsseldatei (nur bei Verwendung des Dämons erforderlich)

```
daemon_key_label=wd4zrse
```

Verwendete gskkyman/RACF-Bezeichnung, sofern es sich nicht um die Standardbezeichnung handelt. Wenn die Standardbezeichnung verwendet wird, muss diese Angabe auf Kommentar gesetzt werden. (Nur bei Verwendung des Dämons erforderlich.)

```
server_keystore_file=wd4zssl.jks
```

Name des keytool-Keystore

```
server_keystore_password=rsessl
```

Kennwort für den keytool-Keystore

---

## Verbindung testen

Die SSL-Hostkonfiguration ist jetzt fertig gestellt. Sie können sie testen, indem Sie eine Verbindung zur Clientkomponente von Developer für System z herstellen. Da Sie für SSL eine neue Konfiguration (durch Klonen der vorhandenen Konfiguration) erstellt haben, müssen Sie nun eine neue Verbindung mit folgenden Kenndaten definieren:

- REXEC: Verwendung des Pfads `/etc/wd4z/ssl`
- Dämon: Verwendung des Ports 4047

**Anmerkung:** Für die Ausführung einer System-SSL-Anwendung (Dämonverbindung) muss `SYS1.SIEALNKE` in der `LINKLIST` oder `STEPLIB` enthalten sein. Wenn Sie die `STEPLIB`-Methode bevorzugen, fügen Sie am Ende von `rsed.envvars` die folgende Anweisung hinzu. Beachten Sie jedoch, dass sich die Verwendung von `STEPLIB` unter `z/OS UNIX` negativ auf die Leistung auswirkt. Lesen Sie hierzu die Informationen im Abschnitt „Verwendung von `STEPLIB` vermeiden“ auf Seite 67.

- Wenn die letzte `STEPLIB`-Anweisung, die zuvor in `rsed.envvars` definiert wurde, `STEPLIB=NONE` lautet:  
`STEPLIB=SYS1.SIEALNKE`
- Wenn die letzte `STEPLIB`-Anweisung, die zuvor in `rsed.envvars` definiert wurde, nicht `STEPLIB=NONE` lautet:  
`STEPLIB=$STEPLIB:SYS1.SIEALNKE`



Wenn die Verbindung hergestellt ist, beginnen Host und Client mit dem Handshakeverfahren, um einen sicheren Pfad einzurichten. Im Rahmen dieses Handshakeverfahrens werden Zertifikate ausgetauscht. Wenn die Clientkomponente von Developer für System z das Hostzertifikat nicht erkennt, fragt sie beim Benutzer an, ob dieses Zertifikat vertrauenswürdig ist.



Abbildung 12. Hostzertifikat importieren

Der Benutzer kann dieses Zertifikat als vertrauenswürdig akzeptieren, indem er auf die Schaltfläche 'Finish' klickt. Danach wird die Verbindungsinitialisierung fortgesetzt.

**Anmerkung:** Die Dämonverbindung verwendet zwei Zertifikatpositionen (System SSL und Java SSL). Aus diesem Grund gibt es zwei verschiedene Zertifikate und somit auch zwei Bestätigungen.

Wenn der Client ein Zertifikat einmal anerkannt hat, wird dieser Dialog nicht mehr angezeigt. Die Liste vertrauenswürdiger Zertifikate kann verwaltet werden. Wählen Sie dazu **Fenster > Benutzervorgaben... > Ferne Systeme > SSL** aus, um den folgenden Dialog aufzurufen:

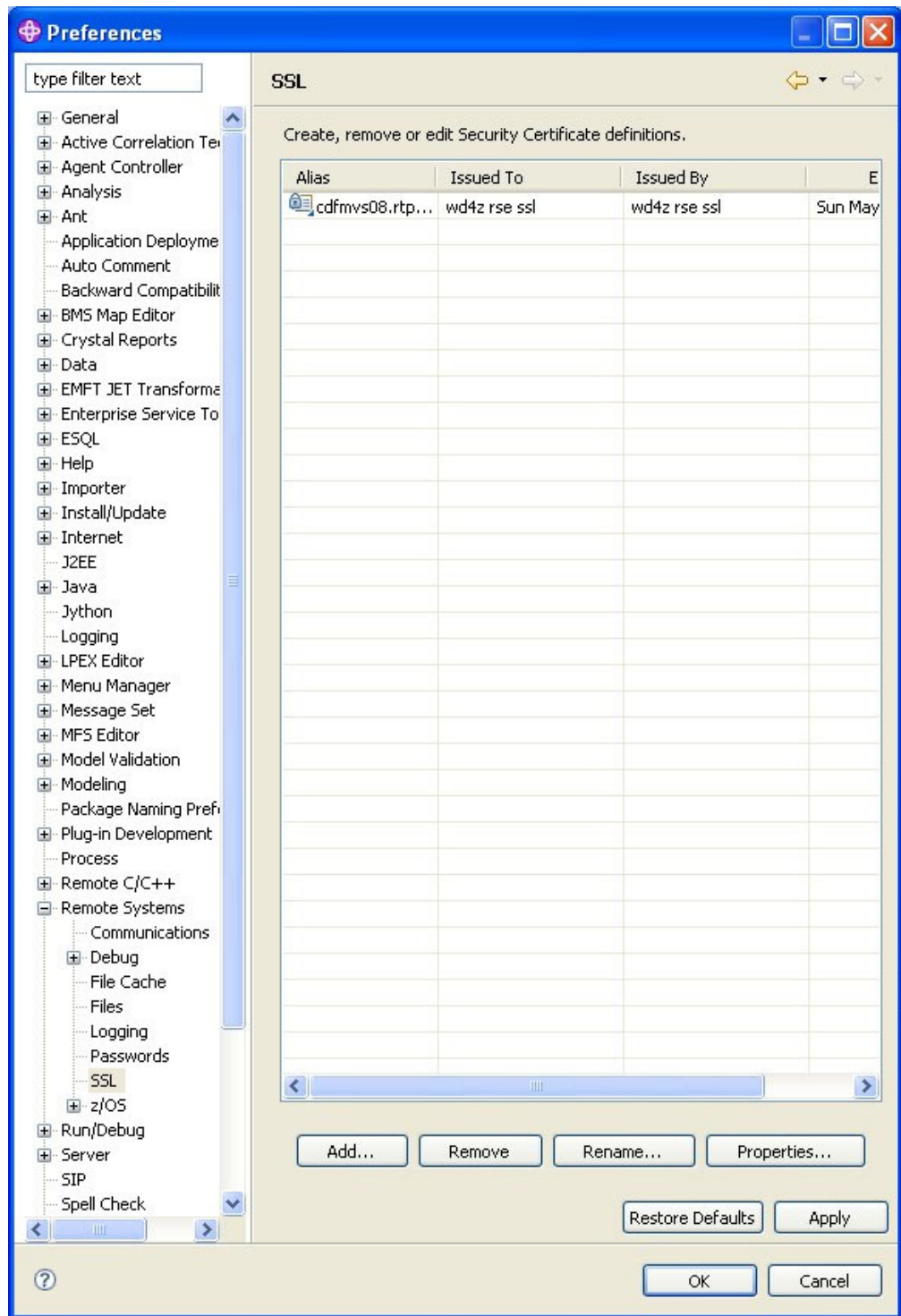


Abbildung 13. Vorgaben

Wenn die SSL-Kommunikation fehlschlägt, gibt der Client eine Fehlermeldung zurück. Weitere Informationen sind in den verschiedenen Protokolldateien verfügbar (home/.eclipse/RSE/USERID/\* und /tmp/rsedaemon.log). Lesen Sie die diesbezügliche Beschreibung im Abschnitt „RSE-Protokollierung“ auf Seite 80.

---

## Anhang F. APPC konfigurieren

Dieser Anhang soll Sie bei einigen allgemeinen Problemen unterstützen, die beim Konfigurieren von APPC (Advanced Program-to-Program Communication) oder beim Überprüfen und/oder Modifizieren einer vorhandenen Konfiguration auftreten könnten.

Zusätzliche Informationen zum APPC-Management und zu den nachfolgend beschriebenen PARMLIB-Membren enthalten die Veröffentlichungen *MVS Planning APPC/MVS Management (IBM Form SA22-7599)* und *MVS Initialization and Tuning Reference (IBM Form SA22-7592)*.

Beachten Sie, dass es sich nicht um eine komplette Konfiguration für APPC handelt. Das Beispiel hebt nur einige wichtige Aspekte hervor, die auf Ihren Standort anwendbar sein könnten.

Der Member SYS1.SAMPLIB(ATBALL) enthält eine Liste und Beschreibungen aller APPC-bezogenen Member (Beispiel-Member) in SYS1.SAMPLIB.

---

### VSAM

APPC/MVS speichert die Konfigurationsdaten in SYS1.PARMLIB-Membren und zwei VSAM-Dateigruppen:

- Die VSAM-Dateigruppe für Transaktionsprogramme (Standardname SYS1.APPCTP) enthält Zeitplanungs- und Sicherheitsinformationen für z/OS-Programme.
- Die VSAM-Dateigruppe für Nebeninformationen (Standardname SYS1.APPCSI) enthält die von lokalen z/OS-Transaktionsprogrammen und APPC/MVS-Servern verwendeten Umsetzungen der Namen symbolischer Bestimmungsorte.

Ein Transaktionsprogramm ist ein Anwendungsprogramm, das über APPC mit einem Transaktionsprogramm auf demselben oder einem anderen System kommuniziert, um auf Ressourcen zuzugreifen. Die APPC-Konfiguration für Developer für System z aktiviert ein neues Transaktionsprogramm mit dem Namen FEKFRSRV, das auch als TSO Commands Service bezeichnet wird.

Der in Abbildung 14 dargestellte Beispieljob ist eine Verkettung der Beispiel-Member SYS1.SAMPLIB(ATBTPVSM) und SYS1.SAMPLIB(ATBSIVSM) und kann zum Definieren der APPC-VSAMs verwendet werden.

```

//APPCVSAM JOB <Jobparameter>
//*
/* ACHTUNG: Dies ist keine JCL-Prozedur und kein vollständiger Job.
/* Vor Verwendung dieses Beispiels müssen Sie die folgenden
/* Änderungen vornehmen:
/* 1. Passen Sie die Jobparameter an Ihre Systemanforderungen an.
/* 2. Ersetzen Sie ***** durch die Platteneinheit für die APPC-VSAMs.
//*
//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))
/*

```

Abbildung 14. JCL zum Erstellen von APPC-VSAMs

## VTAM

APPC ist eine Implementierung des LU-6.2-Protokolls der Systemnetzwerkarchitektur (SNA). Die SNA stellt Formate und Protokolle bereit, die verschiedene physische und logische SNA-Komponenten, z. B. die logische Einheit (LU, Logical Unit), definieren. LU 6.2 ist ein Typ logischer Einheiten, der speziell für die Kommunikation zwischen Anwendungsprogrammen konzipiert ist.

Wenn Sie die SNA in MVS verwenden möchten, müssen Sie VTAM (Virtual Telecommunications Access Method) installieren und konfigurieren. Die APPC-System-Tasks können erst verwendet werden, wenn VTAM aktiv ist.

Der APPC-spezifische Teil der VTAM-Konfiguration umfasst drei Schritte:

1. Definieren Sie den verwendeten Modusnamen (z. B. APPCHOST) für VTAM, indem Sie SYS1.SAMPLIB(ATBLMODE) mit SYS1.SAMPLIB(ATBLJOB) assemblieren und eine Programmverknüpfung zu Ihrer SYS1.VTAMLIB herstellen. Weitere Details finden Sie in der Beschreibung zum Member SYS1.SAMPLIB(ATBLMODE).
2. Definieren Sie APPC/MVS als eine VTAM-Anwendung. Kopieren Sie dazu den Beispiel-Member SYS1.SAMPLIB(ATBAPPL) in eine Dateigruppe der SYS1.VTAMLST-Kette. Weitere Details finden Sie in der Beschreibung zum Member SYS1.SAMPLIB(ATBAPPL).

3. Führen Sie den Konsolbefehl **v net,act,id=atbappl** aus, um die neu definierte Anwendung zu aktivieren. (Hier steht **net** für den Namen Ihrer VTAM-STC.) Mit dem Konsolbefehl **d net,appls** können Sie überprüfen, ob die Anwendung aktiv ist. Fügen Sie den Member-Namen zu **SYS1.VTAMLST(ATCCONxx)** hinzu, wenn die Anwendung beim Start von VTAM aktiviert werden soll.

Der im Beispiel-Member **SYS1.SAMPLIB(ATBAPPL)** verwendete **ACBNAME (MVS-LU01)** kann an die Standards Ihres Standortes angepasst werden. Er muss jedoch mit den Definitionen im Member **SYS1.PARMLIB(APPCPMxx)** übereinstimmen.

```

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

```

Abbildung 15. **SYS1.SAMPLIB(ATBAPPL)**

Weitere Informationen zum Konfigurieren von VTAM finden Sie im *Communications Server Bookshelf (F1A1BK61 for z/OS 1.7)*.

---

## **SYS1.PARMLIB(APPCPMxx)**

Zur Aktivierung und Unterstützung des Datenaustauschs zwischen Systemen müssen an Standorten LUs (logische Einheiten) definiert werden, zwischen denen Sitzungen stattfinden können. Voraussetzung für eine APPC/MVS-Verarbeitung - auch auf einem einzelnen System - ist, dass an einem Standort mindestens eine LU definiert ist. LUs gehören zu den Elementen, die in **SYS1.PARMLIB(APPCPMxx)** definiert werden.

Der TSO Commands Service erfordert, dass APPC mit einer Basis-LU für die Bearbeitung ein- und abgehender Anforderungen konfiguriert ist.

Die LU-Definition muss zum Member **SYS1.PARMLIB(APPCPMxx)** hinzugefügt werden und die Parameter **BASE** und **SCHED(ASCH)** enthalten. Der Member **APPCPMxx** gibt auch an, welche VSAM-Dateigruppen mit Transaktionsprofilen (TP) und Nebeninformationen (SI) verwendet werden sollen.

Abb. 16 auf Seite 126 zeigt ein Beispiel für einen **SYS1.PARMLIB(APPCPMxx)**-Member, der für den TSO Commands Service verwendet werden kann.

```

LUADD
  ACBNAME(MVSLU01)
BASE
  SCHED(ASCH)
  TPDATA(SYS1.APPCTP)
SIDEINFO DATASET(SYS1.APPCSI)

```

Abbildung 16. *SYS1.PARMLIB(APPCPMxx)*

Wenn ein System mehrere LU-Namen hat, müssen Sie ggf. Änderungen vornehmen, je nachdem, welche LU das System als Basis-LU auswählt. Die Basis-LU für das System wird wie folgt bestimmt:

1. Die Basis-LU des Systems wird von der letzten LUADD-Anweisung angegeben, die die beiden Parameter NOSCHED und BASE enthält. Mit diesem Basis-LU-Typ des Systems können abgehende Anforderungen verarbeitet werden, wenn keine Transaktions-Scheduler aktiv sind.
2. Wenn es keine LUADD-Anweisung gibt, die sowohl NOSCHED als auch BASE enthält, wird die Basis-LU des Systems von der letzten LUADD-Anweisung repräsentiert, die den Parameter BASE enthält und ASCH als APPC/MVS-Transaktions-Scheduler angibt. Dazu muss entweder SCHED(ASCH) codiert sein, oder der Parameter SCHED muss gar nicht codiert sein (denn ASCH ist der Standardwert für SCHED).

Falls es auf Ihrem System eine LU mit den Parametern BASE und NOSCHED gibt, wird diese LU als Basis-LU verwendet werden, der TSO Commands Service würde jedoch nicht funktionieren, weil diese LU keinen Transaktions-Scheduler für die Bearbeitung von Anforderungen an die Transaktion FEKFRSRV hat. Wenn Sie den Parameter NOSCHED dieser LU nicht entfernen können, setzen Sie die Umgebungsvariable `_FEKFSCMD_PARTNER_LU` in `rsed.envvars` auf die LU, für die die Parameter BASE und SCHED(ASCH) definiert sind. Beispiel:

```
_FEKFSCMD_PARTNER_LU=MVSLU01
```

Weitere Informationen zu `rsed.envvars` enthält der Abschnitt „RSE-Konfigurationsdatei `rsed.envvars` anpassen“ auf Seite 35.

---

## SYS1.PARMLIB(ASCHPMxx)

Der APPC/MVS-Transaktions-Scheduler (mit dem Standardnamen ASCH) leitet bei eingehenden Dialoganforderungen Transaktionsprogramme ein und steuert den zeitlichen Ablauf dieser Programme. Der Member `SYS1.PARMLIB(ASCHPMxx)` steuert seine Funktionen unter anderem mit Transaktionsklassendefinitionen.

Die für den TSO Commands Service verwendete APPC-Transaktionsklasse muss genug APPC-Initiatoren haben, damit für jeden Benutzer von Developer für System z ein Initiator verfügbar ist.

**Anmerkung:** Es gibt einen Unterschied zwischen APPC-Initiatoren und z/OS-Initiatoren (JES). Wenn ein Developer für System z Client eine Verbindung zum Host herstellt, wird der TSO Commands Server unter Verwendung des APPC-Initiators gestartet. Developer für System z verwendet einen z/OS-Initiator (JES), wenn ein Projekt-Build erstellt, eine ferne Syntaxprüfung durchgeführt oder ein Job übergeben wird.

Der TSO Commands Service erfordert außerdem die Angabe der Standard-spezifikationen in den Abschnitten `OPTIONS` und `TPDEFAULT`.

Abb. 17 zeigt ein Beispiel für einen SYS1.PARMLIB(ASCHPMxx)-Member, der für den TSO Commands Service verwendet werden kann.

```
CLASSADD  
CLASSNAME(A)  
MAX(20)  
MIN(1)  
    MSGLIMIT(200)  
  
OPTIONS  
    DEFAULT(A)  
  
TPDEFAULT  
REGION(2M)  
TIME(5)  
MSGLEVEL(1,1)  
    OUTCLASS(X)
```

Abbildung 17. SYS1.PARMLIB(ASCHPMxx)

**Anmerkung:** Das IBM Support Center kann Sie bitten, für das Debugging den Wert von MSGLIMIT zu erhöhen, damit mehr Ausgaben in die Protokolldatei geschrieben werden.

---

## APPC-Änderungen aktivieren

Die in den obigen Schritten dokumentierten Konfigurationsänderungen können jetzt aktiviert werden. Hierfür gibt es - je nach vorliegender Situation - verschiedene Möglichkeiten:

- Wenn APPC noch nicht aktiv ist, geben Sie die folgenden Konsolbefehle ein, um APPC/MVS zu starten. (Hier steht xx für die beiden letzten Zeichen der zugehörigen SYS1.PARMLIB-Member.)
  1. S APPC,SUB=MSTR,APPC=xx
  2. S ASCH,SUB=MSTR,ASCH=xxFügen Sie diese Befehle zu SYS1.PARMLIB(COMMNDxx) hinzu, damit sie beim Systemstart gestartet werden.
- Wenn APPC bereits aktiv ist, kann APPC die SYS1.PARMLIB-Member mit einem Konsolbefehl SET dynamisch neu laden. (Hier steht xx für die beiden letzten Zeichen der zugehörigen SYS1.PARMLIB-Member.)
  1. SET APPC=xx
  2. SET ASCH=xx

Mit den Konsolbefehlen **D APPC** und **D ASCH** kann die APPC-Konfiguration überprüft werden. Weitere Informationen zu den angegebenen Konsolbefehlen enthält die Veröffentlichung *MVS System Commands (IBM Form SA22-7627)*.

---

## Transaktion für den TSO Commands Service definieren

Wenn APPC/MVS aktiv ist, kann der TSO Commands Service von Developer für System z definiert werden. Lesen Sie hierzu die Beschreibung im Abschnitt „APPC-Transaktion für den TSO Commands Service definieren“ auf Seite 24.

Für eine dokumentierte Definition der APPC-Transaktion müssen Sie HLQ.SFEKSAMP(FEKAPPCC) anpassen und übergeben. HLQ steht hier für den High Level Qualifier, der während der Installation von Developer für System z verwendet wurde (standardmäßig FEK).

Die APPC-Transaktion kann auch interaktiv über die APPC-ISPF-Schnittstelle definiert werden. Diese Schnittstelle ist in einem White Paper dokumentiert. In diesem White Paper ist auch beschrieben, wie die APPC-Transaktion für die Erfassung benutzerspezifischer Accountinformationen konfiguriert werden kann.

Das White Paper *APPC and WebSphere Developer for System z* (IBM Form SC23-5885) ist in der Internetbibliothek zu Developer für System z unter <http://www-306.ibm.com/software/awdtools/devzseries/library/> verfügbar.

**Anmerkung:** In Developer für System z Version 7.1 hat sich die von APPC zum Starten des TSO Commands Service verwendete Transaktionsprogramm-JCL geändert. Wenn Sie den im White Paper beschriebenen Anweisungen für das Definieren des Transaktionsprogramms folgen, müssen Sie das Schlüsselwort NESTMACS zur PARM-Zeile hinzufügen. Beispiel:

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



---

# Glossar

## A

**Aktions-ID.** Eine numerische Kennung zwischen 0 und 999 für eine Aktion.

### Antwortdatei.

1. Eine Datei, die vordefinierte Antworten auf Fragen enthält, die ein Programm stellt. Die Antworten werden verwendet, so dass diese Werte nicht einzeln eingegeben werden müssen.
2. Eine ASCII-Datei, die mit Installations- und Konfigurationsdaten angepasst werden kann, die eine Installation automatisieren. Die Installations- und Konfigurationsdaten müssen während einer interaktiven Installation eingegeben werden, aber mit einer Antwortdatei kann die Installation ohne jeglichen Benutzereingriff durchgeführt werden.

### Anwendungsserver.

1. Ein Programm, das alle Anwendungsoperationen zwischen browserbasierten Computern und den Back-End-Geschäftsanwendungen oder -Datenbanken einer Organisation bearbeitet. Es gibt eine spezielle Klasse von Java-basierten Anwendungsservern, die dem Standard J2EE entsprechen. J2EE-Code kann ohne großen Aufwand zwischen diesen Anwendungsservern portiert werden. Diese Anwendungsserver können JSPs und Servlets für dynamischen Webinhalt und EJBs für Transaktionen und Datenbankzugriffe unterstützen.
2. Das Ziel einer Anforderung, die von einer fernen Anwendung stammt. In der DB2-Umgebung wird die Anwendungsserverfunktion von einer verteilten Dateneinrichtung bereitgestellt und für den Zugriff auf DB2-Daten in fernen Anwendungen verwendet.
3. Ein Serverprogramm in einem verteilten Netz, das die Ausführungsumgebung für ein Anwendungsprogramm bereitstellt.
4. Das Ziel einer Anforderung, die von einem Anwendungs-Requester stammt. Das Datenbankmanagementsystem (DBMS) auf der Anwendungsserversite stellt die angeforderten Daten bereit.
5. Software, die die Kommunikation mit dem Client, der ein Asset anfordert, und Abfragen des Content Manager bearbeitet.

**Ausgabesicht.** Zeigt Nachrichten, Parameter und Ergebnisse an, die sich auf die von Ihnen bearbeiteten Objekte beziehen.

**Ausgabesicht der Konsole.** Zeigt die Ausgabe eines Prozesses an und ermöglicht Ihnen, über die Tastatur Eingaben an einen Prozess zu senden.

## B

**Bidirektional (BIDI).** Bezeichnung für Scripts in Sprachen wie Arabisch und Hebräisch, die im Allgemeinen von rechts nach links geschrieben werden. Ausnahmen sind Zahlen, die von links nach rechts geschrieben werden. Diese Definition stammt aus dem LISA-Glossar (Localization Industry Standards Association).

**Bidirektionales Attribut.** Texttyp, Textausrichtung, numerische Ersetzung und symmetrische Ersetzung.

**Build-Anforderung.** Eine Anforderung eines Clients zum Ausführen einer Build-Transaktion.

**Build-Transaktion.** Ein unter MVS gestarteter Job, der Builds erstellt, wenn vom Client eine Build-Anforderung empfangen wird.

## C

### Container.

1. In CoOperative Development Environment/400 ein Systemobjekt, das Quellendateien enthält und organisiert. Beispiele für einen Container sind eine i5/OS-Bibliothek und eine partitionierte MVS-Datei.
2. In J2EE eine Entität, die Komponenten Sicherheits-, Deployment- und Laufzeitservices sowie Services für die Verwaltung des Lebenszyklus bereitstellt. (Sun) Jeder Containertyp (EJB, Web, JSP, Servlet, Applet und Anwendungsclient) stellt außerdem komponentenspezifische Services bereit.
3. In Backup Recovery and Media Services das physische Objekt, das zum Lagern und Umlagern von Datenträgern verwendet wird, wie z. B. Boxen, Schachteln oder Regale.
4. In einem Virtual Tape Server (VTS) ein Behälter, in dem exportierte logische Datenträger gespeichert werden können. Ein Stapeldatenträger mit einem oder mehreren logischen Datenträger(n), der sich außerhalb einer VTS-Bibliothek befindet, wird als Container für diese Datenträger betrachtet.
5. Eine physische Speicherposition der Daten, z. B. eine Datei, ein Verzeichnis oder eine Einheit.
6. Eine Spalte oder Zeile, die verwendet wird, um das Layout eines Portlets oder anderer Container auf einer Seite zu gestalten.
7. Ein Element der Benutzerschnittstelle, das Objekte enthält. Im Ordnermanager ein Objekt, das andere Ordner oder Dokumente enthalten kann.

## D

**Dateigruppe.** Die Haupteinheit für das Speichern und Abrufen von Daten, die sich aus einer Sammlung von Daten in einer von mehreren vorgegebenen Zusammenstellungen zusammensetzt und durch Steuerinformationen beschrieben wird, auf die das System Zugriff hat.

**Datenbank.** Eine Sammlung von in Wechselbeziehung zueinander stehenden oder unabhängigen Datenelementen, die zur Bereitstellung für eine oder mehrere Anwendung(en) zusammen gespeichert werden.

**Datendefinitionssicht.** Enthält eine lokale Darstellung von Datenbanken und ihren Objekten und stellt Features für die Bearbeitung dieser Objekte und deren Export in eine ferne Datenbank bereit.

**Debug.** Fehler in Programmen finden, diagnostizieren und beheben.

**Debug-Sitzung.** Die Debug-Aktivitäten, die in dem Zeitraum zwischen dem Starten eines Debuggers durch den Entwickler und dem Beenden des Debuggers stattfinden.

## F

**Fehlerpuffer.** Ein Teil des Speichers, in dem Fehlernachrichten vorübergehend gespeichert werden.

**Fernes Dateisystem.** Ein Dateisystem, das sich auf einem anderen Server oder Betriebssystem befindet.

**Fernes System.** Jedes andere System im Netz, mit dem Ihr System kommunizieren kann.

## G

**Gateway.**

1. Eine Middlewarekomponente, die eine Brücke zwischen Internet und Intranet-Umgebungen während Web-Service-Aufrufen bildet.
2. Software, die Services zwischen Endpunkten und dem Rest der Tivoli-Umgebung bereitstellt.
3. Eine Komponente eines Voice over Internet Protocol, die eine Brücke zwischen VoIP und Umgebungen mit Wahlverbindungen darstellt.
4. Eine Einheit oder ein Programm, mit der bzw. dem Netze oder Systeme mit unterschiedlichen Netzarchitekturen miteinander verbunden werden können. Die Systeme können unterschiedliche Merkmale haben, z. B. unterschiedliche Kommunikationsprotokolle, unterschiedliche Netzarchitekturen oder unterschiedliche Sicherheitsrichtlinien. In diesem Fall übernimmt das Gateway sowohl eine Umsetzungs- als auch eine Verbindungsrolle.

## I

**Interactive System Productivity Facility (ISPF).** Ein IBM Lizenzprogramm, das als Gesamtanzeigeditor und Dialogmanager eingesetzt wird. Das Programm wird für das Schreiben von Anwendungsprogrammen verwendet und ermöglicht dem Benutzer, Standardanzeigen und Dialoge zwischen dem Anwendungsprogrammierer und dem Endbenutzer zu generieren. ISPF setzt sich aus vier Hauptkomponenten zusammen: DM, PDF, SCLM und C/S. Die Komponente DM ist der Dialog Manager, der die Services für Dialoge und Endbenutzer bereitstellt. Die Komponente PDF ist die Program Development Facility, die Services für die Unterstützung von Dialog- und Anwendungsentwicklern bereitstellt. Die Komponente SCLM ist der Software Configuration Library Manager, der Anwendungsentwickler Services für die Verwaltung Ihrer Anwendungsentwicklungsbibliotheken bereitstellt. Die Komponente C/S ist die Client/Server-Komponente, die es Ihnen ermöglicht, ISPF auf programmierbaren Workstations auszuführen, um die Anzeigen mit der Anzeigefunktion des Workstationbetriebssystems anzuzeigen und Workstation-Tools und -daten in Host-Tools und -daten zu integrieren.

**Interpreter.** Ein Programm, das jede Instruktion einer höheren Programmiersprache übersetzt und ausführt, bevor es die nächste Instruktion übersetzt und ausführt.

**Isomorph.** Jedes zusammengesetzte Element (in anderen Worten jedes Element, das weitere Elemente enthält) des XML-Instanzdokuments hat ausgehend vom Stammverzeichnis genau ein entsprechendes COBOL-Gruppenelement, dessen Verschachtelungstiefe mit der Verschachtelungstiefe seines XML-Äquivalents identisch ist. Jedes nicht zusammengesetzte Element (in anderen Worten jedes Element, das keine weiteren Elemente enthält) im XML-Instanzdokument hat ausgehend vom Stamm genau ein entsprechendes Datenelement, dessen Verschachtelungstiefe mit der Verschachtelungstiefe seines XML-Äquivalents identisch ist und dessen Speicheradresse zur Laufzeit eindeutig identifiziert werden kann.

## K

**Kompilieren.**

1. In ILE-Sprachen (Integrated Language Environment) das Umsetzen von Quellenanweisungen in Module, die anschließend in Programme oder Serviceprogramme eingebunden werden können.
2. Das Umsetzen eines vollständigen Programms oder von Teilen eines Programms, das in einer höheren Programmiersprache geschrieben ist, in ein Computerprogramm in IL, Assemblersprache oder Maschinensprache.

## L

**Ladebibliothek.** Eine Bibliothek mit Lademodulen.

**LINKAGE SECTION.** Der Abschnitt im Datenteil einer aktivierten Einheit (einem aufgerufenen Programm oder einer aufgerufenen Methode), der Datenelemente beschreibt, die von der aktivierten Einheit (Programm oder Methode) zur Verfügung gestellt werden. Die aktivierte Einheit und die aktivierende Einheit können auf diese Datenelemente verweisen.

## N

**Navigatoransicht.** Eine hierarchische Sicht der Ressourcen in der Workbench.

**Nicht isomorph.** Eine einfache Zuordnung von COBOL-Elementen und XML-Elementen, die zu XML-Dokumenten und COBOL-Gruppen gehören, die keine identische Form haben (nicht isomorph sind). Eine nicht isomorphe Zuordnung kann auch zwischen nicht isomorphen Elementen isomorpher Strukturen erstellt werden.

## P

**Perspektive.** Eine Gruppe von Sichten, die verschiedene Aspekte der Ressourcen in der Workbench zeigen. Der Workbench-Benutzer kann - je nach auszuführender Task - die Perspektive wechseln und auch das Layout der Sichten und Editoren innerhalb einer Perspektive anpassen.

**Perspektive für ferne Systeme.** Eine Schnittstelle für die Verwaltung ferner Systeme unter Einhaltung von Konventionen, die ISPF ähnlich sind.

## R

**RAM.** Repository Access Manager

**Repository.**

1. Ein Speicherbereich für Daten. Jedes Repository hat einen Namen und einen zugehörigen Geschäftselementtyp. Standardmäßig ist der Repository-Name identisch mit dem Namen des Geschäftselements. Beispielsweise hat ein Repository für Rechnungen den Namen Rechnungen. Es gibt zwei Typen von Informations-Repositories: lokale (prozessspezifische) und globale (wiederverwendbare) Repositories.
2. Eine VSAM-Datei, in der die Status von BTS-Prozessen gespeichert werden. Wenn ein Prozess nicht unter der Steuerung von BTS ausgeführt wird, werden der Prozessstatus (und die Status der zugehörigen Aktivitäten) erhalten, indem sie in eine Repository-Datei geschrieben werden. Die Status aller Prozesse eines bestimmten Prozesstyps (und der

zugehörigen Aktivitätsinstanzen) werden in derselben Repository-Datei gespeichert. Es können Datensätze für mehrere Prozesstypen in dasselbe Repository geschrieben werden.

3. Ein permanenter Speicherbereich für Quellcode und andere Anwendungsressourcen. In einer Teamprogrammierungsumgebung ermöglicht ein gemeinsam benutztes Repository den Zugriff mehrerer Benutzer auf Anwendungsressourcen.
4. Eine Sammlung von Informationen über die Warteschlangenmanager, die zu einem Cluster gehören. Zu diesen Informationen gehören die Namen der Warteschlangenmanager, ihre Positionen, ihre Channel, die zugehörigen Warteschlangen usw.

**Repository-Instanz.** Ein Projekt oder eine Komponente, das bzw. die in einem SCM-System vorhanden ist.

**Repository-Sicht.** Zeigt die CVS-Repository-Positionen an, die Ihrer Workbench hinzugefügt wurden.

## S

**Serveransicht.** Zeigt eine Liste mit allen Servern und den zugehörigen Konfigurationen an.

**Shell.** Eine Softwareschnittstelle zwischen Benutzern und dem Betriebssystem, die Befehle und Benutzerinteraktionen interpretiert und diese an das Betriebssystem übermittelt. Ein Computer kann mehrere Shell-Ebenen für unterschiedliche Ebenen von Benutzerinteraktionen haben.

**Shell-Name.** Der Name der Shell-Schnittstelle.

**Shell-Script.** Eine Datei mit Befehlen, die von der Shell interpretiert werden können. Der Benutzer gibt den Namen der Script-Datei an der Shell-Eingabeaufforderung ein und veranlasst die Shell damit, die Script-Befehle auszuführen.

**Sidedeck.** Eine Bibliothek, in der die Funktionen eines DLL-Programms veröffentlicht werden. Die Eintrags- und Modulnamen werden nach der Kompilierung des Quellcodes in der Bibliothek gespeichert.

**Sperraktion.** Sperrt einen Member.

## T

**Task-Liste.** Eine Liste mit Prozeduren, die in einem Steuerungsablauf ausgeführt werden können.

## U

**Unbeaufsichtigte Deinstallation.** Ein Deinstallationsprozess, bei dem keine Nachrichten an die Konsole

gesendet werden, sondern Nachrichten und Fehler nach dem Aufruf des Deinstallationsbefehls in Protokolldateien gespeichert werden.

**Unbeaufsichtigte Installation.** Eine Installation, bei der keine Nachrichten an die Konsole gesendet, sondern Nachrichten und Fehler in Protokolldateien gespeichert werden. Bei einer unbeaufsichtigten Installation können Antwortdateien für die Dateneingabe verwendet werden.

**URL.** Uniform Resource Locator

---

## Bemerkungen

Die vorliegenden Informationen wurden für Produkte und Services entwickelt, die auf dem deutschen Markt angeboten werden.

Möglicherweise bietet IBM die in dieser Dokumentation beschriebenen Produkte, Services oder Funktionen in anderen Ländern nicht an. Informationen über die gegenwärtig im jeweiligen Land verfügbaren Produkte und Services sind beim IBM Ansprechpartner erhältlich. Hinweise auf IBM Lizenzprogramme oder andere IBM Produkte bedeuten nicht, dass nur Programme, Produkte oder Services von IBM verwendet werden können. An Stelle der Produkte, Programme oder Dienstleistungen können auch andere, ihnen äquivalente Produkte, Programme oder Dienstleistungen verwendet werden, solange diese keine gewerblichen oder anderen Schutzrechte der IBM verletzen. Die Verantwortung für den Betrieb von Fremdprodukten, Fremdprogrammen und Fremddienstleistungen liegt beim Kunden.

Für in diesem Handbuch beschriebene Erzeugnisse und Verfahren kann es IBM Patente oder Patentanmeldungen geben. Mit der Auslieferung dieses Handbuchs ist keine Lizenzierung dieser Patente verbunden. Lizenzanforderungen sind schriftlich an folgende Adresse zu richten (Anfragen an diese Adresse müssen auf Englisch formuliert werden):

IBM Director of Licensing  
IBM Europe, Middle East & Africa  
Tour Descartes  
2, avenue Gambetta  
92066 Paris La Defense  
France

Trotz sorgfältiger Bearbeitung können technische Ungenauigkeiten oder Druckfehler in dieser Veröffentlichung nicht ausgeschlossen werden. Die Angaben in dieser Dokumentation werden in regelmäßigen Zeitabständen aktualisiert. Die Änderungen werden in Überarbeitungen bzw. neuen Auflagen der Veröffentlichung bekanntgegeben. IBM kann ohne weitere Mitteilung jederzeit Verbesserungen und/oder Änderungen an den in dieser Veröffentlichung beschriebenen Produkten und/oder Programmen vornehmen.

Verweise in diesen Informationen auf Websites anderer Anbieter dienen lediglich als Benutzerinformationen und stellen keinerlei Billigung des Inhalts dieser Websites dar. Das über diese Websites verfügbare Material ist nicht Bestandteil des Materials für dieses IBM Produkt; die Verwendung dieser Websites geschieht auf eigene Verantwortung.

Werden an IBM Informationen eingesandt, können diese beliebig verwendet werden, ohne dass eine Verpflichtung gegenüber dem Einsender entsteht.

Lizenznehmer des Programms, die Informationen zu diesem Produkt wünschen mit der Zielsetzung: (i) den Austausch von Informationen zwischen unabhängigen, erstellten Programmen und anderen Programmen (einschließlich des vorliegenden Programms) sowie (ii) die gemeinsame Nutzung der ausgetauschten Informationen zu ermöglichen, wenden sich an folgende Adresse:

IBM Corporation  
P.O. Box 12195, Dept. TL3B/B503/B313  
3039 Cornwallis Rd.  
Research Triangle Park, NC 27709-2195  
USA

Die Bereitstellung dieser Informationen kann unter Umständen von bestimmten Bedingungen - in einigen Fällen auch von der Zahlung einer Gebühr - abhängig sein.

Die Lieferung des im Handbuch aufgeführten Lizenzprogramms sowie des zugehörigen Lizenzmaterials erfolgt im Rahmen der Allgemeinen Geschäftsbedingungen der IBM, der Internationalen Nutzungsbedingungen der IBM für Programmpakete oder einer äquivalenten Vereinbarung.

Alle in diesem Dokument enthaltenen Leistungsdaten stammen aus einer kontrollierten Umgebung. Die Ergebnisse, die in anderen Betriebsumgebungen erzielt werden, können daher erheblich von den hier erzielten Ergebnissen abweichen. Einige Daten stammen möglicherweise von Systemen, deren Entwicklung noch nicht abgeschlossen ist. Eine Gewährleistung, dass diese Daten auch in allgemein verfügbaren Systemen erzielt werden, kann nicht gegeben werden. Darüber hinaus wurden einige Daten unter Umständen durch Extrapolation berechnet. Die tatsächlichen Ergebnisse können davon abweichen. Benutzer dieses Dokuments sollten die entsprechenden Daten in ihrer spezifischen Umgebung prüfen.

Alle Informationen zu Produkten anderer Anbieter stammen von den Anbietern der aufgeführten Produkte, deren veröffentlichten Ankündigungen oder anderen allgemein verfügbaren Quellen. IBM hat diese Produkte nicht getestet und kann daher keine Aussagen zu Leistung, Kompatibilität oder anderen Merkmalen machen. Fragen zu den Leistungsmerkmalen von Produkten anderer Anbieter sind an den jeweiligen Anbieter zu richten.

Die oben genannten Erklärungen bezüglich der Produktstrategien und Absichtserklärungen von IBM stellen die gegenwärtige Absicht von IBM dar, unterliegen Änderungen oder können zurückgenommen werden und repräsentieren nur die Ziele von IBM.

Diese Veröffentlichung enthält Beispiele für Daten und Berichte des alltäglichen Geschäftsablaufes. Sie sollen nur die Funktionen des Lizenzprogramms illustrieren; sie können Namen von Personen, Firmen, Marken oder Produkten enthalten. Alle diese Namen sind frei erfunden; Ähnlichkeiten mit tatsächlichen Namen und Adressen sind rein zufällig.

#### COPYRIGHTLIZENZ:

Diese Veröffentlichung enthält Beispielanwendungsprogramme, die in Quellsprache geschrieben sind. Diese Beispiele wurden nicht unter allen denkbaren Bedingungen getestet. IBM kann deshalb nicht garantieren, dass die Zuverlässigkeit, Wartungsfreundlichkeit und Funktion dieser Programme gegeben ist. Sie dürfen diese Beispielprogramme kostenlos kopieren, ändern und verteilen, wenn dies zu dem Zweck geschieht, Anwendungsprogramme zu entwickeln, verwenden, vermarkten oder zu verteilen, die mit der Anwendungsprogrammierschnittstelle konform sind, für die diese Beispielprogramme geschrieben wurden.

Kopien oder Teile der Musterprogramme bzw. daraus abgeleiteter Code müssen folgenden Copyrightvermerk beinhalten:

© (Name Ihrer Firma) (Jahr). Teile des vorliegenden Codes wurden aus Beispielprogrammen der IBM Corp. abgeleitet. © Copyright IBM Corp. \_Jahr/Jahre angeben\_. Alle Rechte vorbehalten.

---

## Marken und Servicemarken

Folgende Namen sind Marken oder eingetragene Marken der International Business Machines Corporation in den USA und/oder anderen Ländern:

- CICS
- CICSplex
- DB2
- IBM
- IMS
- MVS
- OS/390
- RACF
- Rational
- System z
- VTAM
- WebSphere
- z/OS
- zSeries

Java und alle Java-basierten Marken und Logos sind Marken oder eingetragene Marken von Sun Microsystems, Inc. in den USA und/oder anderen Ländern.

Microsoft, Windows, Windows NT und das Windows-Logo sind Marken oder eingetragene Marken der Microsoft Corporation in den USA und/oder anderen Ländern.

UNIX ist eine eingetragene Marke von The Open Group.

Andere Unternehmens-, Produkt- und Servicenamen können Marken oder Servicemarken anderer Unternehmen sein.





---

# Antwort

IBM Rational Developer für System z  
Hostkonfiguration  
Version 7.1.1

IBM Form SC12-4062-01

Anregungen zur Verbesserung und Ergänzung dieser Veröffentlichung nehmen wir gerne entgegen. Bitte informieren Sie uns über Fehler, ungenaue Darstellungen oder andere Mängel.

Zur Klärung technischer Fragen sowie zu Liefermöglichkeiten und Preisen wenden Sie sich bitte entweder an Ihre IBM Geschäftsstelle, Ihren IBM Geschäftspartner oder Ihren Händler.

**Unsere Telefonauskunft "HALLO IBM" (Telefonnr.: 0180 3 313233) steht Ihnen ebenfalls zur Klärung allgemeiner Fragen zur Verfügung.**

Kommentare:

Danke für Ihre Bemühungen.

Sie können ihre Kommentare betr. dieser Veröffentlichung wie folgt senden:

- Als Brief an die Postanschrift auf der Rückseite dieses Formulars
- Als E-Mail an die folgende Adresse: [ibmterm@de.ibm.com](mailto:ibmterm@de.ibm.com)

\_\_\_\_\_  
Name

\_\_\_\_\_  
Adresse

\_\_\_\_\_  
Firma oder Organisation

\_\_\_\_\_  
Rufnummer

\_\_\_\_\_  
E-Mail-Adresse

IBM Deutschland GmbH  
SW TSC Germany

71083 Herrenberg





Programmnummer: 5724-T07

SC12-4062-01

