

IBM Rational Developer for System z



# Guide de configuration de l'hôte

*Version 7.1.1*



IBM Rational Developer for System z



# Guide de configuration de l'hôte

*Version 7.1.1*

**Important**

Avant d'utiliser le présent document et le produit associé, prenez connaissance des informations générales figurant à la section «Remarques», à la page 141.

**Remarque**

Certaines illustrations de ce manuel ne sont pas disponibles en français à la date d'édition.

**Première édition - décembre 2007**

Réf. US : SC23-7658-01

LE PRESENT DOCUMENT EST LIVRE EN L'ETAT SANS AUCUNE GARANTIE EXPLICITE OU IMPLICITE. IBM DECLINE NOTAMMENT TOUTE RESPONSABILITE RELATIVE A CES INFORMATIONS EN CAS DE CONTREFACON AINSI QU'EN CAS DE DEFAUT D'APTITUDE A L'EXECUTION D'UN TRAVAIL DONNE.

Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. Les informations qui y sont fournies sont susceptibles d'être modifiées avant que les produits décrits ne deviennent eux-mêmes disponibles. En outre, il peut contenir des informations ou des références concernant certains produits, logiciels ou services non annoncés dans ce pays. Cela ne signifie cependant pas qu'ils y seront annoncés.

Pour plus de détails, pour toute demande d'ordre technique, ou pour obtenir des exemplaires de documents IBM, référez-vous aux documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial.

Vous pouvez également consulter les serveurs Internet suivants :

- <http://www.fr.ibm.com> (serveur IBM en France)
- <http://www.can.ibm.com> (serveur IBM au Canada)
- <http://www.ibm.com> (serveur IBM aux Etats-Unis)

*Compagnie IBM France  
Direction Qualité  
Tour Descartes  
92066 Paris-La Défense Cedex 50*

© Copyright IBM France 2007. Tous droits réservés.

© Copyright International Business Machines Corporation 2005, 2007. All rights reserved.

---

# Table des matières

<b>Figures</b>	<b>vii</b>
----------------	------------

<b>Tableaux</b>	<b>ix</b>
-----------------	-----------

<b>A propos de ce manuel</b>	<b>xi</b>
------------------------------	-----------

Public concerné par ce manuel	xi
-------------------------------	----

Publications référencées	xi
--------------------------	----

## Chapitre 1. Installation et configuration des composants de l'hôte . . . . . 1

Remarques relatives à la pré-installation	2
---	---

Préparation de la configuration	3
---------------------------------	---

Configuration nécessaire pour les logiciels et produits requis	3
--	---

Remarques relatives à l'ID utilisateur	4
--	---

Remarques relatives au serveur	5
--------------------------------	---

Droits requis pour l'implémentation des tâches de configuration	6
---	---

Remarques relatives au prédéploiement	6
---------------------------------------	---

IBM Rational Developer for System z, FMID	
---	--

HHOP710	6
---------	---

Gestionnaire IBM CARMA (Common Access	
---------------------------------------	--

Repository Manager), FMID HCMA710	7
-----------------------------------	---

## Chapitre 2. Modifications de l'installation et de la configuration . . . . . 9

Modifications entre la version 7.0 et la version 7.1	9
--	---

IBM Rational Developer for System z, FMID	
---	--

HHOP710	9
---------	---

Gestionnaire IBM CARMA (Common Access	
---------------------------------------	--

Repository Manager), FMID HCMA710	9
-----------------------------------	---

Modifications entre la version 6.0.1 et la version 7.0	10
--	----

IBM WebSphere Developer for System z, FMID	
--	--

HHOP700	10
---------	----

Gestionnaire IBM CARMA (Common Access	
---------------------------------------	--

Repository Manager), FMID HCMA700	11
-----------------------------------	----

Sauvegarde des fichiers déjà configurés	11
---	----

## Chapitre 3. Activation des composants du système MVS de Developer for System z . . . . . 17

Définition de MAXASSIZE dans	
------------------------------	--

SYS1.PARMLIB(BPXPRMxx)	17
------------------------	----

L'APF autorise hlq.SFEKLOAD	17
-----------------------------	----

Personnalisation de FEJCNFG, le fichier de configuration du moniteur de travaux JES	18
---	----

Personnalisation du Langage JCL d'initialisation du	
---	--

moniteur de travaux JES	20
-------------------------	----

Fonction de trace du moniteur de travaux JES	21
--	----

Exécution du moniteur de travaux JES comme un	
---	--

programme STC	21
---------------	----

Autorisations pour les serveurs	22
---------------------------------	----

Vérification du langage JCL d'initialisation du moniteur de travaux JES	23
---	----

Sécurité et accès au spoule JES	23
---------------------------------	----

Accès conditionnel au spoule	23
------------------------------	----

Commandes disponibles	23
-----------------------	----

Accès limité aux fichiers spoule	24
----------------------------------	----

Personnalisation ELAXF*, procédure de construction à distance	24
---	----

(Facultatif) Définition d'une transaction APPC pour	
---	--

le service Commandes TSO	26
--------------------------	----

Préparation	27
-------------	----

Implémentation	29
----------------	----

(Facultatif) Personnalisation ELAXM*, membres de procédure mémorisée DB2	29
--	----

(Facultatif) Personnalisation de la prise en charge de la langue bidirectionnelle CICS (bidi)	30
---	----

(Facultatif) Personnalisation du gestionnaire de déploiement ADM	32
--	----

Référentiel de CRD	33
--------------------	----

Région de connexion CICS primaire	33
-----------------------------------	----

Gestionnaire de message de pipeline	34
-------------------------------------	----

(Facultatif) Régions de connexion CICS non primaires	34
--	----

## Chapitre 4. Activation des composants de Developer for System z z/OS UNIX . 37

Enregistrement du fichier de configuration	
--	--

rsed.envvars dans un autre répertoire	38
---------------------------------------	----

Personnalisation de rsed.envvars, le fichier de configuration de l'Explorateur de systèmes éloignés	
---	--

RSE	39
-----	----

(Facultatif) Définition de la gamme de ports	
--	--

disponible pour RSE	45
---------------------	----

(Facultatif) Définition des paramètres de démarrage Java supplémentaires avec	
---	--

_RSE_*OPTS	46
------------	----

Configuration du démon INETD et de RSE	
--	--

REXEC/SSH	48
-----------	----

Configuration du démon RSE INETD	48
----------------------------------	----

Configuration de INETD REXEC (ou SSH)	50
---------------------------------------	----

Personnalisation d'ISPF.conf, fichier de configuration d'ISPF	51
---	----

Vérification de la configuration du serveur RSE	51
---	----

Disponibilité des ports	53
-------------------------	----

Connexion REXEC	53
-----------------	----

Script de shell REXEC/SSH	54
---------------------------	----

Connexion du démon RSE	55
------------------------	----

Connexion du moniteur de travaux JES	55
--------------------------------------	----

Connexion du service Commandes TSO (avec SCLMDT)	55
--	----

Connexion du service Commandes TSO (avec APPC)	57
--	----

(Facultatif) Personnalisation de ssl.properties, configuration de la couche SSL du RSE	57
--	----

(Facultatif) Personnalisation de rsecomm.properties, configuration de la fonction de trace RSE . . . . .	58
(Facultatif) Personnalisation de projectcfg.properties, configuration de projets hôte . . . . .	59
(Facultatif) Personnalisation de FMIEXT.properties, intégration de File Manager . . . . .	60

## **Chapitre 5. (Facultatif) Activation d'IBM Common Access Repository Manager (CARMA) . . . . . 63**

Personnalisation des composants du gestionnaire CARMA pour MVS. . . . .	63
Personnalisation des composants du gestionnaire CARMA pour z/OS UNIX . . . . .	65
(Facultatif) Activation des modèles de RAM (Repository Access Managers) . . . . .	66
Activation de la RAM SCLM . . . . .	67
Activation de la RAM PDS . . . . .	67
Activation de la RAM du squelette . . . . .	67

## **Chapitre 6. (Facultatif) Activation d'IBM Software Configuration and Library Manager (SCLM) Developer Toolkit . . . 69**

## **Chapitre 7. Remarques relatives au client Developer for System z . . . . . 71**

## **Chapitre 8. Remarques relatives aux performances . . . . . 73**

Utilisation du système de fichiers zFS . . . . .	73
Eviter l'emploi de STEPLIB . . . . .	73
Amélioration de l'accès aux bibliothèques du système. . . . .	74
Bibliothèques d'exécution Language Environment (LE) . . . . .	74
Développement d'application . . . . .	75
Amélioration des performances du contrôle d'autorisations d'accès. . . . .	75
Partage de classes entre JVM . . . . .	75
Activation du partage de classes . . . . .	76
Limites de la taille de la mémoire cache . . . . .	76
Sécurité de la mémoire cache . . . . .	76
SYS1.PARMLIB(BPXPRMxx) . . . . .	76
Espace disque . . . . .	77
Utilitaires de gestion de cache . . . . .	77
Taille de pile Java fixe . . . . .	78
Gestion de la charge de travail . . . . .	78

## **Annexe A. Exécution de plusieurs instances de Developer for System z. . 81**

Niveaux de logiciels identiques, fichiers de configuration différents . . . . .	81
Toutes autres situations . . . . .	82

## **Annexe B. Identification des incidents liés à la configuration . . . . . 85**

Emplacement des fichiers journaux . . . . .	85
Journalisation du moniteur de travaux JES . . . . .	86

Journalisation de la transaction APPC (service Commandes TSO) . . . . .	86
Journalisation du serveur RSE . . . . .	86
Consignation des tests du programme IVP fekfivpc . . . . .	87
Journal d'intégration de Fault Analyzer . . . . .	87
Journal d'intégration de File Manager . . . . .	87
Journal du gestionnaire CARMA . . . . .	88
Fichiers de vidage . . . . .	88
Fichiers de vidage MVS . . . . .	88
Fichiers de vidage Java . . . . .	89
Emplacements de vidage de z/OS UNIX . . . . .	90
Autorisation de contrôle par un programme pour les programmes RSE . . . . .	90
Ports TCP/IP réservés. . . . .	91
Taille d'espace adresse. . . . .	92
Exigences d'INETD. . . . .	92
Limitations définies dans SYS1.PARMLIB(BPXPRMxx). . . . .	93
Limitations stockées dans le profil de sécurité . . . . .	93
Limitations forcées par les sorties du système . . . . .	93
Traçage de suivi des erreurs. . . . .	93
Transaction APPC et service Commandes TSO. . . . .	94
Informations diverses . . . . .	96
Limites du système. . . . .	96
Incidents connus . . . . .	96
Emulateur de connexion à l'hôte . . . . .	97
Contacter le support IBM. . . . .	97

## **Annexe C. Configuration du TCP/IP 101**

Dépendance au nom d'hôte . . . . .	101
Présentation du programme de résolution. . . . .	101
Présentation des ordres de recherche d'informations de configuration . . . . .	102
Ordres de recherche dans l'environnement UNIX sous z/OS . . . . .	102
Fichiers de configuration du programme de résolution de base : . . . . .	103
Tables de conversion : . . . . .	103
Tables de système hôte local : . . . . .	104
Application à Developer for System z . . . . .	105

## **Annexe D. Configuration d'INETD. . . 109**

inetd.conf . . . . .	109
ETC.SERVICES . . . . .	110
Ordre de recherche dans l'environnement z/OS UNIX . . . . .	111
Ordre de recherche dans l'environnement MVS natif . . . . .	111
Définitions de ports PROFILE.TCPIP . . . . .	112
/etc/inetd.pid . . . . .	113
Démarrage . . . . .	113
/etc/rc . . . . .	113
/etc/inittab . . . . .	114
BPXBATCH . . . . .	114
Session de Shell . . . . .	115
Sécurité . . . . .	115
Exigences de Developer for System z . . . . .	116
INETD . . . . .	116
Démon RSE . . . . .	116

## **Annexe E. Configuration du protocole**

### **SSL . . . . . 117**

Clonage de la configuration RSE existante . . . . .	118
Détermination des fichiers de clés à employer . . . . .	118
Création d'un stock de clés avec keytool . . . . .	119
Création d'une base de données de clés (démon uniquement). . . . .	120
Création d'un jeu de clés avec RACF . . . . .	120
Création d'une base de données de clés avec gskkyman . . . . .	121
Activation du protocole SSL par la mise à jour de ssl.properties . . . . .	125
Test de la connexion . . . . .	126

### **Annexe F. Configuration APPC . . . . . 131**

Méthode d'accès VSAM . . . . .	131
VTAM. . . . .	132
SYS1.PARMLIB(APPCLPMxx) . . . . .	133
SYS1.PARMLIB(ASCHPMxx) . . . . .	134
Activation des modifications d'APPC . . . . .	135
Définition de la transaction du service Commandes TSO . . . . .	135

### **Glossaire . . . . . 137**

### **Remarques . . . . . 141**

Marques et logos . . . . .	143
----------------------------	-----





---

## Figures

1.	FEJCNFG, Fichier de configuration du moniteur de travaux JES . . . . .	18
2.	Langage JCL du moniteur de travaux JES . . . . .	21
3.	Langage REXX pour les panneaux ISPF d'APPC . . . . .	27
4.	rsed.envvars – Fichier de configuration RSE . . . . .	41
5.	ISPF.conf - Fichier de configuration ISPF . . . . .	51
6.	ssl.properties – Fichier de configuration RSE . . . . .	58
7.	rsecomm.properties – Fichier de configuration de consignment . . . . .	58
8.	projectcfg.properties – Fichier de configuration de projets basé sur l'hôte . . . . .	59
9.	FMIEXT.properties – Fichier de configuration de File Manager . . . . .	60
10.	CRASRV.properties – Fichier de configuration de CARMA . . . . .	65
11.	Langage JCL de démarrage d'INETD . . . . .	114
12.	Importation du certificat hôte . . . . .	127
13.	Préférences . . . . .	128
14.	Langage JCL pour créer des VSAM d'APPC . . . . .	132
15.	SYS1.SAMPLIB(ATBAPPL) . . . . .	133
16.	SYS1.PARMLIB(APPCPMxx) . . . . .	133
17.	SYS1.PARMLIB(ASCHPMxx) . . . . .	135



---

## Tableaux

1. Publications référencées . . . . .	xi	8. Liste de contrôle des qualificatifs de haut niveau ELAXF* . . . . .	25
2. Matrice d'installation et de configuration de Developer for System z . . . . .	1	9. Liste de contrôle de transaction APPC . . . . .	27
3. Présentation des membres MVS personnalisés . . . . .	11	10. Modèle ELAXM* - Membres de la procédure mémoire DB2 . . . . .	30
4. Présentation des fichiers z/OS UNIX personnalisés . . . . .	14	11. Fichiers de configuration facultatifs . . . . .	39
5. Personnalisation dans des bibliothèques autres que Developer for System z . . . . .	15	12. Liste de contrôle du client Developer for System z . . . . .	71
6. Commandes de console du moniteur de travaux JES . . . . .	23	13. Définitions locales disponibles pour le programme de résolution . . . . .	106
7. Modèles de procédure ELAXF* . . . . .	25		



---

## A propos de ce manuel

Le présent manuel décrit les fonctions d'IBM Rational Developer for System z. Il fournit les instructions permettant de configurer les serveurs IBM Rational Developer for System z sur votre système hôte z/OS.

A partir de maintenant, les noms suivants sont utilisés dans le présent ouvrage :

- *IBM Rational Developer for System z* est appelé *Developer for System z*
- *IBM Rational Developer for System z Common Access Repository Manager* est appelé *Common Access Repository Manager*, parfois abrégé par *CARMA*
- *IBM Software Configuration and Library Manager (SCLM) Developer Toolkit* est appelé *SCLM Developer Toolkit*, parfois abrégé par *SCLMDT*

**Remarque :** Les informations de configuration présentées dans cet ouvrage concernent IBM Rational Developer for System z Version 7.1.1.

Pour les éditions antérieures, telles que IBM WebSphere Developer for System z, IBM WebSphere Developer for zSeries et WebSphere Studio Enterprise Developer, reprenez les informations du guide de configuration de l'hôte et des répertoires de programme de ces éditions.

---

## Public concerné par ce manuel

Le présent manuel s'adresse aux programmeurs système qui souhaitent installer et configurer IBM Rational Developer for System z Version 7.1.1 sur leur système hôte z/OS. Avant d'utiliser le présent manuel, vous devez maîtriser les systèmes hôtes z/OS UNIX et MVS.

---

## Publications référencées

Les publications suivantes sont référencées dans ce document :

*Tableau 1. Publications référencées*

Titre de la publication	Référence de la commande	Référence	Lien de la référence
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>
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>

Tableau 1. Publications référencées (suite)

Titre de la publication	Référence de la commande	Référence	Lien de la référence
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 for System z Host Configuration Guide	SC23-7658	RD/z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/index.html">http://www-306.ibm.com/software/awdtools/rdz/library/index.html</a>
Rational Developer for System z Host Planning Guide	GI11-8296-00	RD/z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/index.html">http://www-306.ibm.com/software/awdtools/rdz/library/index.html</a>
SCLM Developer Toolkit Installation and Customization Guide	SC23-8504	RD/z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/index.html">http://www-306.ibm.com/software/awdtools/rdz/library/index.html</a>
ABCs of z/OS System Programming Volume 9	SG24-6989	Redbook	<a href="http://www.redbooks.ibm.com/">http://www.redbooks.ibm.com/</a>
APPC and WebSphere Developer for System z	SC23-5885	Livre blanc	<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		Livre blanc	<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	Livre blanc	<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>

Tableau 1. Publications référencées (suite)

Titre de la publication	Référence de la commande	Référence	Lien de la référence
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>





# Chapitre 1. Installation et configuration des composants de l'hôte

Pour chacune des fonctions suivantes, installez les FMID requis. Pour de plus amples informations sur les différents FMID, veuillez vous reporter au répertoire du programme correspondant pour le FMID que vous installez.

Tableau 2. Matrice d'installation et de configuration de Developer for System z

Si vous avez besoin de cette fonction d'IBM Rational Developer for System z	Vous devez installer les FMID suivants	Vous trouverez des informations d'installation et de configuration dans les guides suivants
<ul style="list-style-type: none"><li>• Connectivité hôte</li><li>• Connectivité au JES</li><li>• Compilation distante</li><li>• Renvoi d'erreur de la compilation distante</li><li>• Débogage à distance</li><li>• Procédures mémorisées DB2</li><li>• Prise en charge à l'écran des MFS IMS</li><li>• Prise en charge des mappes BMS CICS</li><li>• Prise en charge de la langue bidirectionnelle CICS (bidi)</li><li>• Gestionnaire de déploiement d'application (ADM)</li><li>• Intégration du gestionnaire de fichiers</li><li>• Intégration de l'analyseur d'erreur</li></ul>	HHOP710, HSD3310*	<ul style="list-style-type: none"><li>• <i>Program Directory for IBM Rational Developer for System z (GI11-8298-00)</i></li><li>• <i>Rational Developer for System z Guide de configuration de l'hôte (SC23-7658)</i></li><li>• <i>Rational Developer for System z - Guide de planification de l'hôte (SC11-2412-02)</i></li></ul> en option <ul style="list-style-type: none"><li>• <i>Program Directory for IBM Software Configuration and Library Manager (SCLM) Developer Toolkit (GI11-8306-00)</i></li><li>• <i>SCLM Developer Toolkit Installation and Customization Guide (SC23-8504)</i></li></ul>
<ul style="list-style-type: none"><li>• Gestionnaire CARMA (Common Software Configuration Management Access)</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 (GI11-8299-00)</i></li></ul> en option <ul style="list-style-type: none"><li>• <i>Program Directory for IBM Rational Developer for System z (GI11-8299-00)</i></li><li>• <i>Rational Developer for System z Guide de configuration de l'hôte (SC23-7658)</i></li><li>• <i>Rational Developer for System z - Guide de planification de l'hôte (SC11-2412-02)</i></li></ul>
<ul style="list-style-type: none"><li>• Outils de développement pour le gestionnaire SCLM (Software Configuration and Library Manager)</li></ul>	HSD3310	<ul style="list-style-type: none"><li>• <i>Program Directory for IBM Software Configuration and Library Manager (SCLM) Developer Toolkit (GI11-8306-00)</i></li><li>• <i>SCLM Developer Toolkit Installation and Customization Guide (SC23-8504)</i></li></ul>

(\*) Developer for System z nécessite une connexion au service Commandes TSO sur z/OS. Cette connexion est mise à disposition par l'un des éléments suivants :

1. l'installation et la configuration de SCLM Developer Toolkit, FMID HSD3310 (par défaut)
2. l'installation et la configuration d'une transaction APPC

(\*\*) CARMA nécessite une interface basée sur l'hôte. HHOP710 met cette fonction à disposition, ou bien vous pouvez utiliser une interface personnalisée, basée sur l'hôte, et éviter l'installation de HHOP710

---

## Remarques relatives à la pré-installation

- Si vous disposez de la version précédente d'IBM WebSphere Developer for System z, IBM WebSphere Developer for zSeries ou WebSphere Studio Enterprise Developer, il est recommandé d'enregistrer les fichiers personnalisés associés avant d'installer la mise à niveau pour IBM Rational Developer for System z Version 7.1.1. Pour connaître les fichiers qui nécessitent une personnalisation, consultez la section «Sauvegarde des fichiers déjà configurés», à la page 11.
- Voir aussi Annexe A, «Exécution de plusieurs instances de Developer for System z», à la page 81 si vous prévoyez d'exécuter plusieurs instances de Developer for System z.
- La version 7.1 de Developer for System z a mis à disposition deux méthodes d'accès au service Commandes TSO, qui exécutent les commandes TSO/ISPF, soit implicitement par le produit, soit explicitement par l'utilisateur.
  - Appel d'une fonction de SCLM Developer Toolkit. Cette méthode nécessite l'installation et la configuration de CLMDT, FMID HSD3310, qui est livré avec Developer for System z. Il s'agit de la méthode utilisée par défaut dans les exemples de membres.

Les étapes de personnalisation de SCLMDT suivantes doivent être appliquées. Elles sont décrites dans *SCLM Developer Toolkit Installation and Customization Guide (SC23-8504)* :

- Création de la structure de répertoires et des fichiers de configuration
- Personnalisation d'ISPF.conf

**Remarque :** Ni le produit SCLM, ni le client basé sur Eclipse ne sont obligatoires pour la prise en charge de cette fonction de SCLMDT.

- Transaction APPC (comme dans les éditions précédant la version 7.1). Cette méthode nécessite la mise en place et la configuration d'APPC, qui fait partie de z/OS.
- Le module de chargement bidirectionnel a été modifié dans la version 7.1 (nom et codage) ; il nécessite le retrait des anciens modules et la configuration des nouveaux.
- Le code bidi qui a été créé dans les précédentes versions (avant la version 7.0) doit être RECOMPILE pour utiliser le nouveau module FEJBDTRX.

Pour obtenir des instructions détaillées sur l'installation de SMP/E pour chaque FMID (identificateur de modification de fonction), voir le tableau 2, à la page 1.

---

## Préparation de la configuration

Developer for System z a une liste de logiciels prérequis qui doivent être installés et opérationnels pour que le produit fonctionne. Il y a également une liste de logiciels corequis pour la prise en charge de fonctions spécifiques de Developer for System z. Ces éléments requis doivent être installés et opérationnels au moment de l'exécution pour que les fonctions correspondantes opèrent selon leur conception.

Consultez la liste des éléments prérequis et des éléments corequis pour votre version de Developer for System z dans *Rational Developer for System z - Guide de planification de l'hôte (SC11-2412-02)*.

**Avertissement :** La version 64 bits Java n'est pas prise en charge.

### Configuration nécessaire pour les logiciels et produits requis

Vérifiez auprès de votre programmeur système MVS, de votre administrateur sécurité et de votre administrateur TCP/IP que les logiciels et produits requis sont installés, testés et qu'ils fonctionnent.

- La version 7.1 de Developer for System z a mis à disposition deux méthodes d'exécution des commandes TSO/ISPF, soit implicitement par le produit, soit explicitement par l'utilisateur.
  - Appel d'une fonction de SCLM Developer Toolkit. Cette méthode nécessite l'installation et la configuration de CLMDT, FMID HSD3310, qui est livré avec Developer for System z. Il s'agit de la méthode utilisée par défaut dans les exemples de membres.

Les étapes de personnalisation de SCLMDT suivantes doivent être appliquées. Elles sont décrites dans *SCLM Developer Toolkit Installation and Customization Guide (SC23-8504)* :

- Création de la structure de répertoires et des fichiers de configuration
- Personnalisation d'ISPF.conf

**Remarque :** Ni le produit SCLM, ni le client basé sur Eclipse ne sont obligatoires pour la prise en charge de cette fonction de SCLMDT.

- Transaction APPC (comme dans les éditions précédant la version 7.1). Cette méthode nécessite la mise en place et la configuration d'APPC, qui fait partie de z/OS.
- La bibliothèque de classes DLL C/C++ CBC.SCLBDLL et les bibliothèques d'exécution Language Environment (LE) CEE.SCEERUN et CEE.SCEERUN2 doivent se trouver dans LINKLIST.
- Tous les utilisateurs de z/OS UNIX doivent disposer de droits d'accès READ et EXECUTE aux répertoires Java.
- Developer for System z dépend d'INETD pour la configuration de la connexion client-hôte. Pour plus d'informations sur REXEC et SSH, voir l'Annexe D, «Configuration d'INETD», à la page 109.
- Les actions à distance (basées sur l'hôte) pour les sous-projets z/OS UNIX nécessitent que REXEC ou SSH soit actif sur l'hôte.
- Developer for System z requiert que le protocole TCP/IP dispose du nom d'hôte correct lors de son initialisation. Cela implique que les différents fichiers de configuration du TCP/IP et du programme de résolution Resolver soient configurés correctement. Pour plus d'informations sur la personnalisation de TCP/IP et du programme de résolution, voir l'Annexe C, «Configuration du

TCP/IP», à la page 101 et la section "TCPIP.DATA configuration statements" dans le manuel *Communications Server IP Configuration Reference* (SC31-8776).

Vous pouvez tester votre configuration TCP/IP à l'aide de la commande TSO **HOMETEST**. Pour plus d'informations sur cette commande, consultez la section "TSO Commands" dans *Communications Server IP System Administrator's Commands* (SC31-8781).

Exemple de résultat de la commande **HOMETEST** :

Exécution du programme de  
test de la configuration TCP/IP IBM MVS TCP/IP CS V1R7

Le fichier de paramètre de configuration FTP utilisé est le  
"SYS1.TCPPARMS(FTPDATA)"

Le nom d'hôte TCP est : CDFMVS08

Utilisation du serveur de noms pour une résolution CDFMVS08  
Les adresses IP suivantes correspondent au nom d'hôte TCP : CDFMVS08  
9.42.112.75

Les adresses IP suivantes correspondent aux adresses IP HOME définies dans  
PROFILE.TCPIP :  
9.42.112.75  
127.0.0.1

Toutes les adresses IP pour CDFMVS08 se trouvent dans la liste HOME !

Commande Hometest réussie - Tous les tests sont réussis !

## Remarques relatives à l'ID utilisateur

L'ID d'un utilisateur de Developer for System z doit contenir (au moins) les attributs suivants :

- Un accès TSO (avec une taille de région normale).
- Un segment OMVS défini sur le système de sécurité (RACF, par exemple), à la fois pour l'ID utilisateur et pour son groupe par défaut. Ce segment OMVS peut être un segment par défaut individuel ou étendu à tout le système. Pour plus d'informations sur le segment OMVS par défaut, voir *Security Server RACF Security Administrator's Guide* (SA22-7683).
  - Un répertoire de base (avec des droits d'accès WRITE, READ et EXECUTE) attribué à l'utilisateur, et identifié dans le segment OMVS.
  - Le champ PROGRAM du segment OMVS doit être /bin/sh ou un autre shell z/OS UNIX valide comme /bin/tcsh.
  - Le champ ASSIZEMAX ne doit pas être rempli, de sorte que les valeurs par défaut du système seront utilisées.
  - L'ID utilisateur ne nécessite pas d'UID 0.

Exemple (commande **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
```

- Le groupe par défaut de l’ID utilisateur nécessite un GID.  
Exemple (commande **LISTGRP group NORACF OMVS**):  
GROUP group

```
OMVS INFORMATION
-----
GID= 0000003243
```

- Des droits d’accès READ et EXECUTE aux répertoires d’installation et de configuration de Developer for System z, qui est /usr/lpp/wd4z/\* par défaut.

## Remarques relatives au serveur

Les services hôte suivants, et donc leurs ports, doivent être disponibles pour que client puisse s’y connecter. Tous les autres ports utilisés par Developer for System z présentent un trafic réservé à l’hôte. Cela signifie que, pour Developer for System z, seuls les ports indiqués sur la liste doivent être définis pour votre pare-feu qui protège l’hôte.

- Service INETD pour démarrer le serveur RSE. Developer for System z prend en charge différentes façons de faire cela :
  - Démon RSE, port 4035 par défaut
  - REXEC, port 512 par défaut
  - SSH, port 22 par défaut
- Serveur RSE pour la communication client-hôte. Utilise tout port disponible par défaut, mais peut être limité à une gamme spécifique. La communication sur ce port peut être chiffrée à l’aide de la fonction SSL.
- Service Telnet TN3270 pour Host Connect Emulator, port 23 par défaut. La communication sur ce port peut être chiffrée à l’aide de la fonction SSL (port 992 par défaut). Le port par défaut affecté au service Telnet TN3270 dépend de l’utilisation ou non du chiffrement par l’utilisateur.

**Remarque :** Les anciens clients (version 7.0 et plus ancienne) communiquent directement avec le moniteur de travaux JES, port 6715 par défaut.

**Remarque :** IBM Debug Tool for z/OS est appelé lors d’une session de débogage à distance pour Cobol. Ce produit communique directement avec le client. Cette communication est initiée sur l’hôte, et se connecte au port 8001 du client.

INETD est un processus serveur z/OS UNIX qui est nécessaire pour la configuration des connexions client-hôte de Developer for System z. Les paramètres environnementaux d’INETD, qui sont transmis lors du démarrage du processus, ainsi que les permissions pour l’ID utilisateur d’INETD doivent être configurés convenablement pour qu’INETD puisse démarrer le serveur RSE.

- Si INETD est lancé par JCL à l’aide de BPXBATCH, la taille de la région doit être 0.
- Si INETD est démarré à partir d’une session TSO/OMVS, la taille de la région TSO doit être 2096128 ou plus.
- Des droits d’accès READ et EXECUTE aux répertoires d’installation de Developer for System z, qui est /usr/lpp/wd4z/\* par défaut.
- BPX.DAEMON et si possible le droit UID 0.

Pour plus d’informations sur les droits INETD, voir Annexe D, «Configuration d’INETD», à la page 109.

L'Explorateur de systèmes éloignés (RSE) est le composant de Developer for System z qui fournit les services de base comme la connexion du client à l'hôte.

- Chaque connexion client-hôte a un serveur RSE privé.
- En plus des ports présentés dans ce manuel pour la communication avec d'autres serveurs Developer for System z, le serveur RSE nécessite plusieurs ports pour son usage interne.

## Droits requis pour l'implémentation des tâches de configuration

La configuration de Developer for System z nécessite plus de droits que les droits classiques du programmeur système, par conséquent, un minimum d'entraide est à prévoir. La liste ci-dessous souligne les points les plus importants :

- Mise à jour des droits pour les fichiers système (comme SYS1.PARMLIB)
- Droits d'exécution de commandes de la console
- Droits de création de répertoires z/OS UNIX
- Droits de mise à jour du répertoire z/OS UNIX /etc
- Droits de définition du bit de contrôle de programmes z/OS UNIX
- Droits de définition de nouveaux ports TCP/IP
- Droits BPX.DAEMON et si possible UID 0 pour (re)démarrer INETD
- Droits de gestion WLM/RSM
- Droits de gestion APPC pour la création d'un programme transactionnel
- Droits de l'administrateur sécurité pour la requête et la création de profils de sécurité
- (Facultatif) Droits de l'administrateur DB2 pour la création d'une procédure mémorisée
- (Facultatif) Droits de l'administrateur CICS pour la définition de programmes sous CICS

---

## Remarques relatives au prédéploiement

Developer for System z et Common Access Repository Manager (CARMA) prennent en charge le clonage d'une installation sur un système différent, ce qui épargne le besoin d'une installation SMP/E sur chaque système.

Les fichiers et répertoires suivants sont obligatoires pour le déploiement sur d'autres systèmes. Si vous avez copié un fichier à un emplacement différent pour vos besoins de personnalisations, ce fichier doit remplacer son équivalent dans la liste ci-après.

**Remarque :** La liste ci-après ne couvre pas les besoins relatifs au déploiement des logiciels prérequis et corequis.

## IBM Rational Developer for System z, FMID HHOP710

- hlq.SFEKLOAD(\*)
- hlq.SFEKPROC(\*)
- hlq.SFEKSAMP(FEJJCNFG)
- hlq.SFEKSAMP(FEJJJCL)
- hlq.SFEKSAMP(ELAXF\*)
- /usr/lpp/wd4z/\*

- composants facultatifs
  - hlq.SFEKSAMP(FEJTSO) - soumission TSO
  - hlq.SFEKSAMP(ELAXM\*) - procédure stockée dans DB2
  - définitions CICS créées par hlq.SFEKSAMP(ADNPCCSD) - connexion ADM primaire
  - définitions CICS créées par hlq.SFEKSAMP(ADNARCSO) - connexions ADM non primaire
  - méthode d'accès VSAM créée par hlq.SFEKSAMP(ADNVSAM) - serveur ADM CRD
  - module de chargement créé par hlq.SFEKSAMP(ADNCMSGH) - ADM Pipeline Handler
  - transaction APPC créée par hlq.SFEKSAMP(FEKAPPCC) - service Commandes TSO basé sur APPC
  - /etc/SCLMDT/CONFIG/ISPF.conf - ajout du serveur Commandes TSO à l'environnement ISPF de SCLM Developer Toolkit

**Remarque :** hlq et /usr/lpp/wd4z désignent le qualificatif de haut niveau (FEK par défaut) et le chemin d'accès utilisés lors de l'installation du produit.

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

- hlq.SCRALOAD(\*)
- hlq.SCRACLIST(CRASUBMT)
- VSAM créés par les membres hlq.SCRASAMP ci-après
  - hlq.SCRASAMP(CRA\$VMSG) message VSAM
  - hlq.SCRASAMP(CRA\$VDEF) configuration VSAM
  - hlq.SCRASAMP(CRA\$SVSTR) information personnalisée VSAM

**Remarque :** hlq désigne le qualificatif de haut niveau (CRA par défaut) utilisé au cours de l'installation du produit.





---

## Chapitre 2. Modifications de l'installation et de la configuration

La présente section met en évidence les modifications de l'installation et de la configuration par rapport aux précédentes éditions du produit. Pour de plus amples informations, consultez les sections liées dans ce manuel.

---

### Modifications entre la version 7.0 et la version 7.1

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

- **Ajout** : choix de configuration - commandes TSO/ISPF par l'intermédiaire d'une transaction APPC ou par l'intermédiaire de SCLM Developer Toolkit
- **Modification** : la transaction APPC exploite une nouvelle fonction ISPF
- **Ajout** : les membres personnalisables suivants sont nouveaux
  - samplib ELAXFADT
  - samplib ADNCMSGH
  - /usr/lpp/wd4z/rse/lib/FMIEXT.properties
- **Modification** : les membres suivants ont été déplacés
  - SFEKDLL(FEJBDTRX) -> SFEKLOAD(FEJBDTRX)
- **Modification** : les membres personnalisables suivants ont été modifiés
  - samplib FEKFAPPC
  - /usr/lpp/wd4z/rse/lib/rsed.envvars
  - /usr/lpp/wd4z/rse/lib/setup.env.zseries
  - /usr/lpp/wd4z/rse/lib/server.zseries

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

- **Modification** : la journalisation est inscrite dans l'instruction de définition de données CARMALOG
- **Modification** : la méthode d'accès VSAM au message CARMA (CRAMSG) et la méthode d'accès VSAM à la configuration (CRADEF) ont été mises à jour
- **Ajout** : les membres personnalisables suivants sont nouveaux
  - samplib CRA#ECOB
  - samplib CRA#EPDS
  - samplib CRA#ERAM
  - samplib CRA#ESLM
- **Changement de nom** : les membres personnalisables suivants ont été renommés
  - 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
- **Modification** : les membres personnalisables suivants ont été modifiés
  - clist CRASUBMT

---

## Modifications entre la version 6.0.1 et la version 7.0

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

- **Modification** : IBM WebSphere Developer for System z, FMID HHOP700, associe maintenant les fonctions de cinq produits séparés en un seul produit afin de réduire les tâches de l'installation
  - v601 FMID H001600, IBM WebSphere Developer for zSeries RSE + ICU
  - v601 FMID H002600, IBM WebSphere Developer for zSeries moniteur de travaux JES
  - v601 FMID HEDS500, IBM WebSphere Developer for zSeries Options for z/OS (HHOP700 ne comprend pas toutes les fonctions)
  - v601 FMID HBDI601, IBM WebSphere Developer for zSeries prise en charge bidi
  - (nouvelle) version 700, IBM Application Deployment Manager
- **Modification** : les travaux d'installation sont renommés pour que les étapes à effectuer apparaissent dans l'ordre alphabétique
- **Modification** : l'installation de l'outil SMP/E définit désormais les bits de contrôle de programme dans z/OS UNIX
- **Modification** : le codage du serveur REXX a été déplacé de samplib vers un nouveau fichier, SFEKPROC
- **Modification** : BIDI utilise un nouveau module de chargement, toutes les références aux anciens modules doivent être supprimées
- **Modification** : le paramètre JESNAME n'est plus utilisé dans le fichier de configuration du moniteur de travaux JES (samplib FEJCNFG)
- **Modification** : les membres suivants n'ont plus besoin de personnalisation
  - /usr/lpp/wd4z/rse/lib/setup.env.zseries
- **Ajout** : les membres suivants sont nouveaux
  - samplib ELAXFCPC
  - samplib ELAXFCPP
  - samplib ELAXFMFS
  - /usr/lpp/wd4z/rse/lib/projectcfg.properties
- **Changement de nom** : les membres personnalisables suivants ont été renommés
  - samplib FEKFRDIS -> FEKAPPCL
  - samplib FEKFRDEL -> FEKAPPCX
  - samplib FEKFRTAD -> FEKAPPCC

- **Retrait** : les membres suivants ne sont plus intégrés
  - samplib ELAXFLNK
  - samplib FEJIVJCL
  - loadlib FEJBDTRN
  - loadlib FEJBDTRE

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

- **Modification** : les travaux d'installation sont renommés pour que les étapes à effectuer apparaissent dans l'ordre alphabétique
- **Modification** : les membres suivants n'ont plus besoin de personnalisation
  - /usr/lpp/wd4z/rse/lib/rsed.envvars
  - /usr/lpp/wd4z/rse/lib/rexxsub

---

## Sauvegarde des fichiers déjà configurés

Avant d'installer Developer for System z version 7.1.1, si vous utilisiez la version précédente de Developer for System z, il est recommandé d'enregistrer les fichiers personnalisés associés. Voir Annexe A, «Exécution de plusieurs instances de Developer for System z», à la page 81 avant de lancer la personnalisation si vous envisagez d'exécuter plusieurs instances de Developer for System z.

Les tableau 3 et tableau 4, à la page 14 présentent les fichiers qui ont probablement été personnalisés pour Developer for System z et CARMA version 6.0.1 (et supérieure). Le tableau 5, à la page 15 classe les différentes personnalisations qui sont effectuées sur les fichiers et sur les logiciels et produits prérequis et corequis, au cours de l'installation de Developer for System z et CARMA version 6.0.1 (et supérieure).

Tableau 3. Présentation des membres MVS personnalisés

Membre	Emplacement par défaut v 6.0.1	Emplacement par défaut v 7.0/v7.1	But
FEJJCNFG	hlq.SFEJSAMP (hlq = FEJ)	hlq.SFEKSAMP (hlq = FEK)	Fichier de configuration pour le moniteur de travaux JES
FEJJJCL	hlq.SFEJSAMP (hlq = FEJ)	hlq.SFEKSAMP (hlq = FEK)	Langage JCL pour le moniteur de travaux JES
FEJTSO	hlq.SFEJSAMP (hlq = FEJ)	hlq.SFEKSAMP (hlq = FEK)	Langage JCL pour les soumissions TSO
FEKAPPCC	N'existe pas	hlq.SFEKSAMP (hlq = FEK)	Le langage JCL doit créer une transaction APPC
FEKAPPCL	N'existe pas	hlq.SFEKSAMP (hlq = FEK)	Le langage JCL doit afficher une transaction APPC

Tableau 3. Présentation des membres MVS personnalisés (suite)

Membre	Emplacement par défaut v 6.0.1	Emplacement par défaut v 7.0/v7.1	But
FEKAPPCX	N'existe pas	hlq.SFEKSAMP (hlq = FEK)	Le langage JCL doit supprimer une transaction APPC
FEKFRTAD	hlq.SFEKSAMP (hlq = FEK)	(nouveau nom de membre FEKAPPCC)	Le langage JCL doit créer une transaction APPC
FEKFRDIS	hlq.SFEKSAMP (hlq = FEK)	(nouveau nom de membre FEKAPPCL)	Le langage JCL doit afficher une transaction APPC
FEKFRDEL	hlq.SFEKSAMP (hlq = FEK)	(nouveau nom de membre FEKAPPLX)	Le langage JCL doit supprimer une transaction APPC
ELAXF*	hlq.SCCUSAMP (hlq = CCU)	hlq.SFEKSAMP (hlq = FEK)	Langage JCL pour les constructions de projets distants, etc.
ELAXMSAM	hlq.SCCUSAMP (hlq = CCU)	hlq.SFEKSAMP (hlq = FEK)	Procédure JCL de l'espace adresse WLM du compilateur de procédures mémorisées PL/I et COBOL.
ELAXMJCL	hlq.SCCUSAMP (hlq = CCU)	hlq.SFEKSAMP (hlq = FEK)	Langage JCL définissant le compilateur de procédures mémorisées PL/I et COBOL pour DB2
ADNVSAM	N'existe pas	hlq.SFEKSAMP (hlq = FEK)	Langage JCL définissant le référentiel ADM CRD
ADNPCCSD	N'existe pas	hlq.SFEKSAMP (hlq = FEK)	Langage JCL activant le serveur CRD dans la région CICS primaire
ADNCMSGH	N'existe pas	hlq.SFEKSAMP (hlq = FEK) (n'existe pas dans la version 7.0)	Langage JCL compilant le gestionnaire de message de pipeline

Tableau 3. Présentation des membres MVS personnalisés (suite)

Membre	Emplacement par défaut v 6.0.1	Emplacement par défaut v 7.0/v7.1	But
ADNSMSGH	N'existe pas	hlq.SFEKSAMP (hlq = FEK)	Exemple de code source pour le gestionnaire de message de pipeline
ADNARCSD	N'existe pas	hlq.SFEKSAMP (hlq = FEK)	Langage JCL activant le serveur CRD dans les régions CICS non-primaires
CRASUBMT	hlq.SCRACLST (hlq = CRA)	hlq.SCRACLST (aide = CRA)	Liste de commandes CLIST pour la soumission du langage JCL à un serveur CARMA
CRAMREPR	hlq.SCRASAM (hlq = CRA)	hlq.SCRASAMP (hlq = CRA) (nouveau nom de membre dans la version 7.1 CRA\$VMSG)	Le langage JCL doit créer la méthode d'accès VSAM au message du gestionnaire CARMA
CRAREPR	hlq.SCRASAM (hlq = CRA)	hlq.SCRASAMP (hlq = CRA) (nouveau nom de membre dans la version 7.1 CRA\$VDEF)	Le langage JCL doit créer la méthode d'accès VSAM à la configuration du gestionnaire CARMA
CRASREPR	hlq.SCRASAM (hlq = CRA)	hlq.SCRASAMP (hlq = CRA) (nouveau nom de membre dans la version 7.1 CRA\$VSTR)	Le langage JCL doit créer la méthode d'accès VSAM aux informations personnalisées du gestionnaire CARMA
CRALREPR	hlq.SCRASAM (hlq = CRA)	hlq.SCRASAMP (hlq = CRA) (nouveau nom de membre dans la version 7.1 CRA#VSLM)	Le langage JCL doit créer la méthode d'accès VSAM au message de la RAM du SCLM

Tableau 3. Présentation des membres MVS personnalisés (suite)

Membre	Emplacement par défaut v 6.0.1	Emplacement par défaut v 7.0/v7.1	But
CRASALX	hlq.SCRASAM (hlq = CRA)	hlq.SCRASAMP (hlq = CRA) (nouveau nom de membre dans la version 7.1 CRA#ASLM)	Le langage JCL doit créer les fichiers de la RAM du SCLM
CRATREPR	hlq.SCRASAM (hlq = CRA)	hlq.SCRASAMP (hlq = CRA) (nouveau nom de membre dans la version 7.1 CRA#VPDS)	Le langage JCL doit créer la méthode d'accès VSAM au message de la RAM du fichier partitionné

**Remarque :** hlq désigne le qualificatif de haut niveau utilisé au cours de l'installation du produit. Les paramètres par défaut d'IBM pour le qualificatif hlq sont répertoriés, mais ils ne s'appliquent pas forcément à votre site.

Tableau 4. Présentation des fichiers z/OS UNIX personnalisés

Fichier	Emplacement par défaut v 6.0.1	Emplacement par défaut version 7.0/version 7.1	But
rsed.envvars	/usr/lpp/wd4z/rse/lib/	/usr/lpp/wd4z/rse/lib/	Fichier de configuration RSE
rsecomm.properties	/usr/lpp/wd4z/rse/lib/	/usr/lpp/wd4z/rse/lib/	Fichier de configuration de trace RSE
ssl.properties	/usr/lpp/wd4z/rse/lib/	/usr/lpp/wd4z/rse/lib/	Fichier de configuration SSL RSE
setup.env.zseries	/usr/lpp/wd4z/rse/lib/	(n'est plus personnalisé)	Exportation de variables d'environnement RSE
projectcfg.properties	(N'existe pas)	/usr/lpp/wd4z/rse/lib/	Fichier de configuration de projets hôte
FMIEXT.properties	(N'existe pas)	/usr/lpp/wd4z/rse/lib/ (n'existe pas dans la version 7.0)	Fichier de configuration du gestionnaire de fichiers
CRASRV.properties	/usr/lpp/wd4z/rse/lib/	/usr/lpp/wd4z/rse/lib/	Fichier de configuration du gestionnaire CARMA
rexsub	/usr/lpp/wd4z/rse/lib/	(n'est plus personnalisé)	Le langage z/OS UNIX REXX du gestionnaire CARMA doit exécuter la liste de commandes CRASUBMT CLIST du système MVS

**Remarque :** /usr/lpp/rd4z désigne le chemin d'accès utilisé au cours de l'installation des produits. Cette valeur par défaut d'IBM est présentée mais elle ne s'applique pas forcément à votre site.

*Tableau 5. Personnalisation dans des bibliothèques autres que Developer for System z*

Membre/ Fichier	Emplacement par défaut	Personnalisation obligatoire
BPXPRMxx	SYS1.PARMLIB	Définissez MAXASSIZE sur 2147483647 ou plus
PROGxx	SYS1.PARMLIB	L'APF autorise hlq.SFEKLOAD
ASCHPMxx	SYS1.PARMLIB	Définissez une classe de transaction APPC du service Commandes TSO
services	/etc/	Définissez le démon RSE
inetd.conf	/etc/	Définissez le démon RSE
ISPF.conf	/etc/SCLMDT/ CONFIG/	Définissez l'emplacement du serveur de commandes TSO
/	APPC	Définissez une transaction APPC du service Commandes TSO
/	WLM	Associez un programme transactionnel APPC à un groupe de performances TSO
/	WLM	Attribuez un environnement d'application à la procédure mémorisée DB2





---

## Chapitre 3. Activation des composants du système MVS de Developer for System z

Avant d'installer la version 7.1.1, si vous utilisiez la version précédente de Developer for System z, il est recommandé d'enregistrer la personnalisation associée comme décrit à la section «Sauvegarde des fichiers déjà configurés», à la page 11.

Toutes les références à hlq que vous trouverez dans cette section se rapportent au qualificatif de haut niveau utilisé au cours de l'installation de Developer for System z. L'installation par défaut est FEK, mais peut ne pas s'appliquer à votre site.

**Remarque :** La bibliothèque de classes DLL C/C++ CBC.SCLBDLL et les bibliothèques d'exécution Language Environment (LE) CEE.SCEERUN et CEE.SCEERUN2 doivent se trouver dans LINKLIST.

Les conditions de LINKLIST peuvent être ignorées en ajoutant une instruction STEPLIB à rsed.envvars, voir «Personnalisation de rsed.envvars, le fichier de configuration de l'Explorateur de systèmes éloignés RSE», à la page 39.

---

### Définition de MAXASSIZE dans SYS1.PARMLIB(BPXPRMxx)

MAXASSIZE définit la taille de la région de l'espace adresse (processus). Définissez MAXASSIZE dans SYS1.PARMLIB(BPXPRMxx) sur 2147483647 ou plus.

Cette valeur est vérifiable et définissable de manière dynamique (avant la procédure de chargement initial suivante) au moyen des commandes de la console suivantes, décrites dans le manuel *MVS System Commands (SA22-7627)*.

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

Vous trouverez de plus amples informations sur les autres emplacements au niveau desquels les tailles d'espace adresse peuvent être définies ou limitées dans la section «Taille d'espace adresse», à la page 92.

---

### L'APF autorise hlq.SFEKLOAD

Pour que le moniteur de travaux JES puisse accéder aux fichiers spoules JES, l'APF doit autoriser le module FEJJMON de la bibliothèque de chargement hlq.SFEKLOAD.

Les autorisations de l'APF sont définies dans SYS1.PARMLIB(PROGxx), si votre site se conforme aux recommandations IBM. Pour plus d'informations sur la définition des autorisations APF, consultez le document *MVS Initialization and Tuning Reference (SA22-7592)*.

A des fins de tests, les autorisations APF peuvent également être octroyées de manière dynamique. Elles seront en vigueur jusqu'à la procédure de chargement initial suivante du système. La commande requise de la console se présente comme suit :

```
SETPROG APF,ADD,DSN=h1q.SFEKLOAD,SMS
```

Pour plus d'informations sur la commande **SETPROG**, consultez le manuel *MVS System Commands* (SA22-7627).

---

## Personnalisation de FEJJCNFG, le fichier de configuration du moniteur de travaux JES

Le moniteur de travaux JES est le composant de Developer for System z qui gère l'ensemble de la connectivité JES. Un fichier de configuration permet de personnaliser certains aspects afin de répondre aux normes de votre site.

**Remarque :** Il est recommandé de copier le modèle de fichier de configuration dans un nouveau fichier et de personnaliser cette copie afin d'éviter de l'écraser lors d'une opération de maintenance.

Le modèle de fichier de configuration se trouve dans :

```
h1q.SFEKSAMP(FEJJCNFG)
```

Le fichier comporte un ensemble de définitions de variables d'environnement. Les lignes mises en commentaire commencent par un dièse (#). Le modèle de fichier de configuration est présenté dans la figure 1.

```
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=<tcpip stack name>
```

Figure 1. FEJJCNFG, Fichier de configuration du moniteur de travaux JES

Les définitions suivantes sont requises :

### **SERV\_PORT**

Le numéro de port du serveur hôte du moniteur de travaux JES. Le port par défaut est 6715. Vous pouvez le modifier mais le client et le serveur doivent TOUS DEUX être configurés avec le même numéro de port. Si vous modifiez le numéro de port du serveur, tous les utilisateurs du client de Developer for System z doivent modifier le port du moniteur de travaux JES de ce travaux JES de ce système dans la vue Systèmes distants.

**Remarque :** Avant de sélectionner un port, vérifiez qu'il est disponible sur votre système à l'aide des commandes TSO **NETSTAT** et **NETSTAT PORTL**. Pour plus d'informations, voir «Ports TCP/IP réservés», à la page 91.

**Remarque :** Lors de l'utilisation d'un client dont la version est 7.1 ou plus, toutes les communications sur ce port ne concernent que la machine hôte.

#### **CODEPAGE**

Page de code du poste de travail. La valeur par défaut est UTF-8. La page de code du poste de travail est définie sur UTF-8 et ne doit généralement pas être modifiée. Vous devrez peut-être modifier la page de code UTF-8 pour qu'elle corresponde à la page de code du poste de travail si des caractères NLS, tels que le symbole monétaire, posent problème.

#### **HOST\_CODEPAGE**

Page de code hôte. La valeur par défaut est IBM-1047. Modifiez si nécessaire.

#### **TZ**

Sélecteur de fuseau horaire. La valeur par défaut est EST5EDT. Le fuseau horaire par défaut est le temps universel coordonné + 5 heures (heure d'été de la côte est). Modifiez cette valeur pour afficher votre fuseau horaire. Pour plus d'informations, reportez-vous au manuel *UNIX System Services File System Interface Reference (SA22-7808)*.

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

Longueur de file d'attente d'écoute TCP/IP. La valeur par défaut est 5. Ne pas la modifier sauf recommandation explicite du point service IBM .

#### **MAX\_DATASETS**

La valeur par défaut est 32. Il s'agit du nombre maximal de fichiers de sortie en spoule renvoyés au client par le moniteur de travaux JES (par exemple SYSOUT, SYSPRINT, SYS00001, etc.).

#### **MAX\_THREADS**

La valeur par défaut est 200. Nombre maximal d'utilisateurs qui peuvent utiliser simultanément un moniteur de travaux JES. Si vous augmentez cette valeur, vous devez augmenter la taille de l'espace adresse du moniteur de travaux JES.

#### **TIMEOUT\_INTERVAL**

La valeur par défaut est 1200. Contrôle la fréquence à laquelle le serveur vérifie et arrête les unités d'exécution dont le délai est dépassé (voir **TIMEOUT**). La valeur du paramètre **TIMEOUT\_INTERVAL** indique le nombre de secondes entre chaque vérification.

#### **TIMEOUT**

La valeur par défaut est 3600. Durée, en secondes, avant l'arrêt d'une unité d'exécution, dû à l'absence d'interaction avec le client. La valeur maximale est 2147483647.

#### **AUTHMETHOD**

La valeur par défaut est RACF, ce qui signifie que l'interface de sécurité standard est utilisée. Cette valeur ne doit pas être modifiée, même si vous utilisez un produit autre que RACF.

Les définitions suivantes sont facultatives. Si vous les omettez, les valeurs par défaut seront utilisées comme indiqué ci-après :

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

Définissez la sortie à visualiser par l'utilisateur. La valeur par défaut (**LIMIT\_VIEW=NOLIMIT**) permet à l'utilisateur d'afficher toutes les sorties du moniteur de travaux JES. La spécification **USERID** limite la vue à la sortie dont il est propriétaire.

**Remarque :** Les seuls paramètres valides sont USERID et NOLIMIT.

**\_BPXK\_SETIBMOPT\_TRANSPORT=<nom de pile tcpip>**

Indique la pile TCPIP à utiliser en cas de plusieurs piles actives. La valeur par défaut est TCPIP. <nom de pile tcpip> représente le nom de pile TCPIP demandé, tel que défini dans l'instruction TCPIPJOBNAME du fichier TCPIP.DATA lié.

**Remarque :** Le codage d'une instruction SYSTCPD DD dans le langage de contrôle des travaux ne définit pas l'affinité de pile demandée.

**CONCHAR**

Indique le caractère de commande de la console du moniteur de travaux JES. CONCHAR a pour valeur par défaut CONCHAR=\$ pour JES2, ou CONCHAR=\* pour JES3. Si votre installation comporte un caractère de commande différent, définissez ce dernier par la valeur de CONCHAR.

**SUBMITMETHOD=TSO**

Soumission via TSO. La valeur par défaut (SUBMITMETHOD=JES) permet de soumettre des travaux directement au moniteur JES. La spécification SUBMITMETHOD=TSO entraîne une soumission des travaux via la commande **SUBMIT** de TSO. Cette méthode permet d'appeler des sorties TSO ; toutefois, elle présente des inconvénients en termes de performances et n'est pas recommandée pour cette raison.

**Remarque :** Les seuls paramètres valides sont TSO et JES.

**Remarque :** Si vous spécifiez SUBMITMETHOD=TSO, vous devez également définir TSO\_TEMPLATE.

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

Encapsuleur JCL pour la soumission de travaux via TSO. Il n'y a pas de valeur par défaut. Cette instruction indique le nom de membre complet du JCL à utiliser comme encapsuleur pour la soumission TSO. Pour plus d'informations, voir l'instruction SUBMITMETHOD.

**Remarque :** Un modèle de travail d'encapsuleur est fourni dans hlq.SFEKSAMP(FEJTSO). Consultez ce membre pour plus d'informations sur la personnalisation nécessaire.

**Remarque :** TSO\_TEMPLATE n'a aucun effet si SUBMITMETHOD=TSO n'est pas également spécifié.

**Remarque :** La définition JESNAME n'est plus obligatoire. A partir de la version 7.0, le moniteur de travaux JES détecte automatiquement si votre JES primaire est JES2 ou JES3.

---

## Personnalisation du Langage JCL d'initialisation du moniteur de travaux JES

Le moniteur de travaux JES est le composant de Developer for System z qui gère l'ensemble de la connectivité JES. Pour ce faire, un serveur doit être actif sur le système (que ce soit en travail utilisateur ou en programme STC). Suivez la procédure de personnalisation du langage JCL décrite dans hlq.SFEKSAMP(FEJJJCL) pour créer le serveur de moniteur de travaux JES en fonction des normes de votre site.

**Remarque :** Il est recommandé de copier le modèle de langage JCL dans un nouveau fichier et de personnaliser la copie afin d'éviter de l'écraser lors d'une opération de maintenance.

## Fonction de trace du moniteur de travaux JES

Si vous devez lancer la fonction de trace du moniteur de travaux JES pour un diagnostic, ajoutez "-TV" à la ligne PARM. La fonction de trace réduit les performances et ne doit être effectuée que sur indication du centre de support IBM.

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

Le traçage est également contrôlable par les commandes de la console. Supposons que le nom de travail utilisé soit JMON, la première commande de la console répertoriée ci-dessous active le traçage et la seconde la désactive.

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

## Exécution du moniteur de travaux JES comme un programme STC

Le moniteur de travaux JES s'exécute comme un travail utilisateur ou une tâche lancée (STC). Les tâches suivantes doivent être implémentées pour définir JMON comme un programme STC :

1. Le langage JCL n'est pas nécessaire pour avoir une carte JOB (il est même préférable de ne pas en avoir). La plupart des programmes STC démarrent avec une carte PROC comme dans l'exemple de la figure 2.

```
//JMON      PROC HLQ=FEK,  
//          PRM=  
//*  
//* RD/Z JES JOB MONITOR  
//*  
//JMONITOR EXEC PGM=FEJJMON,TIME=1440,REGION=0M,  
//          PARM=('POSIX(ON),ENVAR("_CEE_ENVFILE=DD:ENVIRON")/&PRM')  
//STEPLIB DD DISP=SHR,DSN=&HLQ..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
```

*Figure 2. Langage JCL du moniteur de travaux JES*

2. Le langage JCL doit résider dans une bibliothèque de procédure système (SYS1.PROCLIB est la valeur IBM par défaut).  
Pour plus de simplicité, il est recommandé que le nom de membre corresponde au nom de procédure (JMON dans l'exemple ci-dessus).
3. Il est conseillé que les programmes STC disposent d'un ID utilisateur dédié. Pour des raisons de sécurité, cet ID utilisateur doit être 'protégé' par la définition du mot clé NOPASSWORD (dans RACF). Cela signifie que RACF échouera à toute tentative d'ouverture de session qui nécessite un mot de passe, mais sans révoquer l'ID utilisateur.  
Utilisez l'exemple de commande ci-après pour créer un tel ID utilisateur, dans lequel userid est l'ID utilisateur en question, groupid est le groupe par défaut et uid est l'identificateur UNIX.

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

4. Les programmes STC doivent être définis dans le logiciel de sécurité (RACF, par exemple). Parmi les différentes méthodes qui existent pour définir un programme STC, il est recommandé d'utiliser la classe STARTED. Pour définir la classe STARTED, votre administrateur de sécurité doit lancer les commandes RACF ;

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

Pour ajouter le moniteur de travaux JES comme un programme STC, vous avez besoin des commandes RACF suivantes, où jmon désigne le nom de membre JCL et userid correspond à l'ID utilisateur qui dispose des droits qui seront utilisés.

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

**Remarque :** L'ajout du mot de passe TRUSTED(YES) au champ STDATA [ STDATA(USER(userid) TRUSTED(YES) ] permet d'éviter de définir les permissions individuelles nécessaires pour l'ID utilisateur du programme STC. RACF saute les contrôles de droits d'accès des fichiers pour les programmes STC dignes de confiance. Toutefois, avant de procéder de la sorte, assurez-vous que l'ID utilisateur en question ne peut pas être détourné en le protégeant comme décrit ci-avant.

Pour plus d'informations sur les tâches lancées et la sécurité, consultez *Security Server RACF Security Administrator's Guide (SA22-7683)*.

5. Une fois les tâches précédentes terminées, vous pouvez démarrer et arrêter le moniteur de travaux JES comme un programme STC.

L'opérateur système peut lancer les commandes suivantes sur la console (où JMON désigne le nom STC du moniteur de travaux JES) :

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

Si vous avez les droits d'accès nécessaires, vous pouvez saisir ces commandes depuis l'utilitaire SDSF, en faisant précéder la commande d'une barre oblique ('/'). Pour plus d'informations sur les commandes de la console, consultez le document *MVS System Commands (SA22-7627)*.

## Autorisations pour les serveurs

L'ID utilisateur assigné au moniteur de travaux JES doit avoir des droits d'accès READ à la bibliothèque de chargement Developer for System z, hlq.SFEKLOAD.

## Vérification du langage JCL d'initialisation du moniteur de travaux JES

Démarrez le travail utilisateur ou la tâche lancée selon la procédure décrite ci-dessus. Si le travail s'arrête avec un code retour 66, cela signifie que l'APF n'autorise pas hlq.SFEKLOAD.

---

## Sécurité et accès au spoule JES

### Accès conditionnel au spoule

Pour être autorisés à exécuter des opérations via le moniteur de travaux JES, les utilisateurs doivent recevoir un droit d'accès à la classe OPERCMDS. Cette opération peut être effectuée de façon conditionnelle, afin que les droits d'accès des utilisateurs ne soient effectifs que lorsqu'ils utilisent le moniteur de travaux JES. Pour utiliser cette vérification d'accès conditionnelle, la classe CONSOLE doit être active et une console doit être définie avec le nom JMON (JMON est le seul nom valide).

Par exemple, votre administrateur de sécurité lancera les commandes RACF suivantes :

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

Utilisez les commandes RACF suivantes pour autoriser les utilisateurs à émettre des commandes JES2 uniquement via le moniteur de travaux JES, dans lesquelles id est l'ID utilisateur ou l'ID groupe des utilisateurs de Developer for 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

#### ATTENTION :

**La définition des commandes JES à l'aide de l'accès universel NONE dans votre logiciel de sécurité peut avoir une incidence sur les autres applications et opérations. Vous devez les essayer avant de les activer sur un système de production.**

### Commandes disponibles

Le moniteur de travaux JES ne met pas à disposition des utilisateurs de Developer for System z un accès complet au spoule JES. Seules les commandes figurant dans le tableau 6 sont disponibles. Les commandes sont émises par la sélection de l'option appropriée dans la structure de menu du client (il n'y a pas d'invite de commande). La portée des commandes est encore limitée par les techniques décrites ci-après.

Tableau 6. Commandes de console du moniteur de travaux JES

Commande	JES2	JES3
Mettre en attente	\$Hjobid	*F,J=jobid,H
Publier	\$Ajobid	*F,J=jobid,R



Tableau 6. Commandes de console du moniteur de travaux JES (suite)

Commande	JES2	JES3
Annuler	\$Cjobid	*F,J=jobid,C
Purger	\$Cjobid,P	*F,J=jobid,C

**Remarque :** Si l'utilisateur n'est pas le propriétaire du fichier spoule, seule la visualisation est autorisée. Toutes les actions répertoriées dans tableau 6, à la page 23 entraîneront l'affichage du message "N'est pas autorisé pour le travail JOBxxx".

Sans autorisation pour ces commandes de console, les utilisateurs peuvent toujours soumettre des travaux et lire les sorties de travaux, s'ils disposent d'autorisations d'accès aux fichiers spoule.

## Accès limité aux fichiers spoule

Pour restreindre les utilisateurs à leur propres travaux sur le spoule JES, définissez l'instruction "LIMIT\_VIEW=USERID" dans le fichier de configuration du moniteur de travaux JES (FEJJCNGF). S'ils ont besoin d'accéder à davantage de travaux, mais pas à tous, utilisez les fonctions de protection du fichier spoule standard de votre produit de sécurité, telles que la classe JESSPOOL dans RACF.

Quand vous définissez d'autres protections, notez que le moniteur de travaux JES se sert de **SAPI** (interface de programme d'application SYSOUT) pour accéder au spoule. Par conséquent, l'utilisateur a besoin, au minimum, de droits d'accès de mise à jour des fichiers spoules, même pour des fonctionnalités de navigation. Cet élément prérequis ne s'applique pas si vous utilisez z/OS v 1.7 (z/OS 1.8 pour JES3) ou supérieure. Dans ce cas, les droits d'accès READ suffisent pour les fonctionnalités de navigation.

Pour plus d'informations sur la protection du fichier spoule JES, consultez le guide *Security Server RACF Security Administrator's Guide (SA22-7683)*.

---

## Personnalisation ELAXF\*, procédure de construction à distance

Developer for System z met à disposition des exemples de procédures de langage JCL qui peuvent être utilisés lors de la construction de langage JCL, de la génération de projets distants, et pour les fonctions de vérification syntaxique à distance des mappes BMS CICS, des écrans MFS IMS et des programmes COBOL, PL/I, Assembleur et C/C++.

Ces procédures permettent aux installations d'appliquer leurs propres normes. Cela garantit également que les développeurs utilisent les mêmes procédures, options de compilateur et niveaux de compilateur.

SMP/E installe ces exemples de procédures JCL dans hlq.SFEKSAMP. Si vous prévoyez d'utiliser ces procédures, vous devez :

1. Copier toutes les procédures nommées ELAXF\* dans une bibliothèque de procédures système (SYS1.PROCLIB) qui est disponible pour les utilisateurs.
2. Ces procédures copiées doivent être personnalisées pour refléter les conventions de dénomination utilisées sur le système cible. La personnalisation requise est documentée dans chaque procédure JCL.



Les exemples de procédures à copier et personnaliser sont répertoriés dans le tableau 7.

*Tableau 7. Modèles de procédure ELAXF\**

Membre	But
ELAXFADT	Modèle de procédure pour l'assemblage et le débogage des programmes Assembleur de haut niveau.
ELAXFASM	Modèle de procédure pour l'assemblage des programmes Assembleur de haut niveau.
ELAXFBMS	Modèle de procédure de création d'un objet BMS CICS BMS et de sa copie correspondante, dsect, ou du membre d'inclusion.
ELAXFCOC	Modèle de procédure pour l'exécution de compilations COBOL, de traductions CICS et DB2 intégrées.
ELAXFCOP	Modèle de procédure pour l'exécution du pré-processus DB2 des instructions SQL EXEC intégrées dans des programmes COBOL.
ELAXFCOT	Modèle de procédure pour l'exécution d'une traduction CICS des instructions CICS EXEC intégrées dans des programmes COBOL.
ELAXFCPC	Modèle de procédure pour l'exécution de compilations C.
ELAXFCPP	Modèle de procédure pour l'exécution de compilations C++.
ELAXFGO	Modèle de procédure pour l'étape GO.
ELAXFLNK	Modèle de procédure pour la liaison des programmes C/C++, COBOL, PLI et Assembleur de haut niveau.
ELAXFMFS	Modèle de procédure pour la création d'écrans IMS MFS.
ELAXFPLP	Modèle de procédure pour l'exécution du pré-processus DB2 des instructions SQL EXEC intégrées dans des programmes PLI.
ELAXFPLT	Modèle de procédure pour l'exécution d'une traduction CICS des instructions CICS EXEC intégrées dans des programmes PLI.
ELAXFPL1	Modèle de procédure pour l'exécution de compilations PL/I, de traductions CICS et DB2 intégrées.
ELAXFUOP	Modèle de procédure pour générer l'étape UOPT lors de la création de programmes de génération s'exécutant dans CICS ou des sous-systèmes IMS.

Le tableau 8 contient une liste de contrôle pour les différents qualificatifs de haut niveau utilisés dans les procédures ELAXF\*.

*Tableau 8. Liste de contrôle des qualificatifs de haut niveau ELAXF\**

Produit	Valeur par défaut HLQ	Valeur
RD/z	FEK	
CICS	CICSTS31.CICS	
DB2	DSN810	
IMS	IMS	
COBOL	IGY.V3R4M0	
PL/I	IBMZ.V3R6M0	
C/C++	CBC	
LE	CEE	
système LINKLIB	SYS1	
système MACLIB	SYS1	

Si les procédures ELAXF\* ne peuvent pas être copiées dans une bibliothèque de procédures système, copiez-les dans une bibliothèque privée et demandez aux utilisateurs de Developer for System z d'ajouter une carte JCLLIB (tout de suite après la carte JOB) aux propriétés de travail du client. Ne personnalisez pas l'exemple de langage JCL dans le fichier d'installation, car les opérations de maintenance peuvent remplacer ces membres et annuler vos personnalisations.

```
//MYJOB JOB <paramètres du travail>  
//PROCS JCLLIB ORDER=(hlq.SFEKSAMP)
```

**Remarque :** La génération JCL, les constructions de projets à distance et les opérations de vérification syntaxique à distance effectuées à partir d'un client Developer for System z supposent que ces procédures sont personnalisables et disponibles pour l'utilisateur.

Les noms des procédures et les noms des étapes des procédures correspondent aux propriétés par défaut livrées avec le client. Si vous décidez de modifier le nom de la procédure ou le nom d'une étape de la procédure, le fichier des propriétés correspondantes du client doit également être modifié. Nous recommandons de ne pas modifier les noms de la procédure et de l'étape.

**Remarque :** IBM Debug Tool devra être commandé, installé et configuré pour que le débogage à distance des programmes assembleur, COBOL et PL/I soit pris en charge. Consultez le *Rational Developer for System z - Guide de planification de l'hôte (SC11-2412-02)* pour connaître le niveau de Debug Tool qui est nécessaire pour votre version de Developer for System z. L'installation et la personnalisation de ce produit ne sont pas décrites dans le présent ouvrage.

---

## (Facultatif) Définition d'une transaction APPC pour le service Commandes TSO

La définition de transaction APPC est devenue optionnelle dans la version 7.1. Developer for System z utilise maintenant par défaut SCLM Developer Toolkit pour fournir le service Commandes TSO. Cette configuration s'effectue dans `rsed.envvars`, voir «Personnalisation de `rsed.envvars`, le fichier de configuration de l'Explorateur de systèmes éloignés RSE», à la page 39.

**Remarque :** Le programme transactionnel de langage JCL qui est utilisé par APPC pour lancer le service Commandes TSO a changé dans la version 7.1. Si vous avez précédemment affecté au service Commandes TSO la capture de la sortie ISPEXEC, vous devez soit définir une nouvelle transaction APPC, soit ajouter le mot clé NESTMACS à la ligne PARM, par exemple :

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

Le service Commandes TSO est mis en oeuvre comme un programme transactionnel APPC, FEKFRSRV et doit être actif pour que les connexions Developer for System z s'établissent entre l'hôte et le client. FEKFRSRV joue le rôle de serveur hôte pour exécuter les commandes TSO provenant du poste de travail via TCP/IP. Il n'est pas nécessaire d'installer APPC sur le poste de travail, car celui-ci communique avec FEKFRSRV via TCP/IP. Chaque poste de travail peut disposer d'une connexion active à plusieurs hôtes simultanément.

**Remarque :** Si vous n'êtes pas familier avec APPC, voir Annexe F, «Configuration APPC», à la page 131 avant de poursuivre la présente section.

## Préparation

- Les tâches suivantes sont prérequis et doivent être effectuées avant de configurer le serveur de Commandes TSO. Les manuels indiqués décrivent ces tâches.
  1. Installation, configuration, et démarrage de VTAM sur le système z/OS. Pour plus d'informations, voir *Communications Server bookshelf (F1A1BK61)*.
  2. Installation, configuration et démarrage de TCP/IP sur le système z/OS. Pour plus d'informations, voir le manuel Annexe C, «Configuration du TCP/IP», à la page 101 et *Communications Server bookshelf (F1A1BK61)*.
  3. Configurez et démarrez APPC et le sous-système ASCH (planificateur de transactions APPC). Pour plus d'informations, voir Annexe F, «Configuration APPC», à la page 131 et le manuel *MVS bookshelf (IEA2BK60)*.
- Le langage REXX de la figure 3 permet de gérer APPC via les panneaux ISPF.

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

Figure 3. Langage REXX pour les panneaux ISPF d'APPC

**Remarque :** Vous devez savoir que vous pouvez désactiver la transaction APPC avec cet outil ; la transaction existe toujours mais n'accepte aucune connexion.

- La définition de la transaction APPC nécessite des compétences dans plusieurs domaines du système d'exploitation MVS. Parcourez les différents points de la liste ci-dessous avec des administrateurs expérimentés avant de poursuivre.

Tableau 9. Liste de contrôle de transaction APPC

Com pétences	Information obligatoire : • Valeur par défaut • Emplacement de la réponse	Valeur
APPC	Nom de fichier de TPDATA • Par défaut : SYS1.APPCTP • La valeur se trouve dans SYS1.PARMLIB(APPCPMxx)	
APPC	Nom de transaction à utiliser (peut ne pas exister) • Par défaut : FEKFRSRV • Les transactions existantes sont interrogées en sélectionnant "Administration de profil du programme transactionnel" dans le menu ISPF APPC	

Tableau 9. Liste de contrôle de transaction APPC (suite)

Com pétences	Information obligatoire : • Valeur par défaut • Emplacement de la réponse	Valeur
APPC	Classe de transaction APPC à utiliser • Par défaut : A • Les classes APPC sont définies dans SYS1.PARMLIB(ASCHPMxx)	
WLM/ SRM	Domaine et groupe de performances TSO • Aucune valeur IBM par défaut (dépend du site)	
RACF	Chaque utilisateur de Developer for System z dispose d'un accès à un segment OMVS (obligatoire). • Aucune valeur IBM par défaut (dépend du site) • La commande RACF TSO <b>OMVS d'un ID utilisateur de l'unité logique</b> affiche un segment OMVS personnel existant	
RACF	Chaque utilisateur de Developer for System z doit disposer de droits d'accès READ à hlq.SFEKPROC (FEKFRSRV) • Aucune valeur IBM par défaut (dépend du site) • La commande RACF TSO <b>LD AUTHUSER DATASET(hlq.SFEKPROC.**)</b> affiche les personnes et groupes ainsi que leurs niveaux d'accès aux fichiers du profil de fichier.	

Pour plus d'informations sur la gestion WLM/SRM, consultez le manuel *MVS Planning Workload Management (SA22-7602)*. Pour plus d'informations sur les segments OMVS et les profils de protection de fichiers, consultez le guide *Security Server RACF Security Administrator's Guide (SA22-7683)*.

**Remarque :** La classe de transaction APPC utilisée doit comporter suffisamment de demandeurs APPC pour en autoriser un par utilisateur de Developer for System z.

**Remarque :** La transaction APPC utilise le langage REXX exec FEKFRSRV, situé dans hlq.SFEKPROC. Ne modifiez pas cet emplacement si vous souhaitez avoir la possibilité d'activer automatiquement une opération de maintenance SMP/E.

## Implémentation

Le programmeur système ou l'administrateur APPC doit effectuer les étapes suivantes pour configurer la fonction de commande :

1. Définissez les informations de calendrier (classe) pour le planificateur de transactions APPC si vous n'utilisez pas de classe de transaction existante. Indiquez une définition dans SYS1.PARMLIB(ASCHPMxx) pour la classe que le programme transactionnel FEKFRSRV doit utiliser. Cette classe est utilisée dans le modèle JCL hlq.SFEKSAMP(FEKAPPCC). Par conséquent, la classe FEKAPPCC doit correspondre à la classe définie dans SYS1.PARMLIB(ASCHPMxx). Par exemple :

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

**Remarque :** Le service Commandes TSO nécessite également que les spécifications par défaut soient indiquées aux sections OPTIONS et TPDEFAULT de SYS1.PARMLIB(ASCHPMxx). Pour plus d'informations, voir l'Annexe F, «Configuration APPC», à la page 131.

2. Définissez la transaction APPC qui agira en tant que serveur de commandes. Pour définir cette transaction, vous pouvez utiliser le modèle JCL hlq.SFEKSAMP(FEKAPPCC). Des instructions sur la méthode de personnalisation du langage JCL se trouvent dans le JCL. Le modèle JCL permet également d'afficher, hlq.SFEKSAMP(FEKAPPCL), ou de supprimer, hlq.SFEKSAMP(FEKAPPCX), la transaction.

**Remarque :** Si vous avez modifié le nom de programme transactionnel (FEKFRSRV par défaut), le nouveau nom doit être attribué à \_FEKFSCMD\_TP\_NAME\_ dans rsed.envvars voir «Personnalisation de rsed.envvars, le fichier de configuration de l'Explorateur de systèmes éloignés RSE», à la page 39.

3. Contrôlez la priorité de distribution du programme transactionnel hlq.SFEKPROC(FEKFRSRV) en associant FEKFRSRV à un domaine et un groupe de performances dans Workload Manager (WLM). FEKFRSRV émettant des commandes TSO, il doit être affecté à un groupe de performances TSO.
4. Définissez un segment OMVS par défaut pour le système ou un segment individuel pour chaque utilisateur devant utiliser l'édition, la compilation ou le débogage à distance.
5. Octroyez aux utilisateurs de Developer for System z des droits d'accès READ à hlq.SFEKPROC(FEKFSERV), le serveur de Commandes TSO.

**Remarque :** La vérification de la configuration sera effectuée au cours de la vérification RSE. En effet, RSE met en application la connexion TCP/IP au serveur de commandes TSO.

---

## (Facultatif) Personnalisation ELAXM\*, membres de procédure mémorisée DB2

La personnalisation suivante est nécessaire pour incorporer les modèles de procédures mémorisées DB2 (Compilateur de procédures mémorisées PL/I et COBOL) dans votre système. Vous aurez besoin de l'aide d'un administrateur WLM et d'un administrateur DB2 pour effectuer ces tâches.

Quand l'outil SMP/E s'applique, le modèle de bibliothèque hlq.SFEKSAMP et la bibliothèque de procédure hlq.SFEKPROC contiennent les membres de la procédure mémorisée DB2 classée dans le tableau 10.

Tableau 10. Modèle ELAXM\* - Membres de la procédure mémorisée DB2

Membre	But
hlq.SFEKSAMP(ELAXMJCL)	Modèle de langage JCL définissant le compilateur de procédures mémorisées PL/I et COBOL pour DB2.
hlq.SFEKSAMP(ELAXMSAM)	Modèle de procédure JCL de l'espace adresse WLM du compilateur de procédures mémorisées PL/I et COBOL.
hlq.SFEKPROC(ELAXMREX)	Code REXX du compilateur de procédures mémorisées PL/I et COBOL.

**Remarque :** La procédure mémorisée DB2 utilise le code REXX exec ELAXMREX, situé dans hlq.SFEKPROC. Ne modifiez pas cet emplacement si vous souhaitez avoir la possibilité d'activer automatiquement une opération de maintenance SMP/E.

**Remarque :** Si vous souhaitez renommer les membres ELAXMSAM ou ELAXMREX, voir l'Annexe A, «Exécution de plusieurs instances de Developer for System z», à la page 81.

1. Copiez ELAXMSAM dans une bibliothèque de procédure (comme SYS1.PROCLIB) disponible pour les utilisateurs de procédures mémorisées DB2 et personnalisez le langage JCL conformément aux commentaires. Vérifiez que la carte de définition de données SYSEXEC désigne la bibliothèque où réside le membre ELAXMREX (hlq.SFEKPROC par défaut).
2. Au moyen des panneaux de gestion de charge de travail (WLM), associez un environnement d'application à la procédure JCL de l'espace adresse WLM du compilateur de procédures mémorisées PL/I et COBOL. Pour plus d'informations sur la méthode à utiliser, consultez le manuel *MVS Planning Workload Management (SA22-7602)*.

**Remarque :** Vous pouvez créer un nouvel environnement d'applications dans WLM pour le compilateur de procédures mémorisées PL/I et COBOL, ou bien ajouter les définitions nécessaires à un environnement existant.

3. Copiez ELAXMJCL dans une bibliothèque JCL privée, personnalisez la copie conformément aux commentaires et demandez à un administrateur DB2 de soumettre le travail. Vérifiez que la clause WLM ENVIRONMENT de l'instruction CREATE PROCEDURE spécifie le nom de la procédure d'environnement WLM qui a été défini pour le compilateur de procédures mémorisées PL/I et COBOL (ELAXMSAM, par défaut).

---

## (Facultatif) Personnalisation de la prise en charge de la langue bidirectionnelle CICS (bidi)

Les composants de l'outil EST (Developer for System z Enterprise Service Tools) prennent en charge différents formats de messages d'interface en arabe et en hébreu, ainsi que la présentation et l'édition des données bidirectionnelles dans tous les éditeurs et dans toutes les vues. Dans les applications de terminal, les écrans de gauche à droite et de droite à gauche sont pris en charge, ainsi que les zones numériques et les zones orientées dans le sens contraire de l'écran.

Les fonctions et fonctionnalités bidirectionnelles supplémentaires comprennent notamment :

- La spécification dynamique par le demandeur de service EST, des attributs bidirectionnels des messages d'interface.
- Le traitement des données bidirectionnelles dans les flux de services, est basé sur des attributs bidirectionnels (type de texte, orientation du texte, permutation numérique et permutation symétrique). Ces attributs peuvent être spécifiés à différentes étapes de la création des flux, à la fois pour les flux d'interface et de terminal.
- Le code d'exécution généré par EST comprend la conversion des données entre des zones de message dont les attributs bidirectionnels sont différents.

**Avertissement :** Le module de chargement (nom et codage) a été modifié par rapport aux éditions précédentes (antérieures à la version 7.0). Si vous avez activé une édition bidi précédente, le ou les anciens modules de chargement doivent être supprimés de la concaténation RPL CICS. Le ou les emplacements par défaut sont :

- FEJ.SFEJDLL(FEJBDTRN)
- FEJ.SFEJLMD(FEJBDTRE) (facultatif, version non-PDSE de FEJBDTRN pour l'hébreu)

Vous allez avoir besoin de l'aide de votre administrateur CICS pour effectuer les tâches suivantes :

1. Placez le programme hlq.SFEKLOAD(FEJBDTRX) dans la concaténation RPL CICS (instruction de définition de données DFHRPL). La méthode recommandée consiste à ajouter le fichier d'installation à la concaténation pour que l'opération de maintenance appliquée soit automatiquement disponible dans CICS.

Les transformations bidirectionnelles EST sont effectuées dans l'environnement d'exécution de flux de service (SFR) de CICS par le module FEJBDTRX. Le module FEJBDTRX est appelé lorsque les conversions bidirectionnelles sont nécessaires dans le code généré par EST (lorsque le mappage est généré dans des messages dont les attributs bidirectionnels sont différents).

**Remarque :** Si vous ne concaténez pas le répertoire d'installation mais que vous copiez le module FEJBDTRX dans un fichier, nouveau ou existant, notez que ce module est une bibliothèque de chargement dynamique et DOIT résider dans une bibliothèque PDSE.

**Remarque :** Dans la version 7.1, le programme hlq.SFEKDLL(FEJBDTRX) a été déplacé dans une nouvelle bibliothèque de chargement, hlq.SFEKLOAD(FEJBDTRX).

2. Demandez à l'administrateur CICS de définir l'installation automatique sur autoactive.

Si l'installation automatique est définie sur autoInactive, définissez le programme FEJBDTRX sur CICS au moyen de la commande CEDA, par exemple :

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

De plus, le code généré par EST peut prendre en charge la transformation bidi dans d'autres environnements que SFR CICS (par exemple, des applications par lots). Vous pouvez inclure dans les générateurs EST des appels de routines de conversion bidirectionnelle en spécifiant les options de transformation bidi



appropriées dans les assistants de génération EST et en éditant des liens entre les programmes générés et la bibliothèque de conversion bidirectionnelle appropriée, h1q.SFEKLOAD.

**Avertissement :** Le code bidi qui a été créé dans les précédentes versions (avant la version 7.0) doit être RECOMPILE pour utiliser le nouveau module FEJBDTRX.

---

## (Facultatif) Personnalisation du gestionnaire de déploiement ADM

Le Gestionnaire de déploiement d'application (ADM) propose une approche de déploiement et une interface de programmation d'application de déploiement communes pour tous les composants de Developer for System z.

De plus, ADM met à disposition un client et un serveur (basé sur l'hôte) de définition de ressource CICS (CRD), qui fournissent les fonctions suivantes :

1. L'autorisation aux développeurs d'applications de définir des ressources CICS de manière limitée, contrôlée et sécurisée.
  - Les valeurs par défaut pour la définition des ressources CICS sont mises à disposition par l'administrateur CICS et stockées au niveau du référentiel du serveur CRD sur l'hôte.
  - La capacité de mise à jour de l'attribut ressource est contrôlée par l'administrateur CICS (mettre à jour, protéger, masqué).
  - La CRD se limite à un petit ensemble de ressources dont les développeurs d'applications ont souvent besoin. Cet ensemble comprend TranDB2, Doctemplate, Fichier, Groupe de mappes, Création du type de processus, Programme, Fichier de données transitoires et Transaction.
  - L'autorisation de créer des définitions de ressource CICS est contrôlée par RACF ou par un autre gestionnaire de sécurité externe.
2. L'interdiction d'obtenir un accès de développement CICS aux fichiers VSAM non autorisés ou incorrects en mettant à disposition de l'administrateur CICS le contrôle des noms de fichiers physiques dans les définitions de Fichiers. Cette information de liaison est stockée dans le référentiel de CRD sur l'hôte.
3. Divers caractères de signalisation de développement pour serveur CRD.
  - Nouvelle copie pour les programmes et les groupes de mappes
  - Liste des régions CICS
  - Liste DFHRPL
4. Divers caractères de signalisation de développement de service Web pour serveur CRD.
  - Exécution d'une analyse de pipeline pour effectuer une installation automatique des définitions URIMAP et WEBSERVICE
  - Mise à disposition de la liste des répertoires de collecte de pipeline et WSBIND
  - Mise à disposition de la liste des répertoires des fichiers WSDL
  - Mise à disposition de la liste des points d'arrêt URI

Developer for System z met à disposition trois transactions qui sont utilisées par le serveur CRD lors de la définition et de la consultation des ressources CICS. Un exemple de langage COBOL est fourni afin de permettre une personnalisation en fonction du site.

### ADMD

Pour les demandes qui définissent les ressources par défaut du service Web et de CICS. Généralement destiné aux administrateurs CICS.



**ADMI** Pour les demandes qui définissent, installent ou désinstallent les ressources CICS.

**ADMR**

Pour toutes les autres demandes qui récupèrent des informations sur les ressources ou l'environnement CICS.

Pour plus d'informations sur le gestionnaire de déploiement d'application, voir *Rational Developer for System z Application Deployment Manager User's Guide* (SC11-2793-00).

Les étapes de personnalisation ci-après sont nécessaires pour activer le serveur CRD.

**Remarque :** L'assistance de l'administrateur CICS est nécessaire pour effectuer certaines des tâches ci-après.

Avant d'installer la version 7.1.1, si vous utilisiez la version précédente du serveur CRD, il est recommandé d'enregistrer la personnalisation associée comme décrit à la section «Sauvegarde des fichiers déjà configurés», à la page 11.

Copiez les membres à personnaliser depuis le répertoire d'installation vers une bibliothèque personnelle et personnalisez les copies afin d'éviter de les écraser lors d'une opération de maintenance :

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

facultatif (gestionnaire de message de pipeline)

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

facultatif (mise à jour de correctif CSD pour les régions non-primaires)

- hlq.SFEKSAMP(ADNARCSD)

## Référentiel de CRD

Personnalisez et soumettez un travail ADNVSAM pour attribuer et initialiser le fichier de méthode d'accès VSAM du référentiel de serveur CRD. Pour obtenir des instructions de personnalisation, consultez la documentation qui se trouve dans ADNVSAM.

Il est conseillé de créer un référentiel séparé pour chaque région de connexion primaire CICS. Le partage de référentiel implique que toutes les régions CICS concernées utiliseront les mêmes valeurs stockées dans le référentiel, et que de multiples espaces adresse écriront dans la méthode d'accès VSAM, qui doit être configurée correctement pour cela.

**Remarque :** Sauf indication contraire, votre référentiel de serveur CRD actuel (qui comporte les valeurs personnalisées) peut être réutilisé d'une édition de Developer for System z à l'autre.

## Région de connexion CICS primaire

Le serveur CRD doit être défini pour la région de connexion primaire. Il s'agit de la région qui traite les requêtes de service Web en provenance de Developer for System z.

- Placez les modules de chargement ADM hlq.SFEKLOAD(ADNCRDS) et hlq.SFEKLOAD(ADNCRDR) dans la concaténation RPL CICS (instruction de définition de données DFHRPL) de la région de connexion CICS primaire. La méthode recommandée consiste à ajouter le fichier d'installation à la concaténation pour que l'opération de maintenance appliquée soit automatiquement disponible dans CICS.
- Personnalisez et soumettez un travail ADNPCCSD pour mettre à jour la définition de système CICS (CSD) pour la région de connexion CICS principale. Pour obtenir des instructions de personnalisation, consultez la documentation qui se trouve dans ADNPCCSD.
- Utilisez la commande CEDA appropriée pour installer le groupe ADM pour cette région, par exemple :  
CEDA INSTALL GROUP(ADNPCRGP)

## Gestionnaire de message de pipeline

Le gestionnaire de messages du pipeline (ADNSMSGH) est utilisé par sécurité dans le traitement de l'ID utilisateur et des mots de passe dans l'en-tête du protocole SOAP. ADNSMSGH est référencé par le fichier de configuration du pipeline et doit donc être placé dans la concaténation RPL CICS. ADNSMSGH est également utilisé pour définir l'ID de transaction pour ADMD, ADMR ou ADMI selon l'opération demandée. Vous souhaitez peut-être personnaliser ADNSMSGH pour utiliser des ID de transaction différents.

Utilisation des valeurs par défaut :

- Placez les modules de chargement hlq.SFEKLOAD(ADNSMSGH) dans la concaténation RPL CICS (instruction de définition de données DFHRPL) de la région de connexion CICS primaire. La méthode recommandée consiste à ajouter le fichier d'installation à la concaténation pour que l'opération de maintenance appliquée soit automatiquement disponible dans CICS.

Personnalisation d'ADNSMSGH :

- Personnalisez l'échantillon de code source ADNSMSGH (langage COBOL) pour le gestionnaire de message de pipeline.
- Personnalisez et soumettez un travail ADNCMSGH pour compiler la source personnalisée ADNSMSGH. Pour obtenir des instructions de personnalisation, consultez la documentation qui se trouve dans ADNCMSGH.
- Placez le module de chargement résultant ADNSMSGH dans la concaténation RPL CICS (instruction de définition de données DFHRPL) de la région de connexion CICS primaire.

**Remarque :** Assurez-vous que le module de chargement personnalisé ADNSMSGH est localisé avant toute référence à hlq.SFEKLOAD, sinon la valeur par défaut sera utilisée.

## (Facultatif) Régions de connexion CICS non primaires

Le serveur CRD peut également être utilisé avec une ou plusieurs régions de connexion non primaire, qui sont généralement des régions gérant les applications (AOR).

**Remarque :** Il n'est pas nécessaire de suivre ces étapes si CICSplex SM est utilisé pour gérer l'environnement CICS.

- Placez le module de chargement ADM hlq.SFEKLOAD(ADNCRDS) dans la concaténation RPL CICS (instruction de définition de données DFHRPL) de ces

régions de connexion non primaire. La méthode recommandée consiste à ajouter le fichier d'installation à la concaténation pour que l'opération de maintenance appliquée soit automatiquement disponible dans CICS.

- Personnalisez et soumettez un travail ADNARCSD pour mettre le correctif CSD à jour pour d'autres régions de connexion non primaire. Pour obtenir des instructions de personnalisation, consultez la documentation qui se trouve dans ADNARCSD.
- Utilisez la commande CEDA appropriée pour installer le groupe ADM pour ces régions, par exemple :  
CEDA INSTALL GROUP(ADNARRGP)



---

## Chapitre 4. Activation des composants de Developer for System z z/OS UNIX

Avant d'installer la version 7.1.1, si vous utilisiez la version précédente de Developer for System z, il est recommandé d'enregistrer la personnalisation associée décrite dans la section «Sauvegarde des fichiers déjà configurés», à la page 11.

Si vous n'êtes pas familiarisé avec le système z/OS UNIX, il est conseillé de demander l'aide d'un administrateur expérimenté dans z/OS UNIX ou un autre UNIX pour effectuer les tâches répertoriées dans ce chapitre.

Les commandes z/OS UNIX nécessaires pour effectuer ces tâches sont décrites brièvement pour votre convenance. Pour plus d'informations sur ces commandes, consultez l'ouvrage *UNIX System Services Command Reference (SA22-7802)*, sauf indication contraire.

Les tâches décrites ci-après nécessitent des actions de votre part dans le système z/OS UNIX. Vous pouvez les effectuer en lançant la commande TSO **OMVS**. Utilisez **exit** pour retourner à TSO.

MVS permet de modifier les fichiers z/OS UNIX à l'aide de l'utilitaire ISPF par l'intermédiaire de la commande **OEDIT**. Cette commande s'utilise sous TSO et OMVS.

La plupart des fichiers z/OS UNIX présentent des droits d'accès en écriture restreints au propriétaire du fichier. Cette restriction peut être ignorée de plusieurs manières.

- **UID 0**

Cette solution n'est pas recommandée pour les ID utilisateur "humains" dans la mesure où il n'y a pas de restrictions liées à z/OS UNIX.

- **Droits d'accès READ au profil BPX.SUPERUSER dans la classe FACILITY**

Permet à l'utilisateur d'obtenir un UID 0 par l'intermédiaire de la commande **su**. Il s'agit de la configuration recommandée.

- **Droits d'accès UPDATE au profil SUPERUSER.FILESYS dans la classe UNIXPRIV**

Permet à un utilisateur de lire/écrire dans tout fichier, et de lire ou de rechercher dans tout répertoire. Des droits d'accès **CONTROL** (ou plus) à ce profil de sécurité ajoutent l'écriture dans tout répertoire à la liste des autorisations.

Pour plus d'informations concernant la sécurité de z/OS UNIX, voir *UNIX System Services Planning (GA22-7800)*.

Toutes les instructions de chemin d'accès `/usr/lpp/wd4z/` que vous trouverez dans ce chapitre se rapportent au chemin utilisé au cours de l'installation de Developer for System z. Le chemin par défaut est `/usr/lpp/wd4z/`, mais il peut ne pas s'appliquer à votre site.

**Remarque :** Developer for System z requiert que le protocole TCP/IP dispose du nom d'hôte correct lors de son initialisation. Cela implique que les différents fichiers de configuration du TCP/IP et du programme de

résolution Resolver soient configurés correctement. Pour plus d'informations sur la personnalisation nécessaire, voir Annexe C, «Configuration du TCP/IP», à la page 101.

Vous pouvez tester votre configuration TCP/IP à l'aide de la commande TSO **HOMETEST**. Pour plus d'informations sur cette commande, voir *Communications Server IP System Administrator's Commands* (SC31-8781).

**Remarque :** Developer for System z dépend d'INETD pour la configuration de la connexion client-hôte. Pour plus d'informations sur INETD, voir Annexe D, «Configuration d'INETD», à la page 109.

**Remarque :** Les actions à distance (basées sur l'hôte) pour les sous-projets z/OS UNIX nécessitent que REXEC ou SSH soit actif sur l'hôte.

**Remarque :** Java 31-bit est obligatoire, et tous les utilisateurs de z/OS UNIX doivent disposer de droits d'accès EXECUTE et READ aux répertoires de système hiérarchique de fichiers Java.

**Avertissement :** La version 64 bits Java n'est pas prise en charge.

Les publications suivantes peuvent être utiles pour comprendre le système z/OS UNIX :

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

---

## Enregistrement du fichier de configuration rsed.envvars dans un autre répertoire

Il est recommandé de copier le fichier `/usr/lpp/wd4z/rse/lib/rsed.envvars` dans un nouveau répertoire (tel que `/etc/wd4z/`) et de personnaliser la copie afin d'éviter d'écraser votre personnalisation lors d'une opération de maintenance. Toutefois, lors de cette procédure, vous devez également copier les fichiers suivants, depuis le répertoire d'installation (`/usr/lpp/wd4z/rse/lib/`, par défaut) vers le nouvel emplacement :

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

**Remarque :** Bien qu'aucune personnalisation ne soit nécessaire pour les fichiers `*.zseries`, il est important de remplacer les versions précédentes par les versions en cours. Ceci afin de les conserver synchronisés avec la version en cours de `rsed.envvars`.

Les fichiers dont la liste figure dans le tableau 11 doivent être également copiés si vous prévoyez d'utiliser les dispositifs en option dont ils font partie :

Tableau 11. Fichiers de configuration facultatifs

fichier	fonction
projectcfg.properties	Projets basés sur l'hôte  Voir «(Facultatif) Personnalisation de projectcfg.properties, configuration de projets hôte», à la page 59
FMIEXT.properties	Intégration du gestionnaire de fichiers  Voir «(Facultatif) Personnalisation de FMIEXT.properties, intégration de File Manager», à la page 60
CRASRV.properties	CARMA  Voir Chapitre 5, «(Facultatif) Activation d'IBM Common Access Repository Manager (CARMA)», à la page 63

Les différents exemples de commandes suivants,

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

créent un répertoire nommé `/etc/wd4z` et copient `rsed.envvars` à partir du répertoire actuel vers `/etc/wd4z`. Répétez la commande de copie pour le reste des fichiers.

Le résultat de la copie est vérifiable à l'aide de la commande `ls /etc/wd4z`, qui doit donner une sortie semblable à ceci (\$ est l'invite de z/OS UNIX) :

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

**Remarque :** Pour conserver tous les fichiers Developer for System z dans un même système hiérarchique de fichiers (privé), et placer en même temps les fichiers de configuration dans `/etc/wd4z`, utilisez les liens symboliques. Les exemples de commandes suivants permettent de créer un nouveau répertoire (`/usr/lpp/wd4z/rse/lib/cust`) dans le système hiérarchique de fichiers d'installation et de définir un lien symbolique (`/etc/wd4z`) vers ce dernier :

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

## Personnalisation de `rsed.envvars`, le fichier de configuration de l'Explorateur de systèmes éloignés RSE

Toutes les méthodes de connexion au client Developer for System z utilisent les mêmes variables définies dans le fichier `rsed.envvars`, situé par défaut dans le répertoire d'installation, `/usr/lpp/wd4z/rse/lib/`, mais qui peut être copié dans un autre répertoire à l'étape précédente. Le modèle de fichier fourni comporte les

instructions dont la liste est présentée dans la figure 4, à la page 41, où les lignes mises en commentaire commencent par un dièse (#).



```

#=====
# (1) required customizations
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="$_RSE_JAVAOPTS -DTSO_SERVER=APPC"
#=====
# (2) required customizations if SCLMDT is used
_CMDSERV_BASE_HOME=/usr/lpp/SCLMDT
_CMDSERV_BASE_LOAD=BWB.SBWBLOAD
_CMDSERV_CONF_HOME=/etc/SCLMDT
_CMDSERV_WORK_HOME=/var/SCLMDT
STEPLIB=NONE
#STEPLIB=$_CMDSERV_BASE_LOAD
_RSE_CMDSERV_OPTS=""
#=====
# (3) optional customizations
#_RSE_PORTRANGE=8108-8118
#_FEKFLOCK_USERID=user_id
#_FEKFLOCK_JOBNAME=job_name
#_FEKFSCMD_TP_NAME=tp_name
#_FEKFSCMD_PARTNER_LU=lu_name
#=====
# (4) do not change unless directed by IBM support center
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) additional environment variables

```

Figure 4. *rse.d.envvars* – Fichier de configuration RSE

Les définitions suivantes sont requises :

#### **JAVA\_HOME**

Répertoire de base Java. La valeur par défaut est /usr/lpp/java/J1.4. Modifiez en fonction de votre installation Java.

#### **RSE\_HOME**

Répertoire de base RSE. La valeur par défaut est /usr/lpp/wd4z. Modifiez en fonction de votre installation de Developer for System z.

**TZ** Sélecteur de fuseau horaire. La valeur par défaut est EST5EDT. Le fuseau horaire par défaut est le temps universel coordonné + 5 heures (heure d'été de la côte est). Modifiez cette valeur pour afficher votre fuseau horaire. Pour plus d'informations, voir *UNIX System Services File System Interface Reference (SA22-7808)*.

#### **LANG**

Indique le nom des paramètres régionaux par défaut. La valeur par défaut est C. C indique les paramètres régionaux POSIX, et Ja\_JP indique les paramètres régionaux japonais. Modifiez cette valeur pour afficher vos paramètres régionaux.

**PATH** Chemin d'accès de la commande. La valeur par défaut est /bin:/usr/sbin:.. Peut être modifiée au besoin.

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

Options Java supplémentaires pour le partage de classe. La valeur par défaut est "". Pour plus d'informations sur cette définition, voir «(Facultatif) Définition des paramètres de démarrage Java supplémentaires avec \_RSE\_\*OPTS», à la page 46.

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

Options Java supplémentaires spécifiques à l'explorateur de systèmes éloignés RSE. La valeur par défaut est "". Pour plus d'informations sur cette définition, voir «(Facultatif) Définition des paramètres de démarrage Java supplémentaires avec \_RSE\_\*OPTS», à la page 46.

Developer for System z utilise par défaut SCLM Developer Toolkit pour fournir le service Commandes TSO. APPC est utilisé lorsque l'option \_RSE\_JAVAOPTS suivante est retirée de la mise en commentaire :

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

**Remarque :** Les deux méthodes de service de Commandes TSO nécessitent plus de personnalisations que simplement celles de rsed.envvars. Les personnalisations nécessaires à la configuration d'APPC sont décrites dans «(Facultatif) Définition d'une transaction APPC pour le service Commandes TSO», à la page 26, celles pour SCLMDT sont décrites dans «Personnalisation d'ISPF.conf, fichier de configuration d'ISPF», à la page 51.

Les définitions suivantes sont nécessaires si SCLM Developer Toolkit est utilisé, soit pour le service Commandes TSO, soit lorsque le module d'extension SCLMDT est installé dans le client Developer for System z.

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

Répertoire initial de SCLM Developer Toolkit. La valeur par défaut est usr/lpp/SCLMDT. Modifiez en fonction de votre installation de SCLM

Developer Toolkit. Cette directive n'est nécessaire que dans le cas où SCLM Developer Toolkit est utilisé (service de Commandes TSO ou module d'extension de client).

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

Bibliothèque de chargement de SCLM Developer Toolkit. La valeur par défaut est BWB.SWBLOAD. Modifiez en fonction de votre installation de SCLM Developer Toolkit. Cette directive n'est nécessaire que dans le cas où SCLM Developer Toolkit est utilisé (service de Commandes TSO ou module d'extension de client).

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

Répertoire de configuration de base de SCLM Developer Toolkit. La valeur par défaut est /etc/SCLMDT. Modifiez en fonction de votre personnalisation de SCLM Developer Toolkit. Cette directive n'est nécessaire que dans le cas où SCLM Developer Toolkit est utilisé (service de Commandes TSO ou module d'extension de client).

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

Répertoire de travail de base de SCLM Developer Toolkit. La valeur par défaut est /var/SCLMDT. Modifiez en fonction de votre personnalisation de SCLM Developer Toolkit. Cette directive n'est nécessaire que dans le cas où SCLM Developer Toolkit est utilisé (service de Commandes TSO ou module d'extension de client).

#### **STEPLIB**

STEPLIB pour le Serveur RSE. La valeur par défaut est NONE. Ne modifiez pas cette ligne, qui agit en tant que valeur par défaut.

Developer for System z utilise LINKLIST par défaut pour accéder à la bibliothèque de chargement de SCLM Developer Toolkit. STEPLIB est utilisé lorsque la directive STEPLIB suivante est retirée de la mise en commentaire :

```
STEPLIB=$_CMDSERV_BASE_LOAD
```

**Remarque :** L'utilisation de STEPLIB dans z/OS UNIX présente un impact négatif sur les performances, comme il est indiqué dans «Eviter l'emploi de STEPLIB», à la page 73.

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

Options Java supplémentaires spécifiques au service de Commandes TSO. La valeur par défaut est "". Pour plus d'informations sur cette définition, voir «(Facultatif) Définition des paramètres de démarrage Java supplémentaires avec \_RSE\_\*OPTS», à la page 46. Cette directive n'est nécessaire que dans le cas où SCLM Developer Toolkit est utilisé (service de Commandes TSO ou module d'extension de client).

Les définitions suivantes sont facultatives. Si vous les omettez, les valeurs par défaut seront utilisées.

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

Indique la gamme de ports que le serveur RSE peut ouvrir pour communiquer avec un client. Chaque port peut être utilisé par défaut. Pour plus d'informations sur cette définition, voir «(Facultatif) Définition de la gamme de ports disponible pour RSE», à la page 45.

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

ID utilisateur utilisé par le gestionnaire de verrouillage. La valeur par défaut correspond à l'ID utilisateur de connexion.

**\_FEKFLOCK\_JOBNAME\_**

Nom de travail utilisé par le gestionnaire de verrouillage. La valeur par défaut est FEKFLK00.

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

Nom de programme transactionnel APPC. La valeur par défaut est FEKFRSRV. Supprimez la mise en commentaire et modifiez cette définition si vous n'avez pas utilisé le nom de programme transactionnel par défaut lors de la définition de la transaction APPC. Pour plus d'informations sur la transaction APPC, voir «(Facultatif) Définition d'une transaction APPC pour le service Commandes TSO», à la page 26.

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

Permet de forcer l'Explorateur de systèmes éloignés RSE à utiliser cette unité logique de base APPC. Pour plus d'informations sur cette définition, voir l'Annexe F, «Configuration APPC», à la page 131.

Les définitions suivantes sont nécessaires, et ne doivent pas être modifiées sans instruction du point service IBM.

**RSE\_LIB**

Chemin d'accès à la bibliothèque de l'Explorateur de systèmes éloignés RSE. La valeur par défaut est \$RSE\_HOME/rse/lib. Ne pas modifier.

**ICU\_LIB**

Chemin d'accès à la bibliothèque de composants internationaux pour Unicode (ICU). La valeur par défaut est \$RSE\_HOME/icuc/lib. Ne pas modifier.

**\_CEE\_RUNOPTS**

Options d'exécution de Language Environment (LE) utilisées par le processus. La valeur par défaut est ALL31(ON) HEAP(32M,32K,ANYWHERE,KEEP,,) TRAP(ON). Ne pas modifier.

**\_CEE\_DMPTARG**

Emplacement d'exportation de Language Environment (LE) z/OS UNIX utilisé par la machine virtuelle Java (JVM). La valeur par défaut est \$HOME/.eclipse/RSE/\$RSE\_USER\_ID. Ne pas modifier.

**\_BPX\_SHAREAS**

Exécute les processus d'avant-plan dans le même espace adresse que le shell. La valeur par défaut est YES. Ne pas modifier.

**\_BPX\_SPAWN\_SCRIPT**

Exécute des scripts de shell directement à partir de la fonction spawn(). La valeur par défaut est YES. Ne pas modifier.

**PATH** Chemin d'accès de la commande. La valeur par défaut est \$JAVA\_HOME/bin:\$RSE\_LIB:\$CMDSEV\_BASE\_HOME/lib:\$PATH. Ne pas modifier.

**LIBPATH**

Chemin d'accès à la bibliothèque. La valeur par défaut est \$JAVA\_HOME/bin:\$JAVA\_HOME/bin/classic:\$ICU\_LIB:\$RSE\_LIB:... Ne pas modifier.

**CLASSPATH**

Chemin de classes. La valeur par défaut est trop longue pour être reprise. Ne pas modifier.

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

Options Java supplémentaires spécifiques au service de Commandes TSO. La valeur par défaut est "&SESSION=SPAWN\$\_RSE\_CMDSERV\_OPTS". Ne pas modifier.

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

Options Java supplémentaires spécifiques à l'explorateur de systèmes éloignés RSE. La valeur par défaut est trop longue pour être reprise. Ne pas modifier.

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

Classe Java pour le serveur RSE. La valeur par défaut est `com.ibm.etools.systems.dstore.core.server.Server`. Ne pas modifier.

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

Valeur de dépassement de délai pour le serveur RSE (attente du client) en millisecondes. La valeur par défaut est 120000 (2 minutes). Ne pas modifier.

**Remarque :** Vous pouvez ignorer la nécessité d'avoir les bibliothèques C/C++ et Language Environment (LE) dans LINKLIST en ajoutant l'instruction STEPLIB suivante à la fin (END) de `rsed.envvars` (les noms de fichiers peuvent être différents pour votre site). Soyez toutefois conscient que l'utilisation de STEPLIB dans z/OS UNIX présente un impact négatif sur les performances, voir «Éviter l'emploi de STEPLIB», à la page 73.

- Si la dernière directive STEPLIB définie plus tôt dans `rsed.envvars` est identique à STEPLIB=NONE  
STEPLIB=CEE.SCEERUN:CEE.SCEERUN2:CBC.SCLBDLL
- Si la dernière directive STEPLIB définie plus tôt dans `rsed.envvars` est n'est pas identique à STEPLIB=NONE  
STEPLIB=\$STEPLIB:CEE.SCEERUN:CEE.SCEERUN2:CBC.SCLBDLL

**Remarque :** Les liens symboliques sont autorisés pour l'indication des répertoires dans `rsed.envvars`.

## **(Facultatif) Définition de la gamme de ports disponible pour RSE**

Cette partie de la personnalisation de `rsed.envvars` spécifie les ports par lesquels le serveur RSE peut communiquer avec le client. Cette gamme de ports n'a aucune connexion avec les ports du démon RSE ou de REXEC/SSH.

Afin de mieux comprendre l'utilisation des ports, une brève description du processus de connexion RSE est incluse ci-après :

1. Le client se connecte au port hôte 4035, service démon INETD RSE, ou au port hôte 512, service INETD REXEC, ou au port hôte 22, service INETD SSH.
2. Le service INETD choisi crée un processus RSE.
3. Le processus RSE ouvre un port hôte pour que le client se connecte. Le choix de ce port peut être configuré par l'utilisateur, soit au niveau du client dans l'onglet des propriétés du sous-système (méthode non recommandée) soit par l'intermédiaire de la définition `_RSE_PORTRANGE` du fichier `rsed.envvars`.
4. Le service INETD renvoie le numéro de port au client.
5. Le client se connecte au port hôte.

Pour spécifier la gamme de ports permettant au client de communiquer avec z/OS, supprimez la mise en commentaire et modifiez la ligne suivante dans `rsed.envvars` :

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

**Remarque :** Avant de sélectionner une gamme de ports, vérifiez qu'elle est disponible sur votre système à l'aide des commandes **NETSTAT** et **NETSTAT PORTL**. Pour plus d'informations, voir «Ports TCP/IP réservés», à la page 91.

Le format du paramètre `PORTRANGE` est le suivant : `_RSE_PORTRANGE=min-max` (max est non inclusif ; par exemple, `_RSE_PORTRANGE=8108-8118` indique la plage des ports de 8108 à 8117 qui sont utilisables). Le numéro de port utilisé par le serveur RSE est déterminé en fonction des priorités suivantes :

1. Si un numéro de port différent de zéro est spécifié dans les propriétés de sous-système du client, ce numéro de port est utilisé. Si le port n'est pas disponible, la connexion échoue. Cette configuration n'est pas recommandée.
2. Si un numéro de port dans les propriétés de sous-système a pour valeur 0 et que `_RSE_PORTRANGE` est spécifié dans `rsed.envvars`, la gamme de ports spécifiée par `_RSE_PORTRANGE` est utilisée. Si aucun port de la gamme n'est disponible, la connexion échoue.
3. Si un numéro de port dans les propriétés de sous-système a pour valeur 0 et que `_RSE_PORTRANGE` n'est pas spécifié dans `rsed.envvars`, tout port disponible est utilisé.

**Remarque :** Lorsqu'un serveur ouvre un port et passe en mode écoute, le numéro de port ne peut pas être utilisé par un autre serveur, mais lorsque la connexion est établie, ce numéro de port est de nouveau utilisable. Ainsi, le nombre de ports de la gamme ne limite en rien le nombre d'utilisateurs susceptibles de se connecter simultanément.

## (Facultatif) Définition des paramètres de démarrage Java supplémentaires avec `_RSE_*OPTS`

Avec les différentes directives `_RSE_*OPTS`, `rsed.envvars` met à disposition la possibilité de donner des paramètres supplémentaires à Java lors du démarrage du serveur RSE.

Les exemples d'options inclus dans `rsed.envvars` peuvent être activés en supprimant la mise en commentaire.

### `_RSE_JAVAOPTS`

`_RSE_JAVAOPTS` définit les options standard et spécifiques à RSE Java.

```
_RSE_JAVAOPTS=""
```

Initialisation variable. Ne pas modifier.

```
#_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Xquickstart"
```

Améliore le délai du démarrage en retardant la compilation et les optimisations JIT. Ceci se fait au détriment des fichiers exécutables compilés légèrement moins efficaces, qui affectent les longues tâches d'exécution. Supprimez la mise en commentaire pour activer.

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

Définit la taille de pile initiale (Xms) et maximale (Xmx). Les valeurs

par défaut du système sont respectivement 1M et 64M. Supprimez la mise en commentaire et modifiez pour appliquer les valeurs de taille de pile indiquées.

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

Sélection de page de codes de l'hôte. Supprimer la mise en commentaire et modifier pour appliquer l'utilisation de la page de codes indiquée.

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

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

Option d'enregistrement du mot de passe. Supprimez la mise en commentaire pour empêcher les utilisateurs d'enregistrer leur mot de passe hôte sur le client. Les mots de passe enregistrés auparavant seront supprimés. Cette option ne fonctionne qu'avec des clients des versions 7.1 et ultérieures.

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

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

Démarre la fonction de trace dstore. A utiliser uniquement sur instruction du point service IBM.

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

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

Démarre la fonction de trace de mémoire dstore. A utiliser uniquement sur instruction du point service IBM.

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

Utilise APPC pour le service de Commandes TSO. Pour plus d'informations, voir «(Facultatif) Définition d'une transaction APPC pour le service Commandes TSO», à la page 26.

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

L'instruction `_RSE_CLASS_OPTS` définit les options Java 5.0 (et supérieure) nécessaires au partage de classes entre plusieurs serveurs RSE. Pour plus d'informations, voir «Partage de classes entre JVM», à la page 75.

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

Initialisation variable. Ne pas modifier.

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

Java version 5.0 et supérieure uniquement. Active le partage de classes. Supprimer la mise en commentaire pour partager des classes entre plusieurs serveurs RSE.

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

Java version 5.0 et supérieure uniquement. Définit la taille du cache des classes partagées. La valeur par défaut du système est 16M.

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

Les instructions `_RSE_CMDSERV_OPTS` sont des options Java spécifiques à RSE, qui prennent effet uniquement lorsque SCLM Developer Toolkit est utilisé comme serveur de Commandes TSO.

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

Initialisation variable. Ne pas modifier.

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

Utilise une profil ISPF existant pour le serveur de Commandes TSO. Supprimez la mise en commentaire et modifiez le nom du



fichier pour utiliser le profil ISPF indiqué. &SYSUID. peut être utilisé comme substitution à l'ID utilisateur du développeur.

---

## Configuration du démon INETD et de RSE REXEC/SSH

Developer for System z se repose sur le service INETD pour démarrer le serveur de l'Explorateur de systèmes éloignés (RSE) lorsqu'un client demande une connexion. INETD est un démon z/OS UNIX standard qui gère d'autres démons qui exécutent les tâches (dans ce cas, démarrer le serveur RSE). La configuration d'INETD ne fait pas partie de la personnalisation de Developer for System z, mais pour trouver des informations importantes, voir Annexe D, «Configuration d'INETD», à la page 109.

Developer for System z prend en charge différentes façons de démarrer le serveur RSE.

Vous devez personnaliser au moins une de ces méthodes selon le fonctionnement prévu pour les utilisateurs.

- Démon RSE, démarré en connectant INETD au port 4035 (par défaut). Cette méthode est recommandée, car elle offre davantage de contrôle et de sécurité à l'utilisateur que REXEC.
- Serveur de commandes REXEC (Remote Execution), port 512 par défaut. Il exécute un script de shell qui appelle l'Explorateur de systèmes éloignés RSE. Cette méthode de connexion s'adresse plutôt aux étapes de tests qu'à une utilisation sur le long terme. L'avantage de cette méthode est qu'elle nécessite moins de travail d'installation et de configuration, si REXEC a été configuré avant (configuration effectuée normalement au cours de la configuration du TCP/IP).
- Serveur de commandes SSH (Secure Shell), port 22 par défaut. Il exécute un script de shell qui appelle l'Explorateur de systèmes éloignés RSE. Cette méthode est comparable à la méthode utilisant REXEC, mais SSH utilise une communication sécurisée (chiffrée) avec le client.

**Remarque :** Les actions à distance (basées sur l'hôte) pour les sous-projets z/OS UNIX nécessitent que REXEC ou SSH soit actif sur l'hôte.

Vous pouvez vérifier qu'INETD fonctionne à l'aide de la commande **ps -e** (lancée par un utilisateur autorisé). La sortie doit contenir une référence à INETD, par exemple (# est l'invite de z/OS UNIX) :

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

**Remarque :** Pour que les serveurs z/OS UNIX (comme le démon RSE, REXEC et SSH) prennent en charge les connexions IPv6, tcp6 doit être indiqué pour le protocole du nom de service dans le fichier /etc/inetd.conf. Lorsque tcp6 est indiqué, les clients IPv4 sont également pris en charge. /etc/services prend uniquement en charge le mot clé tcp, sans numéro suffixe.

## Configuration du démon RSE INETD

1. Modifiez /etc/services en ajoutant cette ligne :  
rse 4035/tcp #Developer for System z RSE



**rse** nom de service du démon, par défaut rse (minuscule). Le nom doit correspondre à celui qui sera utilisé dans `/etc/inetd.conf`

#### **4035/tcp**

port et protocole utilisés, le port par défaut est 4035, le protocole doit être tcp.

Le port doit correspondre à celui qui a été défini chez le client lors de la création d'une nouvelle connexion z/OS.

**Remarque :** Avant de sélectionner un port, vérifiez qu'il est disponible sur votre système à l'aide des commandes **NETSTAT** et **NETSTAT PORTL**. Pour plus d'informations, voir «Ports TCP/IP réservés», à la page 91.

#### **#Developer for System z RSE**

commentaire, qui doit commencer par un dièse (#)

2. Modifiez `/etc/inetd.conf` en ajoutant ces deux lignes. Pour les règles de continuation, voir l'Annexe D, «Configuration d'INETD», à la page 109.

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

**rse** Nom de service du démon. La valeur par défaut est rse (minuscules). Le nom doit correspondre à celui utilisé dans `/etc/services`

#### **stream tcp nowait**

Instructions de configuration propres à INETD (type de socket, protocole, indicateur d'attente). Ne pas modifier.

**Remarque :** Utilisez le mot clé `tcp6` à la place de `tcp` pour prendre en charge des connexions IPv6.

#### **OMVSKERN**

ID utilisateur pour le processus démon RSE. La valeur par défaut est OMVSKERN. Cet ID utilisateur doit présenter un segment de sécurité OMVS valide, des droits d'accès BPX.DAEMON et des droits d'accès READ et EXECUTE aux répertoires d'installation et de configuration de Developer for System z. Pour plus d'informations sur les exigences concernant les ID utilisateur utilisés pour les services du système, voir Annexe D, «Configuration d'INETD», à la page 109.

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

Programme serveur (emplacement absolu de `fekfrsed`). La valeur par défaut est `/usr/lpp/wd4z/rse/lib/fekfrsed`.

Tout ce qui se trouve après cet argument INETD sont des arguments de serveur, commençant par le nom de serveur.

**rsed** Nom de serveur. Ne pas modifier.

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

Répertoire de travail (emplacement des fichiers de configuration du serveur RSE). La valeur par défaut est `/usr/lpp/wd4z/rse/lib`.

**Remarque :** Il est recommandé de copier les fichiers de configuration RSE personnalisés dans un nouveau répertoire (tel que `/etc/wd4z/`) pour éviter de les écraser lors d'une opération de maintenance. Le répertoire de travail défini ici doit refléter cette modification. Par exemple :

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

Les définitions suivantes sont facultatives. Si vous les omettez, les valeurs par défaut seront utilisées.

**-t 60** Option de dépassement de délai d'attente, pour indiquer le nombre de secondes durant lequel le démon RSE attend de recevoir la réponse du serveur RSE. La valeur par défaut est 60 secondes. Le dépassement de délai d'attente du client pour le serveur RSE est défini dans `rsed.envvars`. La valeur par défaut est de 2 minutes.

- INETD doit être redémarré par un utilisateur autorisé pour activer les modifications faites aux fichiers `/etc`, voir Annexe D, «Configuration d'INETD», à la page 109. Voir les modèles de commandes ci-après (# désigne l'invite z/OS UNIX) :

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

**Remarque :** Si le profil `BPX.DAEMON` est défini dans la classe `FACILITY` du produit de sécurité et que l'utilisateur qui (re)démarré INETD n'a pas accès à cette ressource, le message de sécurité suivant risque d'apparaître pour chaque client se connectant à RSE, dans lequel `IBMUSER` désigne l'ID utilisateur utilisé pour démarrer INETD.

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

## Configuration de INETD REXEC (ou SSH)

Aucune configuration propre à Developer for System z n'existe pour utiliser le serveur de commandes INETD REXEC (ou SSH). Toutefois, le client doit connaître 2 valeurs pour démarrer une connexion RSE par l'intermédiaire de REXEC/SSH :

- Le répertoire dans lequel se trouve le script de démarrage `server.zseries`. Par défaut, ce répertoire d'installation est `(/usr/lpp/wd4z/rse/lib/)`. Toutefois, `server.zseries` fait partie des fichiers qui doivent être également copiés si `rsed.envvars` est copié dans un répertoire différent, comme `/etc/wd4z/`.
- Le port qui est utilisé.

Un port commun utilisé par REXEC est 512. Pour une vérification rapide du port, utilisez la commande **NETSTAT**, comme dans l'exemple ci-après (\$ est l'invite de z/OS UNIX) :

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

Pour vérifier, vous pouvez contrôler `/etc/inetd.conf` et `/etc/services` afin de trouver le numéro de port utilisé.

- Trouvez le nom de service (1er mot, exec dans cet exemple) du serveur `rexecd` (7ème mot) dans `/etc/inetd.conf`  

```
exec stream tcp nowait OMVSKERN /usr/sbin/orexecd rexecd -LV
```
- Trouvez le port (2ème mot, 512 dans cet exemple) rattaché à ce nom de service (1er mot) dans `/etc/services/`  

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

Le même principe s'applique à SSH. Son port commun est 22, et le nom de serveur est `sshd`.

**Remarque :** Les actions à distance (basées sur l'hôte) pour les sous-projets z/OS UNIX nécessitent que REXEC ou SSH soit actif sur l'hôte. Si REXEC/SSH n'est pas configuré pour utiliser le port par défaut, le client de Developer for System z doit définir le port correct à utiliser avec les sous-projets z/OS UNIX. Pour cela, sélectionnez la page de préférences **Window > Préférences... > Solutions z/OS > Sous-projets USS > Options d'action distante**.

---

## Personnalisation d'ISPF.conf, fichier de configuration d'ISPF

Cette étape n'est obligatoire que dans le cas de l'utilisation de SCLM Developer Toolkit pour le service de Commandes TSO (cas par défaut). Elle n'est pas obligatoire dans le cas où APPC est utilisé pour le service de Commandes TSO.

SCLM Developer Toolkit nécessite que les définitions dans ISPF.conf créent un environnement valide pour exécuter les services ISPF. Le service Commandes TSO doit être ajouté à cet environnement ISPF.

ISPF.conf est créé lors de la personnalisation de SCLM Developer Toolkit, qui est décrite dans *SCLM Developer Toolkit Installation and Customization Guide (SC23-8504)*. L'emplacement par défaut est /etc/SCLMDT/CONFIG, mais il peut ne pas s'appliquer à votre site.

Ajoutez les lignes ci-après à ISPF.conf, dans lesquelles hlq est identique au qualificatif de haut niveau utilisé pour installer Developer for System z (la valeur par défaut est FEK).

```
*****
* Developer for System z – TSO Commands server
*****
sysexec=hlq.SFEKPROC
```

Le résultat doit ressembler à l'exemple, voir la figure 5.

```
sysproc=ISP.SISPCLIB
ispmlib=ISP.SISPMENU
isptlib=ISP.SISPTENU
ispplib=ISP.SISPPENU
ispslib=ISP.SISPSLIB
ispllib=BWB.SWBLOAD
*****
* Developer for System z – TSO Commands server
*****
sysexec=FEK.SFEKPROC
```

Figure 5. ISPF.conf - Fichier de configuration ISPF

**Remarque :** Si l'instruction sysexec est déjà définie, ajoutez-lui le fichier hlq.SFEKPROC à la fin, en séparant les noms de fichiers par une virgule (,).

---

## Vérification de la configuration du serveur RSE

L'installation de Developer for System z met à disposition plusieurs programmes de vérification de l'installation (IVP) pour le serveur RSE. Les scripts IVP se trouvent dans le répertoire d'installation, /usr/lpp/wd4z/rse/lib/ par défaut.

- fekfivp : «Connexion du service Commandes TSO (avec APPC)», à la page 57

- fekfivpc : «Connexion du service Commandes TSO (avec SCLMDT)», à la page 55
- fekfivpd : «Connexion du démon RSE», à la page 55
- fekfivpj : «Connexion du moniteur de travaux JES», à la page 55
- fekfivpr : «Connexion REXEC», à la page 53
- fekfivps : «Script de shell REXEC/SSH», à la page 54

Tous les exemples de commandes de la présente section sous-entendent que les variables de l'environnement RSE sont configurées. De cette manière, les scripts IVP sont disponibles par l'intermédiaire de l'instruction PATH, et l'emplacement de rsed.envvars est connu. Utilisez les commandes **pwd** et **cd** pour vérifier votre répertoire de travail, et en changer pour le répertoire dans lequel rsed.envvars est personnalisé. Le script de shell setup.env.zseries peut alors être utilisé pour configurer les variables d'environnement RSE, comme dans l'exemple ci-après (\$ est l'invite de z/OS UNIX) :

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

Le script de shell . ./setup.env.zseries, qui se trouve dans le même répertoire que rsed.envvars, exporte les variables d'environnement, de sorte que les autres processus peuvent les utiliser. Le premier "." (point) de . ./setup.env.zseries désigne une commande z/OS UNIX pour exécuter le shell dans l'environnement en cours, afin que les variables d'environnement définies dans le shell soient effectives même après avoir quitté ce dernier. Le second fait référence au répertoire de travail.

**Remarque :** Si . ./setup.env.zseries n'est pas exécuté avant les scripts fekfivp\*, le chemin d'accès à ces scripts doit être indiqué lorsqu'ils sont appelés, comme dans l'exemple ci-après :

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

La plupart des scripts fekfivp\* demanderont également l'emplacement du rsed.envvars personnalisé si . ./setup.env.zseries n'est pas exécuté en premier.

**Remarque :** Certains tests IVP utilisent l'API du socket TCP/IP REXX, qui nécessite que la bibliothèque de chargement TCP/IP, TCPIP.SEZALOAD par défaut, soit dans LINKLIST ou dans STEPLIB. Les commandes suivantes peuvent être nécessaires pour pouvoir exécuter ces tests IVP (\$ représente l'invite z/OS UNIX) :

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

ou

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

Pour plus d'informations concernant le diagnostic des incidents de connexion RSE, voir l'Annexe B, «Identification des incidents liés à la configuration», à la page 85 ou la page relative aux notes techniques de Developer for System z, <http://www-306.ibm.com/software/awdtools/devzseries/support/>.

## Disponibilité des ports

La disponibilité des ports du moniteur de travaux JES, de REXEC, de SSH et du démon RSE peut être vérifiée en émettant la commande **netstat**. Le résultat doit présenter les ports utilisés par ces services, comme dans les exemples ci-après (\$ est l'invite de z/OS UNIX) :

### 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
```

## Connexion REXEC

Testez la connexion REXEC en exécutant la commande ci-après. Remplacez 512 par le port utilisé par REXEC et USERID par un ID utilisateur valide.

```
fekfivpr 512 USERID
```

Après vous avoir invité à saisir un mot de passe, la commande doit renvoyer la trace REXEC, un avertissement de dépassement du délai d'attente, la version de Java et le message du serveur RSE, comme dans l'exemple ci-après (\$ est l'invite de z/OS UNIX) :

```
$ fekfivpr 512 USERID
Enter password:
$ EZYRC01I Calling function rexec_af with the following:
EZYRC02I Host: CDFMVS08, user USERID, cmd cd /etc/wd4z;export RSE_USER_ID=USERID;./server.zseries -ivp, port 512
EZYRC19I 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
```

**Remarque :** Si vous n'obtenez aucune sortie de serveur Java et RSE, il est probable que la taille de la région INETD est trop petite (elle doit être de 2096128 ou plus pour un lancement à partir d'une session shell TSO/OMVS, ou d'une taille de région 0 pour BPXBATCH).

**Remarque :** Vous pouvez tester séparément le script de shell utilisé par REXEC, comme le décrit le prochain test IVP, «Script de shell REXEC/SSH».

**Remarque :** Le serveur est démarré sans qu'un client tente de se connecter, par conséquent il va arriver au bout de son délai d'attente (au bout de 5 secondes). Il en résulte une trace de pile Java (25 lignes et plus) qui ressemble à l'exemple ci-après :

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

```

## Script de shell REXEC/SSH

Ce test IVP peut être évité si le test précédent décrit dans «Connexion REXEC», à la page 53, a été réussi.

Testez le script utilisé par la connexion REXEC et SSH en exécutant la commande ci-après :

```
fekfivps
```

La commande doit renvoyer un avertissement de dépassement du délai d'attente, la version de Java et le message du serveur RSE, comme dans l'exemple ci-après (\$ est l'invite de z/OS UNIX) :

```
$ 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 j9vmmz3123-20070201 (JIT
enabled)
J9VM - 20070131_11312_bHdSMr
JIT - 20070109_1805ifx1_r8
GC - 200701_09)
JCL - 20070126
```

```
Server Started Successfully
1751
Server running on: CDFMVS08$
```

**Remarque :** Si vous n'obtenez aucun résultat, la taille de votre région (TSO) est probablement trop petite (elle doit être 2096128 ou plus).

**Remarque :** Le serveur est démarré sans qu'un client tente de se connecter, par conséquent il va arriver au bout de son délai d'attente (au bout de 5 secondes). Il en résulte une trace de pile Java (25 lignes et plus) qui ressemble à l'échantillon ci-après :

```
$ 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.
...

```

## Connexion du démon RSE

Testez la connexion du démon RSE en exécutant la commande ci-après. Remplacez 4035 par le port utilisé par le démon RSE et USERID par un ID utilisateur valide.

```
fekfivpd 4035 USERID
```

Après vous avoir demandé un mot de passe, la commande doit renvoyer un résultat semblable à cet exemple (\$ est l'invite de z/OS UNIX) :

```
$ fekfivpd 4035 USERID
Password:
SSL is disabled
connected
8108
570655399
Succès

```

**Remarque :** Lors du test d'une connexion SSL activée, assurez-vous que avez indiqué le bon port si vous recevez le message d'erreur suivant :  
gsk\_secure\_socket\_init() failed: Socket closed by remote partner

## Connexion du moniteur de travaux JES

Testez la connexion du moniteur de travaux JES en exécutant la commande ci-après. Remplacez 6715 par le numéro de port du moniteur de travaux JES.

```
fekfivpj 6715
```

La commande doit renvoyer l'accusé de réception du moniteur de travaux JES, comme dans l'exemple ci-après (\$ est l'invite de z/OS UNIX) :

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

```

Succès

## Connexion du service Commandes TSO (avec SCLMDT)

Ce test IVP n'est nécessaire que dans le cas où vous utilisez SCLM Developer Toolkit, soit pour le service Commandes TSO soit pour le module d'extension client.

Testez la connexion au serveur Commandes TSO à l'aide de SCLM Developer en exécutant la commande ci-après.

```
fekfivpc
```

La commande doit renvoyer le résultat des vérifications relatives à SCLM Developer Toolkit (variables, modules HFS, environnement d'exécution REXX, démarrage et arrêt de session TSO/ISPF), comme dans l'exemple suivant (\$ est l'invite de z/OS UNIX) :

```
$ 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.SBWBLOAD

_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
-----
```

**Remarque :** Si l'une des vérifications SCLMDT échoue, de plus amples informations seront affichées.

fekfivpc présente plusieurs paramètres facultatifs ne dépendant pas de la position :

**-file** fekfivpc peut produire de grandes quantités en sortie (des centaines de lignes). Le paramètre -file envoie la sortie vers un fichier, home/.eclipse/RSE/USERID/fekfivpc.log, dans lequel home est le chemin d'accès vers l'accueil défini dans votre segment OMVS (ou le segment OMVS par défaut si vous n'avez pas de segment OMVS) et USERID est votre ID utilisateur (majuscule).

**-plugin**

Par défaut, fekfivpc vérifie uniquement les fonctions nécessaires au service



Commandes TSO. Le paramètre -plugin ajoute des tests supplémentaires pour les modules d'extension de client SCLMDT.

**-debug**

Le paramètre -debug crée une sortie détaillée des tests. N'utilisez pas cette option sans instruction du point service IBM.

## Connexion du service Commandes TSO (avec APPC)

N'exécutez pas cette procédure si vous n'avez pas configuré APPC pour le service Commandes TSO.

Testez la connexion au serveur Commandes TSO (en utilisant APPC) en exécutant la commande ci-après. Remplacez USERID par un ID utilisateur valide.

fekfivpa USERID

Après vous avoir demandé un mot de passe, la commande doit renvoyer la conversation APPC, comme dans l'exemple ci-après (\$ est l'invite de z/OS UNIX) :

```
$ 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
```

Pour plus d'informations concernant le diagnostic des incidents de connexion RSE, voir l'Annexe B, «Identification des incidents liés à la configuration», à la page 85 ou la page relative aux notes techniques de Developer for System z, <http://www-306.ibm.com/software/awdtools/devzseries/support/>.

---

## (Facultatif) Personnalisation de ssl.properties, configuration de la couche SSL du RSE

Toutes les méthodes de connexion client de Developer for System z utilisent les variables de la couche SSL (Secure Socket Layer) définies dans le fichier ssl.properties, situé par défaut dans le répertoire d'installation, /usr/lpp/wd4z/rse/lib/. Toutefois, ssl.properties fait partie des fichiers qui doivent être également copiés si rsed.envvars est copié dans un répertoire différent, comme /etc/wd4z/. Le modèle de fichier fourni comporte les instructions dont la liste figure dans la figure 6, à la page 58, où les lignes mises en commentaire commencent par un dièse (#).

```
# Specify this property as true to enable SSL
enable_ssl=false

#####
# Daemon Properties
# The key database file and password need to be specified for
# daemon to use.
# The key label need to be specified if not default key.
#daemon_keydb_file=
#daemon_keydb_password=
#daemon_key_label=

#####
# Server Properties
# The keystore file and password need to be specified for the
# server to use.
#server_keystore_file=
#server_keystore_password=
```

Figure 6. *ssl.properties* – Fichier de configuration RSE

Les propriétés du serveur et du démon doivent être configurées uniquement si vous activez la couche SSL. Pour plus d'informations sur la configuration de SSL, voir l'Annexe E, «Configuration du protocole SSL», à la page 117.

## (Facultatif) Personnalisation de *rsecomm.properties*, configuration de la fonction de trace RSE

Toutes les méthodes de connexion client de Developer for System z utilisent les variables définies dans le fichier *rsecomm.properties*, situé par défaut dans le répertoire d'installation, */usr/lpp/wd4z/rse/lib/*. Toutefois, *rsecomm.properties* fait partie des fichiers qui doivent être également copiés si *rsed.envvars* est copié dans un répertoire différent, comme */etc/wd4z/*. Le modèle de fichier fourni comporte les instructions dont la liste figure dans la figure 7, où les lignes mises en commentaire commencent par un dièse (#).

```
# server.version - DO NOT MODIFY!
server.version=5.0.0

# Logging level
# 0 - Log error messages
# 1 - Log error and warning messages
# 2 - Log error, warning and info messages
# 3 - Log error, warning, info and debug messages
debug_level=1

# Log location
# Log_To_StdOut
# Log_To_File
log_location=Log_To_File
```

Figure 7. *rsecomm.properties* – Fichier de configuration de consignment

Quand vous sélectionnez *log\_location=Log\_To\_File* (fichier par défaut), les informations sont consignées dans *home/.eclipse/RSE/USERID/rsecomm.log*, où *home* désigne le chemin d'accès vers l'accueil défini dans le segment OMVS de l'utilisateur (ou le segment OMVS par défaut si l'utilisateur n'en possède pas) et *USERID* correspond à l'ID utilisateur de connexion (majuscule).

**Remarque :** La définition `debug_level` contrôle également le niveau de consignation des autres fichiers journaux qui peuvent se trouver dans ce répertoire.

**Avertissement :** La modification de ces paramètres réduit les performances et ne doit être effectuée que sur indication du centre de support IBM.

---

## (Facultatif) Personnalisation de `projectcfg.properties`, configuration de projets hôte

Vous pouvez définir des projets z/OS individuellement via la perspective Projets z/OS du client ou de manière centrale sur l'hôte et les envoyer au client sur une base 'Par utilisateur'. Ces "projets basés sur l'hôte" ressemblent et fonctionnent exactement comme des projets définis sur le client à l'exception que leurs structure, membres et propriétés ne sont pas modifiables par le client et qu'ils sont accessibles uniquement lorsque ce dernier est connecté à l'hôte.

L'emplacement des définitions du projet est défini dans `projectcfg.properties`, situé par défaut dans le répertoire d'installation `/usr/lpp/wd4z/rse/lib/`. Toutefois, `projectcfg.properties` fait partie des fichiers qui doivent être également copiés si `rsed.envvars` est copié dans un répertoire différent, comme `/etc/wd4z/`.

Le modèle de fichier fourni comporte les instructions dont la liste figure dans la figure 8, où les lignes mises en commentaire commencent par un dièse (#).

```
#
# host based projects – root configuration file
#
# WSED-VERSION – do not modify !
WSED-VERSION=7.0.0.0
# specify the location of the host based project definition files
PROJECT-HOME=/var/wd4z/projects
```

Figure 8. `projectcfg.properties` – Fichier de configuration de projets basé sur l'hôte

La seule variable à modifier est `PROJECT-HOME`. Sa valeur, `/var/wd4z/projects` par défaut, correspond au répertoire principal des définitions du projet.

**Remarque :** Afin d'activer les projets basés sur l'hôte, un fichier `project.instance` doit exister dans `/var/wd4z/projects/USERID`, où `/var/wd4z/projects` désigne l'emplacement des fichiers de définitions du projet et `USERID` correspond à l'ID utilisateur qui permet au développeur de se connecter.

Pour plus d'informations sur les projets basés sur l'hôte, voir le livre blanc *Host Based Projects in WebSphere Developer for System z version 7.0* dans la bibliothèque internet Developer for System z, <http://www-306.ibm.com/software/awdtools/devzseries/library/>

---

## (Facultatif) Personnalisation de FMIEXT.properties, intégration de File Manager

Developer for System z prend en charge l'accès direct du client à un ensemble limité de fonctions d'IBM File Manager for z/OS. IBM File Manager for z/OS met à disposition des outils pour travailler avec les fichiers MVS, les fichiers z/OS UNIX, les données DB2, IMS et CICS. Ces outils comprennent les fonctionnalités habituelles de visualisation, d'édition, de copie et d'impression que l'on trouve dans ISPF, étendues afin de répondre aux besoins des développeurs d'applications. La version en cours de Developer for System z prend en charge uniquement la visualisation/édition des fichiers MVS (dont VSAM KSDS et ESDS).

Veuillez noter que le produit IBM File Manager for z/OS doit être commandé, installé et configuré séparément. Consultez *Rational Developer for System z - Guide de planification de l'hôte (SC11-2412-02)* pour savoir quel niveau de File Manager est nécessaire pour votre version de Developer for System z. L'installation et la personnalisation de ce produit ne sont pas décrites dans le présent ouvrage.

Les définitions nécessaires à File Manager Developer for System z sont stockées dans FMIEXT.properties, qui se trouve par défaut dans le répertoire d'installation, /usr/lpp/wd4z/rse/lib/. Toutefois, FMIEXT.properties fait partie des fichiers qui doivent être également copiés si rsed.envvars est copié dans un répertoire différent, comme /etc/wd4z/.

L'exemple de fichier fourni comporte les instructions dont la liste figure dans la figure 9, où les lignes mises en commentaire commencent par un dièse (#).

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

Figure 9. FMIEXT.properties – Fichier de configuration de File Manager

### startup.script

Emplacement absolu de fmiSub, le script de démarrage du serveur FMI. La valeur par défaut est /usr/lpp/wd4z/rse/lib/fmiSub.

### startup.port

Premier port employé pour la communication entre le serveur FMI et le serveur RSE, qui relaie l'information au client. Le port par défaut est 1957. La communication sur ce port ne concerne que la machine hôte.

**Remarque :** Avant de sélectionner un port, vérifiez qu'il est disponible sur votre système à l'aide des commandes **NETSTAT** et **NETSTAT PORTL**. Pour plus d'informations, voir «Ports TCP/IP réservés», à la page 91.

### startup.range

Gamme de ports, commençant par startup.port, qui sera utilisée pour la communication du serveur FMI. La valeur par défaut est 100. Par exemple,

quand vous utilisez les valeurs par défaut, les ports 1957 à 2056 (inclus) peuvent être utilisés par le serveur FMI.

**startup.fmload**

Emplacement absolu de la bibliothèque de chargement de File Manager. La valeur par défaut est FMN.SFMNMOD1. N'utilisez pas d'apostrophe (') pour rendre le nom du fichier absolu, aucun préfixe n'est ajouté.

**Remarque :** File Manager possède plusieurs bibliothèques de chargement. Celle qui doit être référencée dans le fichier de configuration est SFMNMOD1.

**startup.jobcard1**

**startup.jobcard2**

**startup.jobcard3**

Information de carte de travail pour le serveur FMI. Les valeurs par défaut sont //JOB CARD JOB <job parameters>, /\* et /\*. Le nom de travail sera remplacé par FEK<port> pour assurer son unicité.

**startup.sysout**

Classe SYSOUT pour le serveur FMI. La valeur par défaut est \*.



---

## Chapitre 5. (Facultatif) Activation d'IBM Common Access Repository Manager (CARMA)

Le gestionnaire CARMA (Common Access Repository Manager) (FMID: HCMA710) représente un gain de productivité pour les développeurs qui créent des API pour les gestionnaires de configuration de logiciel (SCM). A leur tour, ces API peuvent être utilisées par les applications (Developer for System z par exemple) pour accéder aux SCM.

Avant d'installer la version 7.1.1, si vous utilisiez la version précédente du gestionnaire CARMA, il est recommandé d'enregistrer la personnalisation associée comme décrit à la section «Sauvegarde des fichiers déjà configurés», à la page 11.

Après avoir installé CARMA, vous devez le configurer en procédant comme suit :

1. Configurez le serveur CARMA sur votre hôte z/OS (nécessite des interventions dans MVS z/OS UNIX)
2. (Facultatif) Configurez les échantillons de RAM
3. (Facultatif) Limitez l'accès des fichiers d'initialisation et des clusters VSAM.  
Dans la plupart des cas, seuls les administrateurs système et les développeurs de RAM CARMA ont besoin de l'accès en écriture à ces fichiers, tandis que l'accès en lecture suffit pour les autres utilisateurs.

**Remarque :** Les gestionnaires d'accès au référentiel (RAM) sont des API écrites par l'utilisateur permettant de faire l'interface avec les gestionnaires de configuration logicielle z/OS.

Voir le tableau 2, à la page 1 pour la liste des manuels qui proposent plus d'informations sur CARMA et son utilisation.

L'utilisateur peut contrôler la quantité d'informations de trace générées par CARMA en définissant le Niveau de trace dans l'onglet propriétés de la connexion CARMA du client. Les différents Niveaux de trace sont les suivants :

- Désactiver la journalisation
- Journalisation des erreurs
- Journalisation des avertissements
- Journalisation informative
- Déboguer la journalisation

La valeur par défaut est  
Journalisation des erreurs

Pour plus d'informations sur l'emplacement des fichiers journaux, voir l'«Emplacement des fichiers journaux», à la page 85.

---

### Personnalisation des composants du gestionnaire CARMA pour MVS

Toutes les références à hlq que vous trouverez dans cette section se rapportent au qualificatif de haut niveau utilisé au cours de l'installation de CARMA.  
L'installation par défaut est CRA, mais peut ne pas s'appliquer à votre site.

**Remarque :** Dans la version 7.1, de nouveaux messages ont été ajoutés à la méthode d'accès VSAM du message CARMA, CRAMSG. Il est conseillé de mettre à jour la précédente méthode d'accès VSAM. Une modification du nom de la mémoire RAM du squelette de la version 7.1 a été effectuée, ce qui entraîne la modification de la configuration de la méthode d'accès VSAM CARMA, CRADEF. La mise à jour de cette méthode d'accès VSAM est uniquement nécessaire si vous comptez utiliser la mémoire RAM du squelette.

Pour configurer l'hôte MVS, procédez comme suit :

1. Copiez les membres à personnaliser depuis le répertoire d'installation vers une bibliothèque personnelle et personnalisez les copies afin d'éviter de les écraser lors d'une opération de maintenance.
  - hlq.SCRACLST(CRASUBMT)
  - hlq.SCRASAMP(CRA\$VDEF)
  - hlq.SCRASAMP(CRA\$VMSG)
  - hlq.SCRASAMP(CRA\$VSTR)
2. Personnalisez la liste de commandes hlq.SCRACLST(CRASUBMT). Pour obtenir des instructions de personnalisation, consultez la documentation qui se trouve dans CRASUBMT. La liste de commandes CRASUBMT soumet un serveur CARMA.

**Remarque :** Vous pouvez modifier la valeur du délai d'attente du gestionnaire CARMA en modifiant la ligne PROC 1 PORT TIMEOUT(420) dans hlq.SCRACLST(CRASUBMT) CLIST. Cette valeur correspond au nombre de secondes que CARMA devra attendre avant de recevoir la commande suivante du client. Définir une valeur sur 0 entraîne l'application de la valeur du délai d'attente par défaut, actuellement 420 secondes (7 minutes).

3. Personnalisez et soumettez le langage JCL hlq.SCRASAMP(CRA\$VDEF). Pour obtenir des instructions de personnalisation, consultez la documentation qui se trouve dans CRA\$VDEF.

**Remarque :** Le langage JCL CRA\$VDEF permet de mettre à jour ultérieurement le cluster VSAM CRADEF (configuration de CARMA). Pour mettre à jour le cluster, dirigez l'instruction de définition de données INPUT vers le fichier séquentiel voulu, à la place de CRAINIT. Pour plus d'informations sur la définition de ce fichier séquentiel, voir *Rational Developer for System z Common Access Repository Manager Developer's Guide (SC23-7660)*.

4. Personnalisez et soumettez le langage JCL hlq.SCRASAMP(CRA\$VMSG). Pour obtenir des instructions de personnalisation, consultez la documentation qui se trouve dans CRA\$VMSG. CRA\$VMSG crée et amorce la méthode d'accès VSAM du message CARMA, CRAMSG.
5. Personnalisez et soumettez le langage JCL hlq.SCRASAMP(CRA\$VSTR). Pour obtenir des instructions de personnalisation, consultez la documentation qui se trouve dans CRA\$VSTR.

**Remarque :** Le langage JCL CRA\$VSTR permet de mettre à jour ultérieurement le cluster VSAM CRASTRS (information de personnalisation de CARMA). Pour mettre à jour le cluster, dirigez l'instruction de définition de données INPUT vers le fichier séquentiel voulu, à la place de CRASINIT. Pour plus d'informations sur la définition de ce fichier séquentiel, voir *Rational Developer for System z Common Access Repository Manager Developer's Guide (SC23-7660)*.



---

## Personnalisation des composants du gestionnaire CARMA pour z/OS UNIX

Si vous n'êtes pas familiarisé avec le système z/OS UNIX, il est conseillé de demander l'aide d'un administrateur z/OS UNIX ou autre UNIX expérimenté pour effectuer les tâches répertoriées dans cette section.

Les commandes z/OS UNIX nécessaires pour effectuer ces tâches sont décrites brièvement pour votre convenance. Pour plus d'informations sur ces commandes, consultez l'ouvrage *UNIX System Services Command Reference (SA22-7802)*, sauf indication contraire.

- Les tâches décrites ci-après nécessitent des actions de votre part dans le système z/OS UNIX. Vous pouvez les effectuer en lançant la commande TSO **OMVS**. Utilisez **exit** pour retourner à TSO.
- MVS permet de modifier les fichiers z/OS UNIX à l'aide de l'utilitaire ISPF via la commande **OEDIT**. Cette commande s'utilise sous TSO et OMVS.

Toutes les instructions de chemin d'accès `/usr/lpp/wd4z/` que vous trouverez dans cette section se rapportent au chemin utilisé au cours de l'installation de Developer for System z, et non pas du gestionnaire CARMA. Le chemin par défaut est `/usr/lpp/wd4z/`, mais il peut ne pas s'appliquer à votre site.

Suivez les étapes ci-après pour configurer les composants z/OS UNIX CARMA ajoutés lors de l'installation d'IBM Rational Developer for System z (FMID: HHOP710) :

1. Le fichier de configuration `CRASRV.properties` doit se trouver dans le même répertoire que le fichier personnalisé `rse.envvars`. Les deux fichiers se trouvent par défaut dans le répertoire d'installation (chemin d'accès par défaut `/usr/lpp/wd4z/rse/lib/`). Mais il est conseillé, voir l'«Enregistrement du fichier de configuration `rse.envvars` dans un autre répertoire», à la page 38, de les copier dans un autre répertoire afin d'éviter de les écraser lors d'une opération de maintenance. Dans les exemples de ce manuel, ce répertoire est `/etc/wd4z/`.

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

2. Le modèle de fichier de configuration `CRASRV.properties` comporte un ensemble de définitions de variables d'environnement. L'exemple de fichier de configuration doit être modifié pour se conformer aux normes de votre site, et comporte les instructions de la liste, voir la figure 10, où les lignes mises en commentaire commencent par un dièse (#).

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

Figure 10. *CRASRV.properties* – Fichier de configuration de CARMA

### **port.start**

Premier port utilisé pour la communication entre les composants CARMA MVS et z/OS UNIX. Le port par défaut est 5227. La communication sur ce port ne concerne que la machine hôte.

**Remarque :** Avant de sélectionner un port, vérifiez qu'il est disponible sur votre système à l'aide des commandes **NETSTAT** et

**NETSTAT PORTL.** Pour plus d'informations, voir le «Ports TCP/IP réservés», à la page 91.

**port.range**

Gamme de ports, commençant par `port.start`, qui sera utilisée pour la communication du serveur CARMA. La valeur par défaut est 100. Par exemple, quand vous utilisez les valeurs par défaut, les ports 5227 à 5326 (inclus) peuvent être utilisés par le gestionnaire CARMA.

**startup.script.name**

Définit le chemin d'accès absolu du script de soumission REXX `rexsub`. La valeur par défaut est `/usr/lpp/wd4z/rse/lib/rexsub`. Ce REXX exec va déclencher l'exécution de la liste de commandes `CRASUBMT` dans MVS.

**clist.dsname**

Définit l'emplacement de la liste de commandes `CRASUBMT`, en utilisant les conventions de référence MVS. Avec des apostrophes ('), il s'agit d'un emplacement absolu, sans apostrophes le préfixe de l'utilisateur précède le nom de fichier indiqué. La valeur par défaut est `'hlq.SCRACLST(CRASUBMT)'`. L'installation de CARMA SMP/E, qui crée `CRASUBMT`, utilise la valeur par défaut `CRA` pour `hlq`. La liste de commandes va démarrer un serveur CARMA à l'ouverture d'une connexion.

**Remarque :** Dans *Developer for System z* version 7.0, le nom de membre et le fichier de la liste de commandes ont été déplacés de `rexsub` (variable `DSNAME`) dans `CRASRV.properties`, supprimant la nécessité de personnaliser `rexsub`. Laissez `rexsub` dans le répertoire d'installation si vous souhaitez avoir la possibilité d'activer automatiquement une opération de maintenance SMP/E.

---

## (Facultatif) Activation des modèles de RAM (Repository Access Managers)

Les RAM (Repository Access Managers) sont des API écrites par l'utilisateur permettant d'interfacer avec les gestionnaires de configuration de logiciel z/OS. Pour activer les échantillons de RAM voulus, suivez les instructions des sections suivantes.

**Remarque :** Les échantillons de RAM permettent de tester la configuration de votre environnement CARMA et servent d'exemples pour le développement de votre propre RAM. NE PAS utiliser les modèles de RAM fournis dans un environnement de production.

**Remarque :** Toutes les références à `hlq` que vous trouverez dans cette section se rapportent au qualificatif de haut niveau utilisé au cours de l'installation de CARMA. L'installation par défaut est `CRA`, mais peut ne pas s'appliquer à votre site.

Pour plus d'informations sur les échantillons de RAM et les échantillons de code source fournis, voir *Rational Developer for System z Common Access Repository Manager Developer's Guide* (SC23-7660).

## Activation de la RAM SCLM

1. Copiez les membres à personnaliser depuis le répertoire d'installation vers une bibliothèque personnelle et personnalisez les copies afin d'éviter de les écraser lors d'une opération de maintenance.
  - hlq.SCRASAMP(CRA#VSLM)
  - hlq.SCRASAMP(CRA#ASLM)
  - hlq.SCRACLST(CRASUBMT)
2. Personnalisez et soumettez le langage JCL hlq.SCRASAMP(CRA#VSLM). Pour obtenir des instructions de personnalisation, consultez la documentation qui se trouve dans CRA#VSLM. CRA#VSLM crée et amorce la méthode d'accès VSAM du message de la RAM SCLM.
3. Supprimez la mise en commentaire de l'instruction CRARAM2 DD dans CRASUBMT et indiquez le nom de fichier de la méthode d'accès VSAM du message de la mémoire RAM SCLM. Notez que CRASUBMT a été personnalisée auparavant dans «Personnalisation des composants du gestionnaire CARMA pour MVS», à la page 63.
4. Personnalisez le langage JCL hlq.SCRASAMP(CRA#ASLM). Pour obtenir des instructions de personnalisation, consultez la documentation qui se trouve dans CRA#ASLM. CRA#ASLM attribue les fichiers nécessaires aux clients de la RAM SCLM.

**Remarque :** Chaque utilisateur doit soumettre CRA#ASLM une fois avant d'utiliser le gestionnaire CARMA avec la RAM SCLM. Sinon, une erreur d'attribution est engendrée.

## Activation de la RAM PDS

1. Copiez les membres à personnaliser depuis le répertoire d'installation vers une bibliothèque personnelle et personnalisez les copies afin d'éviter de les écraser lors d'une opération de maintenance.
  - hlq.SCRASAMP(CRA#VPDS)
  - hlq.SCRACLST(CRASUBMT)
2. Personnalisez et soumettez le langage JCL hlq.SCRASAMP(CRA#VPDS). Pour obtenir des instructions de personnalisation, consultez la documentation qui se trouve dans CRA#VPDS. CRA#VPDS crée et amorce la méthode d'accès VSAM du message de la RAM PDS.
3. 3. Supprimez la mise en commentaire de l'instruction CRARAM1 DD dans CRASUBMT et indiquez le nom de fichier de la méthode d'accès VSAM du message de la mémoire RAM PDS. Notez que CRASUBMT a été personnalisée auparavant dans «Personnalisation des composants du gestionnaire CARMA pour MVS», à la page 63.

## Activation de la RAM du squelette

1. Copiez les membres à personnaliser depuis le répertoire d'installation vers une bibliothèque personnelle et personnalisez les copies afin d'éviter de les écraser lors d'une opération de maintenance.
  - hlq.SCRASAMP(CRA#CRAM)
  - hlq.SCRACLST(CRASUBMT)
2. Personnalisez et soumettez le langage JCL hlq.SCRASAMP(CRA#CRAM). Pour obtenir des instructions de personnalisation, consultez la documentation qui se trouve dans CRA#CRAM. CRA#CRAM compile la RAM du squelette.

3. Ajoutez la bibliothèque de chargement qui contient le module compilé de la mémoire RAM du squelette, CRARAMSA, à l'instruction STEPLIB DD de CRASUBMT. Notez que CRASUBMT a été personnalisée auparavant dans «Personnalisation des composants du gestionnaire CARMA pour MVS», à la page 63.

---

## Chapitre 6. (Facultatif) Activation d'IBM Software Configuration and Library Manager (SCLM) Developer Toolkit

IBM Software Configuration and Library Manager (SCLM) Developer Toolkit (FMID: HSD3310) permet d'étendre les fonctions de SCLM au client. SCLM est lui-même un gestionnaire de code source hôte livré comme partie intégrante d'ISPF.

SCLM Developer Toolkit, qui est livré avec Developer for System z, est un module d'extension basé sur Eclipse qui procure une interface avec SCLM et permet d'accéder à tous les processus SCLM de développement du code existant et prend en charge le développement de Java et de J2EE complets sur le poste de travail avec la synchronisation à SCLM sur le grand système y compris la construction, l'assemblage et le déploiement du code J2EE du grand système.

Voir le tableau 2, à la page 1 pour la liste des manuels qui proposent plus d'informations sur SCLM Developer Toolkit et son installation, son utilisation et sa personnalisation.



## Chapitre 7. Remarques relatives au client Developer for System z

Les utilisateur du client Developer for System z doivent connaître les résultats de certaines personnalisations de l'hôte, telles que les numéros de port TCP/IP, pour que le client fonctionne correctement. Utilisez la liste de contrôle qui se trouve dans le tableau 12 pour rassembler les informations nécessaires.

Tableau 12. Liste de contrôle du client Developer for System z

Personnalisation	Valeur
Numéro de port de serveur du moniteur de travaux JES (par défaut 6715) :  Voir SERV_PORT dans «Personnalisation de FEJJCNFG, le fichier de configuration du moniteur de travaux JES», à la page 18.	
Emplacement des procédures ELAXF* si elles ne se trouvent pas dans une bibliothèque de procédure système.  Voir les annotations sur JCLLIB dans «Personnalisation ELAXF*, procédure de construction à distance», à la page 24.	
Procédure et/ou noms des étapes des procédures ELAXF* si elles ont été modifiées.  Voir les annotations sur leur modification dans «Personnalisation ELAXF*, procédure de construction à distance», à la page 24.	
Nom de la procédure mémorisée DB2 (ELAXMSAM par défaut) :  Voir les informations sur les procédures mémorisées DB2 dans l'Annexe A, «Exécution de plusieurs instances de Developer for System z», à la page 81.	
Emplacement de la procédure mémorisée DB2 si elle ne se trouve pas dans une bibliothèque de procédure système :  Voir «(Facultatif) Personnalisation ELAXM*, membres de procédure mémorisée DB2», à la page 29.	
Utilisation de la méthode de connexion DAEMON, REXEC ou SSH pour RSE :  Voir «Configuration du démon INETD et de RSE REXEC/SSH», à la page 48.	
Numéro de port TCP/IP du démon RSE (4035 par défaut) :  Voir «Configuration du démon RSE INETD», à la page 48.	
Chemin d'accès au script de shell server.zseries pour REXEC/SSH (/usr/lpp/wd4z/rse/lib par défaut, /etc/wd4z conseillé) :  Voir «Configuration de INETD REXEC (ou SSH)», à la page 50.	
Numéro de port REXEC ou SSH (512 ou 22 par défaut, respectivement) :  Voir «Configuration de INETD REXEC (ou SSH)», à la page 50. <b>Remarque :</b> Les actions à distance (basées sur l'hôte) pour les sous-projets z/OS UNIX nécessitent que REXEC ou SSH soit actif sur l'hôte.	

Tableau 12. Liste de contrôle du client Developer for System z (suite)

Personnalisation	Valeur
<p>Emplacement du langage JCL CRA#ASLM pour les allocations de fichiers RAM SCLM CARMA :</p> <p>Voir les annotations concernant CRA#ASLM dans «Activation de la RAM SCLM», à la page 67.</p>	



---

## Chapitre 8. Remarques relatives aux performances

z/OS est un système d'exploitation fortement personnalisable, et des modifications (parfois mineures) du système peuvent présenter un impact très important sur les performances globales. Le présent chapitre met en évidence certaines modifications qui peuvent être apportées afin d'améliorer les performances de Developer for System z.

Pour plus d'informations sur l'optimisation du système, voir *MVS Initialization and Tuning Guide (SA22-7591)* et *UNIX System Services Planning (GA22-7800)*.

---

### Utilisation du système de fichiers zFS

zFS (zSeries File System) et HFS (Hierarchical File System) sont tous deux des systèmes de fichiers UNIX pouvant être utilisés dans un environnement UNIX z/OS UNIX. Cependant, zFS fournit les fonctions et avantages suivants :

- Des gains de performances sont constatés dans nombre d'environnements de clients lors de l'accès à des fichiers approchant une taille de 8 Ko, fréquemment ouverts et mis à jour. Les performances d'accès aux fichiers de plus petite taille sont équivalentes à celles du système de fichiers HFS.
- Clonage en lecture seule d'un système de fichiers du même fichier. Le système de fichiers cloné peut être accessible aux utilisateurs afin de fournir une copie instantanée en lecture seule d'un système de fichiers. Cette fonction disponible uniquement dans un environnement non sysplex est facultative.
- zFS est le système de fichiers stratégique UNIX z/OS UNIX. La fonctionnalité HFS a été stabilisée, et des optimisations apportées au système de fichiers seront uniquement appliquées à zFS.

Pour plus d'informations concernant zFS, voir *UNIX System Services Planning (GA22-7800)*.

---

### Eviter l'emploi de STEPLIB

Chaque processus z/OS UNIX qui présente un STEPLIB qui est propagé de parent à enfant ou à travers une commande exec consomme environ 200 octets de zone de mémoire commune étendue ECSA. Dans le cas où aucune variable d'environnement STEPLIB n'est définie, ou dans le cas où elle est définie comme STEPLIB=CURRENT, z/OS UNIX propage toutes les allocations TASKLIB, STEPLIB et JOBLIB actuellement actives lors d'une fonction fork(), spawn(), ou exec(). Le serveur RSE lance plusieurs processus, et chaque connexion client dispose d'un serveur RSE privé. De ce fait, le nombre peut augmenter rapidement.

Developer for System z présente une valeur par défaut STEPLIB=NONE codée dans rsed.envvars, comme décrit dans «Personnalisation de rsed.envvars, le fichier de configuration de l'Explorateur de systèmes éloignés RSE», à la page 39. Pour les raisons mentionnées ci-avant, il est conseillé de ne pas modifier cette directive et de placer les fichiers ciblés dans LINKLIST ou dans la zone permanente de programme LPA à la place.

Si vous utilisez effectivement l'instruction STEPLIB, vous devez vérifier le contenu de rsed.envvars pour voir si votre instruction STEPLIB est oui ou non la première.

- Si la dernière directive STEPLIB définie plus tôt dans `rsed.envvars` est identique à `STEPLIB=NONE`  
`STEPLIB=first.steplib.dataset:second.steplib.dataset`
- Si la dernière directive STEPLIB définie plus tôt dans `rsed.envvars` n'est pas identique à `STEPLIB=NONE`  
`STEPLIB=$STEPLIB:first.steplib.dataset:second.steplib.dataset`

---

## Amélioration de l'accès aux bibliothèques du système

Certaines bibliothèques du système ainsi que certains modules de chargement sont fortement utilisés par z/OS UNIX et par les activités de développement d'applications. En améliorant leur accès, comme par leur ajout à la zone permanente de programme (LPA), il est possible d'améliorer les performances du système. Pour plus d'informations sur la modification des membres `SYS1.PARMLIB` décrits ci-après *MVS Initialization and Tuning Reference (SA22-7592)*.

### Bibliothèques d'exécution Language Environment (LE)

Lorsque des programmes C (y compris le shell z/OS UNIX) sont exécutés, ils utilisent fréquemment des routines issues de la bibliothèque d'exécution Language Environment (LE). En moyenne, environ 4 mégaoctets de bibliothèque d'exécution sont chargés dans la mémoire pour chaque espace adresse qui exécute un programme LE activé, et sont copiés à chaque fourche.

Le fichier `CEE.SCEELPA` contient un sous-ensemble de routines d'exécution LE, qui sont fortement utilisées par z/OS UNIX. Il est conseillé, pour gagner le maximum de performances, d'ajouter ce fichier à `SYS1.PARMLIB(LPALSTxx)`. De cette manière, les modules sont lus une seule fois sur le disque, et sont stockés dans un emplacement partagé.

**Remarque :** Ajoutez les instructions suivantes à `SYS1.PARMLIB(PROGxx)` si vous préférez ajouter les modules de chargement à la zone permanente de programme (LPA) dynamique.

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

Il est également conseillé de placer les bibliothèques d'exécution LE `CEE.SCEERUN` et `CEE.SCEERUN2` dans `LINKLIST`, en ajoutant les fichiers à `SYS1.PARMLIB(LNKLSTxx)` ou à `SYS1.PARMLIB(PROGxx)`. Cela élimine le temps système de z/OS UNIX `STEPLIB`, et il y a moins d'entrées/sorties en raison de la gestion par LLA et VLF ou par des produits similaires.

**Remarque :** Ajoutez également la bibliothèque de classe de la bibliothèque de chargement dynamique C/C++ `CBC.SCLBDLL` à `LINKLIST`, pour les mêmes raisons.

Si vous décidez de ne pas mettre ces bibliothèques dans `LINKLIST`, vous devez alors configurer l'instruction `STEPLIB` appropriée dans `rsed.envvars`, comme décrit dans «Personnalisation de `rsed.envvars`, le fichier de configuration de l'Explorateur de systèmes éloignés RSE», à la page 39. Même si cette méthode utilise toujours de la mémoire virtuelle supplémentaire, vous pouvez améliorer les performances en définissant les bibliothèques d'exécution LE pour LLA ou un produit similaire. Cela réduit les entrées/sorties nécessaires pour le chargement des modules.

## Développement d'application

Les performances des systèmes sur lesquels le développement d'applications est l'activité principale peuvent également être améliorées si mettez l'éditeur de liens sur LPA dynamique, en ajoutant les lignes ci-après à SYS1.PARMLIB(PROGxx) :

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

Pour le développement C/C++, vous pouvez également ajouter le fichier du compilateur CBC.SCCNCMP à SYS1.PARMLIB(LPALSTxx).

Les instructions ci-avant sont des exemples de candidats LPA possibles, mais les besoins de votre site peuvent être différents. Pour plus d'informations concernant l'insertion de modules de chargement LE dans le LPA dynamique, voir *Language Environment Customization (SA22-7564)*. Pour plus d'informations sur l'insertion de modules de chargement de compilateur C/C++ dans le LPA dynamique, voir *UNIX System Services Planning (GA22-7800)*.

---

## Amélioration des performances du contrôle d'autorisations d'accès

Afin d'améliorer les performances des contrôles d'autorisations d'accès effectués pour z/OS UNIX, définissez le profil BPX.SAFFASTPATH dans la classe FACILITY de votre logiciel de sécurité. Cela réduit le temps système des contrôles des droits d'accès de z/OS UNIX pour une large gamme d'opérations. Cette gamme comprend le contrôle de l'accès aux fichiers, le contrôle de l'accès aux communications interprocessus, et le contrôle de propriété de processus. Pour plus d'informations sur ce profil, voir *UNIX System Services Planning (GA22-7800)*.

**Remarque :** Les utilisateurs n'ont pas besoin d'avoir de permission pour le profil BPX.SAFFASTPATH.

---

## Partage de classes entre JVM

IBM Java Virtual Machine (JVM) version 5 et suivantes vous permet de partager entre JVM les amorces et les classes d'application par leur stockage dans un cache de la mémoire partagée. Le partage de classes réduit la consommation globale de mémoire virtuelle quand plusieurs JVM partagent un cache. Le partage de classes réduit également le temps de démarrage d'une JVM une fois que le cache a été créé.

Le cache de classes partagées est indépendant de toute JVM active et se conserve au-delà de la durée de vie de la JVM qui l'a créé. Dans la mesure où le cache de classes partagées se conserve au-delà de la durée de vie de toute JVM, le cache est mis à jour de façon dynamique afin de refléter toute modification qui peut avoir été apportée aux JAR ou aux classes sur le système de fichiers.

Le temps système pour créer et remplir un nouveau cache est minimal. Le coût de démarrage en temps d'une JVM unique est généralement entre 0 et 5 % plus lent que celui d'un système qui n'utilise pas le partage de classes, en fonction du nombre de classes qui est chargé. L'amélioration du temps de démarrage de JVM avec un cache rempli est généralement entre 10 % et 40 % plus rapide par rapport à celui d'un système n'utilisant pas le partage de classe, en fonction du système d'exploitation et du nombre de classes chargées. Plusieurs JVM exécutées en même temps présenteront des bénéfices plus importants en terme de temps de démarrage.

Pour plus d'informations sur le partage des classes, voir *Java SDK and Runtime Environment User Guide*.

## Activation du partage de classes

Pour activer le partage de classe pour le serveur RSE, supprimez la mise en commentaire de l'instruction suivante, dans `rse.envvars`, comme il est indiqué dans «(Facultatif) Définition des paramètres de démarrage Java supplémentaires avec `_RSE_*OPTS`», à la page 46. La première instruction définit un cache nommé RSE avec accès de groupe et permet au serveur RSE de démarrer même si le partage de classes échoue. La seconde instruction est facultative. Elle définit la taille du cache à 6 mégaoctets (la valeur par défaut du système étant 16 mégaoctets).

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

**Remarque :** Comme il est mentionné dans «Sécurité de la mémoire cache», tous les utilisateurs des classes partagées doivent avoir le même identificateur de groupe primaire (ID groupe). Cela signifie que les utilisateurs doivent avoir le même groupe par défaut défini dans le logiciel de sécurité, ou que des groupes par défaut différents présentent le même ID groupe dans leur segment OMVS.

## Limites de la taille de la mémoire cache

Le maximum théorique de la taille de la mémoire cache est de 2 gigaoctets. La taille de la mémoire cache que vous pouvez indiquer est limitée par la quantité de mémoire physique et par le fichier d'échange disponible pour le système. Dans la mesure où l'espace d'adresse virtuelle d'un processus est partagée entre la mémoire cache de classe partagée et la pile Java, l'augmentation de la taille maximale de la pile Java réduit la taille de la mémoire cache de classes partagées que vous pouvez créer.

## Sécurité de la mémoire cache

L'accès au cache de classes partagées est limité par les autorisation du système d'exploitation et par les autorisations de sécurité de Java.

Par défaut, les caches de classes sont créés avec une sécurité au niveau utilisateur, de sorte que seul l'utilisateur qui a créé le cache peut y avoir accès. Dans z/OS UNIX, l'option `groupAccess` donne accès à tous les utilisateurs du groupe primaire de l'utilisateur qui a créé le cache. Toutefois, quel que soit le niveau d'accès utilisé, un cache peut uniquement être détruit par l'utilisateur qui l'a créé, ou par le superutilisateur (UID 0).

Pour plus d'informations sur les options de sécurité supplémentaires lors de l'utilisation de Java SecurityManager, voir *Java SDK and Runtime Environment User Guide*.

## SYS1.PARMLIB(BPXPRMxx)

Certains paramètres de `SYS1.PARMLIB(BPXPRMxx)` affectent les performances des classes partagées. L'emploi des mauvais paramètres peut arrêter le fonctionnement des classes partagées. Ces paramètres peuvent également avoir un impact sur les performances. Pour plus d'informations concernant les implications en termes de performances et d'utilisation de ces paramètres, voir *MVS Initialization and Tuning*.

Reference (SA22-7592) et UNIX System Services Planning (GA22-7800). Les paramètres BPXPRMxx les plus significatifs qui affectent le fonctionnement des classes partagées sont :

- MAXSHAREPAGES, IPCSHMPAGES, IPCSHMMPAGES et IPCSHMNSEGS

Ces paramètres affectent la quantité de pages de mémoire partagée disponible pour la machine virtuelle Java. La taille de page partagée pour un service de système z/OS UNIX 31-bit est fixée à 4 kilooctets. Les classes partagées essaient de créer par défaut une mémoire cache de 16 mégaoctets. Définissez donc IPCSHMMPAGES à une valeur supérieure à 4096.

Si vous définissez une taille de cache en utilisant -Xscmx, la machine virtuelle Java arrondira la valeur au mégaoctet le plus proche. Vous devez en tenir compte lors de la définition de IPCSHMMPAGES sur le système.

- IPCSEMNIDS et IPCSEMNSEMS

Ces paramètres affectent la quantité de sémaphores disponibles pour les processus UNIX. Les classes partagées utilisent les sémaphores de communication interprocessus pour dialoguer entre les machines virtuelles Java.

## Espace disque

Le cache de classes partagées nécessite de l'espace disque pour stocker les informations d'identification concernant les caches qui existent sur le système. Ces informations sont stockées dans /tmp/javasharedresources. Si le répertoire des informations d'identification est effacé, la machine virtuelle Java ne peut plus identifier les classes partagées sur le système, et doit recréer le cache.

## Utilitaires de gestion de cache

La ligne de commande Java -Xshareclasses peut prendre un certain nombre d'options, dont certaines sont des utilitaires de gestion de cache. Trois d'entre-elles sont présentées dans l'exemple ci-après (\$ est l'invite z/OS UNIX). Pour une présentation complète des options de ligne de commande prises en charge, voir *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.
```

**Remarque :** Les utilitaires de cache effectuent les opérations nécessaires sur le cache indiqué sans démarrer la machine virtuelle Java, de sorte que le message “Could not create the Java virtual machine.” est normal.

**Remarque :** Un cache ne peut être détruit que dans le cas où toutes les machines virtuelles JAVA qui l'utilisent sont arrêtées et où l'utilisateur dispose d'autorisations suffisantes.

---

## Taille de pile Java fixe

Avec une pile à taille fixe, aucune expansion ou contraction de pile ne peut se produire, ce qui peut permettre des gains de performances significatifs dans certaines situations. Toutefois, l'emploi d'une pile de taille fixe n'est généralement pas une bonne idée, dans la mesure où elle retarde le démarrage de la récupération de place jusqu'au moment où la pile est pleine, ce qui est une tâche très importante. Elle augmente également le risque de fragmentation, qui nécessite une compression de la pile. Aussi, n'utilisez les piles à taille fixe qu'après avoir effectué des essais appropriés, ou sur instructions du point service IBM. Pour plus d'informations concernant les tailles de pile et la récupération de place, voir *Java Diagnostics Guide* (SC34-6650).

La taille de pile initiale par défaut d'une machine virtuelle z/OS Java (JVM) est 1 mégaoctet. La taille maximale est 64 mégaoctets. Les limites peuvent être établies avec les options de commandes Java `-Xms` (initiale) et `-Xmx` (maximale).

Dans Developer for System z, les options de ligne de commande Java sont définies dans la directive `_RSE_JAVA_OPTS` de `rsed.envvars`, comme décrit dans «(Facultatif) Définition des paramètres de démarrage Java supplémentaires avec `_RSE_*OPTS`», à la page 46.

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

---

## Gestion de la charge de travail

Chaque site présente des besoins spécifiques, et il est possible de personnaliser le système d'exploitation z/OS pour tirer le meilleur parti des ressources disponibles pour satisfaire ces besoins. Avec la gestion de charge de travail, vous pouvez définir des objectifs de performances et assigner une importance à chaque but. Vous définissez les buts en terme d'activités, et le système décide quelles ressources, comme le stockage ou l'unité centrale, doivent être allouées à la tâche pour qu'elle atteigne son but.

Les performances de Developer for System z peuvent être équilibrées en paramétrant les but adéquats pour ses processus. Vous trouverez des instructions générales ci-après.

- Attribuez une transaction APPC à un groupe de performances TSO
- Assignez le serveur RSE à un groupe de performances TSO, ou à un groupe juste en dessous si vous avez une utilisation importante de TSO.

Pour plus d'informations sur le sujet, voir *MVS Planning Workload Management* (SA22-7602).





---

## Annexe A. Exécution de plusieurs instances de Developer for System z

Parfois, vous pouvez avoir besoin de plusieurs instances de Developer for System z actives sur un même système, lors du test d'une mise à niveau, par exemple. Cependant, certaines ressources telles que les ports TCP/IP ne peuvent pas être partagées ; les paramètres par défaut ne sont donc pas toujours applicables. Consultez les informations de cette annexe afin de programmer la coexistence des différentes instances de Developer for System z, pour pouvoir ensuite les personnaliser à l'aide de ce guide de configuration.

Bien qu'il soit possible de partager certaines parties de Developer for System z entre 2 instances (ou plus), il est recommandé de NE PAS le faire, sauf si leurs niveaux de logiciel sont identiques et que les seules modifications se trouvent dans les membres de configuration. Developer for System z offre suffisamment de possibilités de personnalisation pour que plusieurs instances ne se chevauchent pas et nous vous recommandons vivement d'utiliser ces fonctions.

---

### Niveaux de logiciels identiques, fichiers de configuration différents

Dans un nombre limité de circonstances, vous pouvez partager tout sauf (certains) des composants personnalisables. La mise à disposition d'un accès non-SSL pour une utilisation sur site, et d'une communication encodée SSL pour une utilisation hors site est un exemple.

**Avertissement :** La configuration partagée NE peut PAS être utilisée de manière sûre pour tester la maintenance, ou effectuer une prévisualisation technique ou une nouvelle édition.

Pour configurer une autre instance d'une installation active de Developer for System z, suivez à nouveau les étapes de personnalisation pour les composants qui sont différents, en utilisant d'autres fichiers/répertoires/ports afin d'éviter un chevauchement avec la configuration en cours.

Dans l'exemple SSL mentionné ci-avant, les personnalisations du serveur RSE en cours peuvent être clonées, après quoi le clone de `ssl.properties` peut être mis à jour. Les personnalisations du système MVS (moniteur de travaux JES etc.) peuvent être partagées entre les instances SSL et les instances non-SSL. Il en suivrait les actions ci-après :

1. Création d'un nouveau répertoire destiné à contenir les membres de configuration SSL personnalisés
2. Copie des membres de configuration active dans ce répertoire
3. Mise à jour de la copie de `ssl.properties` pour fournir les informations relatives à SSL
4. Définition d'un nouveau numéro de port de démon RSE dans `/etc/services`
5. Définition d'un nouveau processus de démon RSE dans `/etc/inetd.conf`, avec le nouveau répertoire comme paramètre `-d`
6. Redémarrage d'INETD pour activer le nouveau démon RSE

---

## Toutes autres situations

Lorsque des modifications de code sont effectuées (maintenance, prévisualisation technique, nouvelle édition) ou que les modifications que vous effectuez sont complexes, il est conseillé de procéder à une autre installation de Developer for System z. La présente section décrit les points possibles de conflit entre différentes installations.

La liste ci-après décrit brièvement les éléments qui doivent (cela est fortement recommandé) être différents entre les instances de Developer for System z :

- Bibliothèques d'installation
- Port TCP/IP du moniteur de travaux JES, ainsi que son fichier de configuration FEJJC�FG
- Langage JCL d'initialisation du moniteur de travaux JES
- Nom de transaction APPC
- Fichier de configuration RSE, rsed.envvars
- Port TCP/IP du démon RSE

Vous trouverez ci-après une présentation plus détaillée.

- Bibliothèques d'installation
  1. Installez chaque instance de Developer for System z dans des fichiers et des répertoires différents. Notez que vous pouvez uniquement modifier le chemin d'accès z/OS UNIX en indiquant un préfixe au paramètre par défaut IBM de /usr/lpp/wd4z. Voici un exemple valide : /service/usr/lpp/wd4z.
- Composants obligatoires
  1. Le fichier de configuration du moniteur de travaux JES hlq.SFEKSAMP(FEJJC�FG) contient le numéro de port TCP/IP du moniteur de travaux JES et par conséquent, ne peut pas être partagé. Le membre lui-même peut être renommé (si le langage JCL est également mis à jour), afin de pouvoir placer toutes les versions personnalisées de ce membre dans un même fichier, si vous n'effectuez pas les mises à jour dans le fichier d'installation.
  2. Le langage JCL du moniteur de travaux JES hlq.SFEKSAMP(FEJJJCL) se rapporte à FEJJC�FG ; par conséquent, il ne peut pas non plus être partagé. Après avoir renommé le membre (et la carte de travail), vous pouvez placer tous ceux du langage JCL dans le même fichier.
  3. La transaction APPC contient une référence à hlq.SFEKPROC(FEKFRSRV), le serveur de commandes TSO. Elle est propre au niveau du logiciel, par conséquent, vous devez créer une transaction APPC par instance. Notez que depuis la version 7.1, le serveur de commandes TSO est également accessible via une fonction SCLM Developer Toolkit.
  4. Le fichier de configuration (SCLM Developer Toolkit) ISPF.conf contient une référence à hlq.SFEKPROC(FEKFRSRV), le serveur de commandes TSO. Elle est propre au niveau du logiciel, par conséquent, vous devez créer un fichier ISPF.conf par instance. Notez que cela n'est nécessaire que dans le cas où vous utilisez SCLM Developer Toolkit pour accéder au serveur de commandes TSO.
  5. Le fichier de configuration /usr/lpp/wd4z/rse/lib/rsed.envvars contient des références au chemin d'installation et au nom de transaction APPC, qui doit être unique. Le nom de fichier est obligatoire, vous ne pouvez donc pas

conserver les différentes copies dans le même répertoire. Le nom de la transaction APPC ayant été modifié, la variable `_FEKFSCMD_TP_NAME_` doit être définie sur `rsed.envvars`.

6. Aucun des fichiers de configuration et des scripts de shell obligatoires qui doivent séjourner dans le même répertoire que `rsed.envvars` ne peuvent être partagés, dans la mesure où `rsed.envvars` doit être dans un emplacement non partagé.
7. Les ports TCP/IP REXEC et SSH peuvent être partagés sans aucune restriction.
8. Le démon RSE ne peut pas être partagé puisque chaque démon RSE dispose de son propre port TCP/IP. Vous devez créer un nouveau démon, avec un port et un nom de service différents.

- Composants facultatifs

1. Certaines procédures ELAXF\* font référence à `hlq.SFEKLOAD`, la bibliothèque de chargement de Developer for System z. Pour savoir comment créer différents ensembles qui soient disponibles pour les utilisateurs, consultez les annotations sur JCLLIB dans «Personnalisation ELAXF\*, procédure de construction à distance», à la page 24.
2. Pour activer deux instances de la procédure mémorisée DB2, vous devez procéder comme suit. Notez toutefois qu'il s'agit d'une description présentée en l'état, non prise en charge.
  - a. Copiez `hlq.SFEKPROC(ELAXMREX)` dans un membre appelé différemment, par exemple, `ELAXMRXX`
  - b. Copiez le modèle de membre `hlq.SFEKSAMP(ELAXMSAM)` dans un membre appelé différemment, par exemple, `ELAXMWDZ`
  - c. Modifiez le modèle de membre `hlq.SFEKSAMP(ELAXMJCL)` pour refléter ces modifications de dénomination, par exemple :

```
//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. Poursuivez la personnalisation décrite dans «(Facultatif) Personnalisation ELAXM\*, membres de procédure mémorisée DB2», à la page 29, mais avec les nouveaux membres.
- e. Le nouveau nom de l'environnement WLM (`ELAXMWDZ` par exemple) doit être utilisé dans l'assistant de la procédure mémorisée DB2 du client.
3. `bidi` dépend d'un membre de bibliothèque de chargement et par conséquent n'est pas partageable entre les éditions. Toutefois, si le nom du module de chargement est identique pour toutes les instances, vous pouvez partager la version la plus récente entre les instances, et même entre les éditions. La compatibilité amont est indisponible si le nom du module de chargement a été modifié.

4. Les modules de chargement ADM présentent une compatibilité amont, et la version la plus récente peut ainsi être partagée entre les éditions.
5. La méthode d'accès VSAM CRD ADM présente une compatibilité amont, et la version la plus récente peut ainsi être partagée entre les éditions.
6. Les définitions de ressource CICS ADM présentent une compatibilité amont, et la version la plus récente peut ainsi être partagée entre les éditions.
7. Aucun des fichiers de configuration et des scripts de shell obligatoires qui doivent séjourner dans le même répertoire que `rsed.envvars` ne peuvent être partagés, dans la mesure où `rsed.envvars` doit être dans un emplacement non partagé.
8. Il est possible dans `rsed.envvars` d'activer le partage de classes Java entre plusieurs serveurs RSE, en mettant à jour la directive `_RSE_JAVA_OPTS`. Le groupe de partage doit avoir un nom unique (`-Xshareclasses:name=RSE`).
9. Les méthodes d'accès VSAM CARMA peuvent changer d'un niveau de logiciel à l'autre ; nous vous recommandons par conséquent de ne pas les partager.

---

## Annexe B. Identification des incidents liés à la configuration

Cette annexe vous aide à résoudre certains incidents fréquents qui peuvent survenir lors de la configuration de Developer for System z.

Pour plus d'informations, consultez la section Support du site Developer for System z (<http://www-306.ibm.com/software/awdtools/devzseries/support/>) où vous trouverez des fiches techniques pour bénéficier des plus récentes informations produites par notre équipe de support.

Dans la section Bibliothèque du site Web, vous pouvez également consulter la dernière version de la documentation de Developer for System z, notamment les livres blancs.

La Bibliothèque z/OS en ligne contient également des informations importantes, que vous pouvez consulter à l'adresse suivante <http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/>

---

### Emplacement des fichiers journaux

Developer for System z crée des fichiers journaux utiles pour vous et pour le point service IBM dans l'identification et la résolution des incidents. La liste suivante présente les fichiers journaux que vous pouvez créer. Après ces journaux qui sont propres au produit, vérifiez bien le SYSLOG de tous les messages associés.

Les fichiers journaux basés sur le système MVS peuvent être localisés par l'intermédiaire de l'instruction de définition de données appropriée. Les fichiers journaux basés sur z/OS UNIX sont situés dans les répertoires suivants :

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

La plupart des fichiers journaux sont situés dans `home/.eclipse/RSE/USERID/`, où `home` désigne le chemin d'accès à l'accueil défini dans le segment OMVS de l'utilisateur (ou le segment OMVS par défaut si l'utilisateur n'en possède pas) et `USERID` correspond à l'ID utilisateur de connexion (majuscule).

- `daemon.log` - Journal du démon
- `fa.log` - Journal d'intégration de Fault Analyzer
- `fekfivpc.log` - Journal du test du programme de vérification d'installation `fekfivpc`
- `ffs.log` - Journal du serveur FFS, qui exécute des fonctions natives
- `ffsget.log` - Journal du programme de lecture de fichier, qui lit un fichier séquentiel ou un membre d'un fichier partitionné
- `ffsput.log` - Journal du programme d'écriture de fichier, qui écrit un fichier séquentiel ou un membre d'un fichier partitionné
- `lock.log` - Journal du gestionnaire de verrouillage, qui verrouille/déverrouille un fichier séquentiel ou un membre d'un fichier partitionné
- `rmt_class_loader.cache.jar` - Mémoire cache des classes chargées par le programme de chargement de classe distante RSE
- `rsecomm.log` - Journal du serveur RSE, qui traite les commandes du client (peut contenir une trace de pile d'exception Java)
- `stderr.log` - Données réacheminées de `stderr`, la sortie d'erreur standard

- stdout.log - Données réacheminées de stdout, la sortie standard
- /tmp/  
/tmp/ contient les fichiers journaux créés avant la vérification de l’ID utilisateur.
- rsedaemon.log - Journal du démon avant connexion

## Journalisation du moniteur de travaux JES

- **SYSOUT DD**  
Journalisation des opérations classiques. La valeur par défaut du modèle JCL hlq.SFEKSAMP(FEJJJCL) est définie par SYSOUT=\*.
- **SYSPRINT DD**  
Journalisation de trace. La valeur par défaut du modèle JCL hlq.SFEKSAMP(FEJJJCL) est définie par SYSOUT=\*. La fonction de trace est activée à l’aide du paramètre –TV, voir «Personnalisation du Langage JCL d’initialisation du moniteur de travaux JES», à la page 20 pour des informations détaillées.

## Journalisation de la transaction APPC (service Commandes TSO)

- **SYSPRINT DD**  
Quand l’utilitaire d’administration APPC ajoute et modifie un profil de programme transactionnel, il vérifie que le profil du programme transactionnel et son JCL ne contiennent pas d’erreurs de syntaxe. Le résultat de cette phase se compose de messages d’erreur de syntaxe de profil du programme transactionnel, de messages de traitement de l’utilitaire et d’instructions de conversion JCL. La journalisation des messages de cette phase est contrôlée par l’instruction SYSPRINT DD pour l’utilitaire ATBDSFMU. La valeur par défaut du modèle JCL hlq.SFEKSAMP(FEKAPPCC) est définie par SYSOUT=\*. Pour des informations détaillées, voir *MVS Planning APPC/MVS Management (SA22-7599)*.
- **&SYSUID.FEKFRSRV.&TPDATE.&TPTIME.LOG**  
Quand un programme transactionnel s’exécute, les messages d’exécution associés, tels que les messages d’allocation et de fin, sont envoyés vers un journal appelé par le mot clé MESSAGE\_DATA\_SET dans le profil du programme transactionnel. La valeur par défaut du modèle JCL hlq.SFEKSAMP(FEKAPPCC) est définie par &SYSUID.FEKFRSRV.&TPDATE.&TPTIME.LOG. Pour plus d’informations, voir *MVS Planning APPC/MVS Management (SA22-7599)*.

**Remarque :** En fonction de vos définitions de transaction APPC et des paramètres par défaut du site, ce fichier journal risque de ne pas apparaître sauf si le mot clé KEEP\_MESSAGE\_LOG(ALWAYS) est ajouté aux définitions de transaction. Pour plus d’informations, consultez le document *MVS Planning APPC/MVS Management (SA22-7599)*.

## Journalisation du serveur RSE

- **home/.eclipse/RSE/USERID/**  
Plusieurs fichiers journaux créés par les composants associés à RSE existent. La plupart sont situés dans home/.eclipse/RSE/USERID/, où home désigne le chemin d’accès à l’accueil défini dans le segment OMVS de l’utilisateur (ou le segment OMVS par défaut si l’utilisateur n’en possède pas) et USERID correspond à l’ID utilisateur de connexion (majuscule).
- daemon.log - Journal du démon
- ffs.log - Journal du serveur FFS, qui exécute des fonctions MVS natives

- `ffsget.log` - Journal du programme de lecture de fichier, qui lit un fichier séquentiel ou un membre d'un fichier partitionné
- `ffsput.log` - Journal du programme d'écriture de fichier, qui écrit un fichier séquentiel ou un membre d'un fichier partitionné
- `lock.log` - Journal du gestionnaire de verrouillage, qui verrouille/déverrouille un fichier séquentiel ou un membre d'un fichier partitionné
- `rmt_class_loader.cache.jar` - Mémoire cache des classes chargées par le programme de chargement de classe distante RSE
- `rsecomm.log` - Journal du serveur RSE, qui traite les commandes du client (peut contenir une trace de pile d'exception Java)
- `stderr.log` - Données réacheminées de `stderr`, la sortie d'erreur standard.
- `stdout.log` - Données réacheminées de `stdout`, la sortie standard

Le volume des données écrites dans `ffs*.log`, `lock.log` et `rsecomm.log` est contrôlé par le paramètre `debug_level` dans `rsecomm.properties`. Pour plus d'informations, voir «(Facultatif) Personnalisation de `rsecomm.properties`, configuration de la fonction de trace RSE», à la page 58.

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

Ce fichier contient un journal du démon avant la journalisation.

## Consignation des tests du programme IVP `fekfivpc`

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

Sortie de la commande `fekfivpc -file` (test du programme IVP associé à SCLM Developer Toolkit), où `home` désigne le chemin d'accès à l'accueil défini dans le segment OMVS de l'utilisateur (ou le segment OMVS par défaut si l'utilisateur n'en possède pas) et `USERID` correspond à l'ID utilisateur de connexion (majuscule).

Pour plus d'informations, voir «Connexion du service Commandes TSO (avec SCLMDT)», à la page 55.

## Journal d'intégration de Fault Analyzer

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

Journal d'intégration de Fault Analyzer, où `home` désigne le chemin d'accès à l'accueil défini dans le segment OMVS de l'utilisateur (ou le segment OMVS par défaut si l'utilisateur n'en possède pas) et `USERID` correspond à l'ID utilisateur de connexion (majuscule).

## Journal d'intégration de File Manager

- **Travail de serveur d'intégration de File Manager**

A l'ouverture d'une connexion avec File Manager, un travail de serveur est lancé, avec l'ID utilisateur de l'utilisateur comme propriétaire. Son nom est `FEKport`, où `port` désigne le port TCP/IP utilisé.

- **Définition de données SYSPRINT**

Le SYSPRINT du travail de serveur contient le journal du serveur FMI.

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

Journal de communication de FMI, où `home` désigne le chemin d'accès à l'accueil défini dans le segment OMVS de l'utilisateur (ou le segment OMVS par défaut si l'utilisateur n'en possède pas) et `USERID` correspond à l'ID utilisateur de connexion (majuscule).



Le volume des données écrites dans ce fichier est contrôlé par le paramètre `debug_level` dans `rsecomm.properties`. Pour plus d'informations, voir «(Facultatif) Personnalisation de `rsecomm.properties`, configuration de la fonction de trace RSE», à la page 58.

## Journal du gestionnaire CARMA

- Travail de serveur CARMA

Quand vous ouvrez une connexion avec CARMA, `hlq.SCRACLST(SCRASUBMT)` démarre un travail de serveur (avec les ID des utilisateurs en tant que propriétaires) appelé `CRAport`, où `port` désigne le port TCP/IP utilisé.

- Définition de données CARMALOG

Si l'instruction de définition de données CARMALOG est indiquée dans `hlq.SCRACLST(SCRASUBMT)`, le journal de CARMA est réacheminé vers cette instruction de définition de données dans le travail de serveur. Dans le cas contraire, elle est dirigée vers `SYSPRINT`.

Le volume de consignation du journal est commandé par le réglage Niveau de trace sur le client. Pour plus d'informations sur la configuration du paramètre Niveau de trace, voir Chapitre 5, «(Facultatif) Activation d'IBM Common Access Repository Manager (CARMA)», à la page 63.

- **SYSPRINT DD**

Le `SYSPRINT` du travail de serveur contient le journal du serveur CARMA, dans le cas où l'instruction de définition de données CARMALOG n'est pas définie.

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

Journal de communication de CARMA, où `home` désigne le chemin d'accès à l'accueil défini dans le segment OMVS de l'utilisateur (ou le segment OMVS par défaut si l'utilisateur n'en possède pas) et `USERID` correspond à l'ID utilisateur de connexion (majuscule).

Le volume des données écrites dans ce fichier est contrôlé par le paramètre `debug_level` dans `rsecomm.properties`. Pour plus d'informations, voir «(Facultatif) Personnalisation de `rsecomm.properties`, configuration de la fonction de trace RSE», à la page 58.

---

## Fichiers de vidage

Quand un produit se termine par une fin anormale, un vidage mémoire est créé pour aider à l'identification de l'incident. La disponibilité et l'emplacement de ces fichiers de vidage dépendent beaucoup des paramètres spécifiques du site. Par conséquent, ils peuvent ne pas être créés, ou être créés à des emplacements différents que ceux mentionnés ci-dessous.

## Fichiers de vidage MVS

Quand le programme fonctionne sous MVS, vérifiez les fichiers de vidage système ainsi que votre JCL pour les instructions de définition de données suivantes (selon le produit) :

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



Pour plus d'informations sur ces instructions de définition de données, voir *MVS JCL Reference (SA22-7597)* et *Language Environment Debugging Guide (GA22-7560)*.

## Fichiers de vidage Java

Dans z/OS UNIX, les fichiers de vidage de Developer for System z sont commandés par la machine virtuelle Java (JVM).

La JVM crée un ensemble d'agents de vidage par défaut lors de son initialisation (SYSTDUMP et JAVADUMP). Vous pouvez changer cet ensemble d'agents de vidage à l'aide de la variable d'environnement `JAVA_DUMP_OPTS` et même changer l'ensemble à l'aide de `-Xdump` sur la ligne de commande. Les options de ligne de commande de la JVM sont définies dans la directive `_RSE_JAVA0PTS` de `rsed.envvars`. Ne modifiez pas les paramètres de vidage sans instruction du point service IBM.

**Remarque :** Vous pouvez utiliser l'option `-Xdump:what` sur la ligne de commande pour déterminer quels agents de vidage existent à l'exécution du démarrage.

Les types de vidage qui peuvent être produits sont :

### SYSTDUMP

Cliché de transaction Java. Vidage mémoire non formaté généré par z/OS.

Le vidage est inscrit dans un fichier séquentiel MVS, avec un nom par défaut au format `&idutilisateur.JVM.TDUMP.&nomdetravail.D&date.H&heure`, ou selon la configuration de la variable d'environnement `JAVA_DUMP_TDUMP_PATTERN`. Si vous ne désirez pas que des clichés de transaction soient réalisés, ajoutez la variable d'environnement `IBM_JAVA_ZOS_TDUMP=NO` à `rsed.envvars`.

### CEEDUMP

Fichier de vidage Language Environment (LE). Récapitulatif formaté de vidage système qui montre les traces de pile pour chaque unité d'exécution du processus JVM, avec les informations de registre et un stockage de vidage court pour chaque registre.

Le vidage est inscrit dans un fichier z/OS UNIX nommé `CEEDUMP.yyyymmdd.hhmmss.pid`, où `yyymmdd` est la date du jour, `hhmmss` est l'heure et `pid` est l'ID processus en cours. Les emplacements possibles de ce fichier sont décrits dans «Emplacements de vidage de z/OS UNIX», à la page 90.

### HEAPDUMP

Vidage formaté (liste) des objets qui sont sur le tas Java.

Le vidage est inscrit dans un fichier z/OS UNIX nommé `HEAPDUMP.yyyymmdd.hhmmss.pid.TXT`, où `yyymmdd` est la date du jour, `hhmmss` est l'heure et `pid` est l'ID processus en cours. Les emplacements possibles de ce fichier sont décrits dans «Emplacements de vidage de z/OS UNIX», à la page 90.

### JAVADUMP

Analyse formatée de la JVM. Contient des données de diagnostic relatives à la JVM et à l'application Java, comme l'environnement d'application, les unités d'exécution, les piles natives, les verrous et la mémoire.

Le vidage est inscrit dans un fichier z/OS UNIX nommé `JAVADUMP.yyyymmdd.hhmmss.pid.TXT`, où `yyymmdd` est la date du jour, `hhmmss`

est l'heure et pid est l'ID processus en cours. Les emplacements possibles de ce fichier sont décrits dans «Emplacements de vidage de z/OS UNIX».

Pour plus d'informations sur les vidages de la JVM, voir *Java Diagnostics Guide* (SC34-6650), et pour les informations spécifiques à LE, voir *Language Environment Debugging Guide* (GA22-7560).

## Emplacements de vidage de z/OS UNIX

La machine virtuelle Java vérifie l'existence et les droits d'accès en écriture pour chacun des emplacements suivants, et stocke les fichiers CEEDUMP, HEAPDUMP et JAVADUMP dans le premier emplacement disponible. Veuillez noter que vous devez disposer d'un espace disque suffisant pour que le fichier de vidage soit écrit correctement.

1. Le répertoire dans la variable d'environnement `_CEE_DMPTARG`, s'il est trouvé. Cette variable est définie dans `rzed.envvars` en tant que `home/.eclipse/RSE/USERID/`, où `home` désigne le chemin d'accès à l'accueil défini dans le segment OMVS de l'utilisateur (ou le segment OMVS par défaut si l'utilisateur n'en possède pas) et `USERID` correspond à l'ID utilisateur de connexion (majuscule).
2. Le répertoire de travail en cours, s'il ne s'agit pas du répertoire principal (`/`), et qu'il est inscriptible.
3. Le répertoire dans la variable d'environnement `TMPDIR` (une variable d'environnement qui indique l'emplacement d'un répertoire temporaire autre que `/tmp`), s'il est trouvé.
4. Le répertoire `/tmp`.
5. Si le vidage ne peut être stocké dans aucun des emplacements ci-avant, il est mis dans `stderr`.

---

## Autorisation de contrôle par un programme pour les programmes RSE

L'Explorateur de systèmes éloignés (RSE) est le composant de Developer for System z qui fournit les services de base comme la connexion du client à l'hôte. L'exécution doit être contrôlée par le programme afin d'effectuer des tâches telles que la commutation sur l'ID utilisateur du client.

Le bit de contrôle du programme est défini pendant l'installation SMP/E au besoin.

Le résultat est vérifiable à l'aide de la commande `ls -lE fekf*` qui donne une sortie semblable à ceci (\$ est l'invite de z/OS UNIX) :

```
$ ls -lE /usr/lpp/wd4z/rse/lib/fekf*
-rwxr-xr-x -ps- 2 user group 94208 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/
fekfdir.dll
-rwxr-xr-x -ps- 2 user group 143360 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfdiwp
-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
```

**Remarque :** `fekfivp*` et `fekfscmd` n'ont pas besoin de l'attribut `+p`.

Exécutez les commandes suivantes si le bit de contrôle du programme a besoin d'être défini manuellement, en supposant que le répertoire d'installation par défaut (/usr/lpp/wd4z/rse/lib/) est utilisé :

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

**Remarque :** Afin de pouvoir utiliser la commande **extattr +p**, vous devez détenir au minimum des droits d'accès READ au profil BPX.FILEATTR.PROGCTL dans la classe FACILITY de votre logiciel de sécurité. Pour plus d'informations, voir *UNIX System Services Planning (GA22-7800)*.

## Ports TCP/IP réservés

A l'aide de la commande **NETSTAT** (TSO ou z/OS UNIX), vous pouvez connaître les ports actuellement en service. Le résultat de cette commande ressemblera aux exemples ci-dessous. Les ports utilisés sont le dernier chiffre (derrière les deux points '.') dans la colonne/ligne "Local Socket". Ces ports étant déjà utilisés, vous ne pouvez pas vous en servir pour la configuration de Developer for System z.

### 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		

Les ports TCP/IP réservés peuvent présenter une autre limitation. Il existe deux espaces communs pour réserver des ports TCP/IP :

- **PROFILE.TCPIP**

Il s'agit du fichier auquel se rapporte l'instruction de définition de données PROFILE de la tâche lancée par TCP/IP, souvent appelée SYS1.TCPPARMS(TCPPROF)

- PORT : réserve un port pour des noms de travaux spécifiés.
- PORTRANGE : réserve une gamme de ports pour des noms de travaux spécifiés.

Pour plus d'informations sur ces instructions, voir *Communications Server IP Configuration Guide (SC31-8775)*.

- **SYS1.PARMLIB(BPXPRMxx)**

- INADDRANYPORT : indique le numéro de port de démarrage pour la série de numéros de port que le système réserve pour être utilisés avec les liaisons PORT 0, INADDR\_ANY. Cette valeur est requise uniquement pour CINET.
- INADDRANYCOUNT : indique le nombre de ports que le système réserve, en commençant par le numéro de port spécifié dans le paramètre INADDRANYPORT. Cette valeur est requise uniquement pour CINET.

Pour plus d'informations sur ces instructions, consultez les ouvrages *UNIX System Services Planning (GA22-7800)* et *MVS Initialization and Tuning Reference (SA22-7592)*.

Ces ports réservés sont répertoriés à l'aide de la commande **NETSTAT PORTL** (TSO ou z/OS UNIX), qui crée un résultat semblable à l'exemple ci-dessous :

```

MVS TCP/IP NETSTAT CS V1R7          TCPIP Name: TCPIP          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

```

Pour plus d'informations sur la commande **NETSTAT**, consultez le document *Communications Server IP System Administrator's Commands (SC31-8781)*.

**Remarque :** La commande **NETSTAT** présente uniquement les informations définies dans **PROFILE.TCPIP**, qui doivent coïncider avec les définitions **BPXPRMxx**. En cas de doute ou d'incident, vérifiez le membre parmlib **BPXPRMxx** pour vérifier les ports réservés dans ce cas.

**Remarque :** Voir «Définitions de ports **PROFILE.TCPIP**», à la page 112 si vous rencontrez des difficultés avec la liaison d'INETD avec les ports réservés.

---

## Taille d'espace adresse

Le serveur RSE, qui est un processus Java UNIX, nécessite une grande taille de région pour exécuter ses fonctions. Il est ainsi important de définir de grandes limites de mémoire pour les espaces d'adresse OMVS.

## Exigences d'INETD

Un serveur RSE est démarré par INETD lorsqu'un client se connecte au port du démon RSE ou quand le client émet le script de démarrage avec REXEC/SSH. Ceci est exécuté avec les limites mémoire d'INETD, qui doivent donc être suffisamment grandes.

- Si le processus serveur UNIX INETD est lancé par JCL à l'aide de BPXBATCH, la taille de la région doit être 0.
- Si INETD est démarré à partir d'une session TSO/OMVS, la taille de la région TSO doit être 2096128 ou plus.

## Limitations définies dans SYS1.PARMLIB(BPXPRMxx)

Configurez MAXASSIZE dans SYS1.PARMLIB(BPXPRMxx), qui définit la taille de région de l'espace adresse (processus) OMVS par défaut, sur 2147483647 ou plus.

Cette valeur est vérifiable et définissable de manière dynamique (avant la procédure de chargement initial suivante) au moyen des commandes de la console suivantes, décrites dans le manuel *MVS System Commands (SA22-7627)* :

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

## Limitations stockées dans le profil de sécurité

Vérifiez ASSIZEMAX dans le segment OMVS de l'ID utilisateur, et définissez-le à 2147483647 ou, de préférence, à NONE pour utiliser la valeur SYS1.PARMLIB(BPXPRMxx).

Avec RACF, cette valeur peut être vérifiée et définie à l'aide des commandes TSO suivantes, décrites dans *Security Server RACF Command Language Reference (SA22-7687)* :

1. LISTUSER userid NORACF OMVS
2. ALTUSER userid OMVS(NOASSIZEMAX)

## Limitations forcées par les sorties du système

Assurez-vous que vous n'autorisez pas aux sorties du système IEFUSI ou IEALIMIT de contrôler les tailles de région d'adresse OMVS. Il est possible de réaliser cela en codant SUBSYS(OMVS,NOEXITS) dans SYS1.PARMLIB(SMFPRMxx).

Les valeurs de SYS1.PARMLIB(SMFPRMxx) peuvent être vérifiées et définies à l'aide des commandes de console suivantes, décrites dans *MVS System Commands (SA22-7627)* :

1. DISPLAY SMF,0
2. SET SMF=xx

---

## Traçage de suivi des erreurs

La procédure suivante permet de rassembler les informations nécessaires pour diagnostiquer les incidents de suivi des erreurs avec les procédures d'assemblage à distance. La fonction de trace réduit les performances et ne doit être effectuée que sur indication du centre de support IBM. Toutes les références à hlq que vous trouverez dans cette section se rapportent au qualificatif de haut niveau utilisé au cours de l'installation de Developer for System z. L'installation par défaut est FEK, mais peut ne pas s'appliquer à votre site.

1. Faites une copie de sauvegarde votre procédure de compilation active ELAXFCOC. Cette procédure est normalement présente dans le fichier hlq.SFEKSAMP à la livraison, mais peut avoir été copiée à un emplacement différent, comme SYS1.PROCLIB, comme décrit dans «Personnalisation ELAXF\*, procédure de construction à distance», à la page 24.
2. Modifiez la procédure active ELAXFCOC afin qu'elle comprenne la chaîne 'MAXTRACE' sur l'option de compilation EXIT(ADEXIT(ELAXMGUX)).

```
//COBOL EXEC PGM=IGYCRCTL,REGION=2048K,  
//*      PARM=('EXIT(ADEXIT(ELAXMGUX))',  
//      PARM=('EXIT(ADEXIT('MAXTRACE',ELAXMGUX))',  
//      'ADATA',  
//      'LIB',
```

```
//          'TEST(NONE,SYM,SEP)',
//          'LIST',
//          'FLAG(I,I)'&CICS &DB2 &COMP)
```

**Remarque :** Vous devez doubler les apostrophes autour de MAXTRACE.  
L'option est maintenant : EXIT(ADEXIT('MAXTRACE',ELAXMGUX))

3. Effectuez une vérification de la syntaxe à distance sur un programme en langage COBOL. Developer for System z comprend un exemple de programme en langage COBOL dans hlq.SFEKSAMP(ADNSMSGH).
4. La partie SYSOUT de la sortie JES démarre en faisant la liste des noms des fichiers pour SIDEFILE1, SIDEFILE2, SIDEFILE3 et 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'
```

**Remarque :** Selon vos paramètres, SIDEFILE1 et SIDEFILE2 peuvent pointer vers une instruction de définition de données (SUCCESSFUL OPEN SIDEFILE1 - NAME = DD:WSEDSF1). Reportez-vous au composant JESJCL de la sortie (qui est situé avant le composant SYSOUT) pour connaître le nom réel du fichier.

```
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. Copiez ces quatre fichiers sur votre PC, par exemple en créant un projet local en langage COBOL dans Developer for System z et en ajoutant les fichiers SIDEFILE1->4.
6. Copiez le journal de travail JES sur votre PC, par exemple en ouvrant la sortie de travaux dans Developer for System z et en l'enregistrant dans le projet local. Pour cela, sélectionnez **Fichier > Enregistrer sous...**
7. Restaurez la procédure ELAXFCOC à son état original, soit en annulant les modifications (retirez la chaîne "MAXTRACE" des options de compilation) soit en restaurant la sauvegarde.
8. Envoyez les fichiers collectés (SIDEFILE1->4 et le journal de travail) au point service IBM.

---

## Transaction APPC et service Commandes TSO

Si vous ne pouvez pas utiliser la version APPC du service Commandes TSO, des incidents peuvent survenir dans les deux domaines suivants : le démarrage de la transaction du serveur APPC et la connexion au RSE.

- Si aucun message sur la configuration d'APPC n'apparaît, recherchez dans le journal système les messages RACF (ID message ICHxxxxx) ou les autres messages concernant la commande lancée ou l'ID utilisateur qui a émis la commande. Les causes fréquentes des incidents sont notamment :
  - Vous ne disposez pas des droits d'accès en lecture pour accéder au fichier hlq.SFEKPROC.
  - Le protocole TCP/IP n'est pas actif, possède un nom DNS incorrect ou le système n'est pas accessible (la commande ping ne peut s'y exécuter) à cause d'incidents réseau, d'une adresse IP incorrecte ou d'autres causes.



- Si vous visualisez les messages de configuration APPC mais qu'aucun message ne confirme que la configuration a réussi, la transaction du serveur APPC n'est peut-être pas parvenue à démarrer. Consultez le journal des erreurs de transaction (userid.FEKFRSRV.&TPDATE.&TPTIME.LOG). Parmi les causes fréquentes d'incident, vous trouverez :
  - La pile TCP/IP n'utilise pas le nom par défaut TCPIP et la carte de pilote de périphérique SYSTCPD n'a pas été définie ou indique un fichier incorrect.
  - Le serveur ne parvient pas à allouer SYSPROC ou SYSTSPRT.
  - Le langage JCL indique un SYSPROC incorrect (SYSPROC doit comprendre hlq.SFEKPROC).
  - Le serveur ne parvient pas à ouvrir ou à accéder au fichier de messages (journal) indiqué par MESSAGE\_DATA\_SET.
  - Le nombre de demandeurs de planificateur APPC est insuffisant.
  - Les emplacements de l'adresse APPC ou ASCH ne sont pas actifs.
  - La classe utilisée (nommée "A" par défaut) n'est pas définie pour l'ASCH du planificateur APPC.
  - Il n'y a pas de segment OMVS par défaut pour le système et l'utilisateur ne possède pas de segment OMVS personnel, ou il y a une erreur de définition dans les deux.
  - Le groupe par défaut du segment OMVS par défaut ou le groupe par défaut de l'utilisateur ne présente pas de numéro d'ID groupe.

Le REXX fourni dans «(Facultatif) Définition d'une transaction APPC pour le service Commandes TSO», à la page 26 permet de résoudre les incidents APPC en vous donnant la possibilité de gérer les APPC de manière interactive via les panneaux ISPF. Toutefois, vous devez savoir que vous pouvez désactiver la transaction avec cet outil ; la transaction existe toujours mais n'accepte aucune connexion.

Les fiches techniques disponibles à la section Support du site Web (<http://www-306.ibm.com/software/awdtools/devzseries/support/>) sont répertoriées ci-dessous. Pour plus d'informations, consultez le support du site Web :

- Echec de la vérification APPC avec un code retour 2016 - EHOSTNOTFOUND
- Echec de la vérification APPC avec un code retour 1004 - EIBMIUCVERR
- Echec de la vérification APPC avec un code retour 9 - Nom du programme transactionnel non reconnu
- Echec de la vérification APPC avec un code retour 10 - Programme transactionnel non disponible, pas de relance
- Echec de la vérification APPC avec un code retour 19 - Erreur de paramètre
- Echec de la vérification APPC avec un code retour 20 - Erreur propre au produit
- Echec de la vérification APPC avec un code retour 26 - Panne de la ressource
- CEE3501S : Module IOSTREAM introuvable
- Echec du démarrage du serveur : EDC5129I Ce fichier ou ce répertoire est introuvable
- exec/tcp : Liaison : EDC5111I Droit refusé, rsn=744C7246
- Aucune réponse du serveur, avec l'un des messages suivants
  - IEA995I SYMPTOM DUMP OUTPUT 473 USER COMPLETION CODE=4093 REASON CODE=0000001C (in SDSF LOG)

- CEE3512S Le chargement du système hiérarchique de fichiers du module libicudata32.0.dll a échoué. Le code retour du système est 0000000157 ; le code anomalie est 0BDF019B. (dans CEEDUMP)
- Echec d'obtention de mémoire (dans client .log)
- La commande C\_CONNECT n'est pas disponible
- "Echec de l'initialisation du serveur FFS" message d'erreur lors de la tentative de connexion à l'hôte
- "EDC5139I Opération non autorisée" lors de la tentative de connexion à l'hôte
- "RSEG1056U Echec de l'initialisation du serveur FFS" à l'ouverture d'un fichier MVS

**Remarque :** Cette liste n'est pas exhaustive, contactez le support du site Web pour obtenir des fiches techniques supplémentaires.

---

## Informations diverses

### Limites du système

SYS1.PARMLIB(BPXPRMxx) définit plusieurs limitations liées à z/OS UNIX, qui peuvent être atteintes lorsque plusieurs clients de Developer for System z sont actifs. La plupart des valeurs BPXPRMxx peuvent être modifiées de façon dynamique avec les commandes de console **SETOMVS** et **SET OMVS**.

#### Connexion refusée

Chaque connexion RSE lance plusieurs processus qui sont actifs de façon permanente. De nouvelles connexions peuvent être refusées en raison de la limite définie dans SYS1.PARMLIB(BPXPRMxx) concernant la quantité de processus, particulièrement lorsque les utilisateurs partagent le même ID utilisateur (quand le segment OMVS par défaut est utilisé par exemple).

- La limite par ID utilisateur est définie par le mot clé MAXPROCUSER, et a une valeur par défaut de 25.
- La limite au niveau du système est définie par le mot clé MAXPROCSYS, et a une valeur par défaut de 200.

La limite d'espaces adresse z/OS et d'utilisateurs z/OS UNIX actifs représente une autre source de connexions refusées.

- La quantité maximale d'ID espace adresse (ASID) est définie dans SYS1.PARMLIB(IEASYSxx) avec le mot clé MAXUSER, et présente une valeur par défaut de 255.
- La quantité maximale d'ID utilisateur (UID)z/OS UNIX est définie dans SYS1.PARMLIB(BPXPRMxx) avec le mot clé MAXUIDS, et présente une valeur par défaut de 200.

### Incidents connus

#### Echec de l'ouverture de fichiers MVS

La lecture et l'écriture d'un fichier MVS nécessite l'emploi d'un socket de domaine de système de fichiers physique. Si le système de fichiers n'est pas correctement défini ou s'il n'a pas suffisamment de sockets, le gestionnaire de verrous (FFS) peut échouer dans les requêtes de lecture/écriture. Le fichiers ffs\*.log présenteront des messages comme :

- Erreur 127 à l'obtention de la paire de sockets - port défini à 0.



- Impossible de créer le socket dans le domaine UNIX. L'erreur est : "La famille d'adresses n'est pas prise en charge"

Assurez-vous que le membre SYS1.PARMLIB(BPXPRMxx) contient les instructions suivantes :

```
FILESYSTYPE TYPE(UDS) ENTRYPPOINT(BPXTUINT)
NETWORK DOMAINNAME(AF_UNIX)
        DOMAINNUMBER(1)
        MAXSOCKETS(200)
        TYPE(UDS)
```

### Echec des sessions de liaison DVIPA

Lors de l'utilisation de VIPA (adresses IP virtuelles) dynamiques au niveau de plusieurs piles TCPIP, une coordination au niveau de sysplex de l'affectation des port éphémères est nécessaire de façon à ce que le 4-uplet (combinaison de ports et d'adresses IP source et destination) reste unique pour chaque connexion. Cela peut être fait par l'ajout du paramètre facultatif SYSPLEXPORTS à la première instruction VIPADISTRIBUTE, voir *Communications IP Configuration Guide (SC31-8775)*.

Lors de l'emploi de cette technique, assurez-vous que la structure de l'unité de couplage EZBEPOR (qui contient les informations d'affectation de port sysplex) a été définie. Pour plus d'informations sur le sujet, voir *Communications Server SNA Network Implementation Guide (SC31-8777)*.

### Emulateur de connexion à l'hôte

- L'émulateur de connexion à l'hôte utilise telnet TN3270 et non le serveur RSE pour se connecter à l'hôte.
- Lorsque vous utilisez telnet sécurisé (SSL) et que vous travaillez avec des certificats qui ne sont pas signés par une autorité de certification bien connue, chaque client doit ajouter le certificat de CA à la liste d'autorités de certification dignes de confiance de son émulateur de connexion à l'hôte.
- L'option NOSNAEXT de TELNETPARMS de TCP/IP peut être nécessaire pour désactiver les extensions fonctionnelles de l'architecture SNA. Dans le cas où NOSNAEXT est indiqué, le serveur telnet TN3270 ne négocie pas pour la résolution des conflits et les fonctions de détection d'architecture SNA.

---

## Contacteur le support IBM

Si les incidents ne sont pas résolus après la lecture du présent manuel et la consultation du site Web de support (<http://www-306.ibm.com/software/awdtools/devzseries/support/>), et que vous avez besoin de l'aide du support IBM, veuillez réunir les informations suivantes et ouvrir un enregistrement d'incident auprès d'IBM :

- Votre niveau de logiciel en cours, qui est indiqué dans :
  - hlq.SFEKJCL(FEK#LVL#) pour Developer for System z. La valeur par défaut pour hlq est FEK.
  - hlq.SCRAJCL(CRA#LVL#) pour CARMA. La valeur par défaut pour hlq est CRA.

Les informations suivantes sont utiles pour résoudre les incidents de connexion :

- La sortie de la/des commande(s) de console suivante(s) :
  - D OMVS,0
- La sortie de la/des commande(s) TSO suivante(s) :
  - HOMETEST

- LISTUSER userid NORACF OMVS
- LISTGROUP group NORACF OMVS
- Le fichier rsed.envvars, situé par défaut dans /usr/lpp/wd4z/rse/lib/, mais dont la personnalisation dans /etc/wd4z/ est conseillée
- Le fichier /etc/inetd.conf
- Tous les fichiers \*.log situés dans home/.eclipse/RSE/USERID/, où home désigne le chemin d'accès à l'accueil défini dans le segment OMVS de l'utilisateur (ou le segment OMVS par défaut si l'utilisateur n'en possède pas) et USERID correspond à l'ID utilisateur de connexion (majuscule).
- Les messages SYSLOG associés
- La sortie complète du ou des test(s) de programme IVP associé(s), comme décrit dans «Vérification de la configuration du serveur RSE», à la page 51.
  - netstat : «Disponibilité des ports», à la page 53
  - fekfivps : «Script de shell REXEC/SSH», à la page 54
  - fekfivpr : «Connexion REXEC», à la page 53
  - fekfivpd : «Connexion du démon RSE», à la page 55.
  - fekfivpj : «Connexion du moniteur de travaux JES», à la page 55.
  - fekfivpc : «Connexion du service Commandes TSO (avec SCLMDT)», à la page 55
  - fekfivpa : «Connexion du service Commandes TSO (avec APPC)», à la page 57

Si vous utilisez SCLM Developer Toolkit pour le service Commandes TSO (cas par défaut)

- La sortie complète du test de programme IVP de SCLMDT (fekfivpc), comme décrit dans «Connexion du service Commandes TSO (avec SCLMDT)», à la page 55
- Le fichier /etc/SCLMDT/CONFIG/ISPF.conf

Si vous utilisez APPC pour le service Commandes TSO

- La sortie complète du test de programme IVP d'APPC (fekfivpa), comme décrit dans «Connexion du service Commandes TSO (avec APPC)», à la page 57
- Le JCL qui définit la transaction du serveur APPC, qui correspond à hlq.SFEKSAMP(FEKAPPCC) si vous avez utilisé le modèle de travail
- Membre SYS1.PARMLIB(APPCPMxx)
- Membre SYS1.PARMLIB(ASCHPMxx)
- Fichier userid.FEKFSERV.&TPDATE.&TPTIME.LOG
- Les sorties des commandes de la console suivantes :
  - D APPC,TP,ALL
  - D APPC,LU,ALL
  - D ASCH,ALL

Mettez à disposition toutes les informations qui semblent importantes pour les incidents de fonctionnement comme

- Le journal de travail
- Les messages SYSLOG associés

- Les fichiers journaux situés dans `home/.eclipse/RSE/USERID/`, où `home` désigne le chemin d'accès à l'accueil défini dans le segment OMVS de l'utilisateur (ou le segment OMVS par défaut si l'utilisateur n'en possède pas) et `USERID` correspond à l'ID utilisateur de connexion (majuscule).



---

## Annexe C. Configuration du TCP/IP

Cette annexe vous aide à résoudre certains incidents fréquents qui peuvent survenir lors de la configuration du TCP/IP, ou au cours de la vérification et/ou de la modification d'une configuration existante.

Pour plus d'informations sur la configuration TCP/IP, consultez les ouvrages *Communications Server: IP Configuration Guide (SC31-8775)* et *Communications Server IP Configuration Reference (SC31-8776)*.

---

### Dépendance au nom d'hôte

Developer for System z requiert que le protocole TCP/IP dispose du nom d'hôte correct lors de son initialisation. Cela implique que les différents fichiers de configuration TCP/IP et du programme de résolution soient configurés correctement.

Vous pouvez tester votre configuration TCP/IP à l'aide de la commande TSO **HOMETEST**. Pour plus d'informations sur la commande **HOMETEST**, consultez le document *Communications Server IP System Administrator's Commands (SC31-8781)*.

Exemple de résultat de la commande **HOMETEST** :

Exécution du programme de test de la configuration TCP/IP IBM MVS  
TCP/IP CS V1R7

Le fichier de paramètre de configuration FTP utilisé est le "SYS1.TCPPARMS(FTPDATA)"

Le nom d'hôte TCP est : CDFMVS08

Utilisation du serveur de noms pour une résolution CDFMVS08  
Les adresses IP suivantes correspondent au nom d'hôte TCP : CDFMVS08  
9.42.112.75

Les adresses IP suivantes correspondent aux adresses IP HOME définies dans  
PROFILE.TCPIP :  
9.42.112.75  
127.0.0.1

Toutes les adresses IP pour CDFMVS08 se trouvent dans la liste HOME !

Commande Hometest réussie – Tous les tests sont réussis !

---

### Présentation du programme de résolution

Le programme de résolution joue le rôle des programmes comme un client qui accède aux serveurs de noms pour une résolution nom/adresse ou adresse/nom. Pour résoudre la requête du programme demandeur, le programme de résolution peut accéder aux serveurs de noms disponibles, utiliser des définitions locales (/etc/resolv.conf, /etc/hosts, /etc/ipnodes, HOSTS.SITEINFO, HOSTS.ADDRINFO, ou ETC.IPNODES, par exemple) ou utiliser une combinaison des deux.

Quand l'espace adresse du programme de résolution démarre, il lit un fichier de configuration du programme de résolution facultatif indiqué par la carte SETUP DD dans la procédure JCL du programme de résolution. Si les informations de configuration ne sont pas fournies, le programme de résolution utilise l'ordre de

recherche MVS ou z/OS UNIX natif applicable, sans aucune information GLOBALTCPDATA, DEFAULTTCPDATA, GLOBALIPNODES, DEFAULTIPNODES ou COMMONSEARCH.

**Remarque :** Si aucune procédure JCL de résolution ne se trouve dans la bibliothèque de procédure système, l'espace d'adresse commencera à utiliser les valeurs par défaut du système (non SETUP DD).

---

## Présentation des ordres de recherche d'informations de configuration

Il est important de connaître les fonctions de l'ordre de recherche des fichiers de configuration utilisés par TCP/IP, et de savoir quand vous pouvez remplacer l'ordre de recherche par défaut par des variables d'environnement, JCL ou autre variable fournie. Vous pouvez ainsi adapter votre fichier de données locales et les normes de dénomination d'un fichier d'un système hiérarchique ; il est également utile de savoir s'il s'agit du fichier de données de configuration ou du fichier du système hiérarchique qui est utilisé au moment d'identifier les incidents.

Autre point important, quand un ordre de recherche est appliqué à n'importe quel fichier de configuration, la recherche s'arrête au premier fichier trouvé. Par conséquent, vous risquez de trouver des résultats inattendus si vous enregistrez des informations de configuration dans un fichier qui n'est jamais trouvé, soit parce que d'autres fichiers le précèdent dans l'ordre de recherche ou parce que le fichier n'est pas inclus dans l'ordre de recherche choisi par l'application.

Quand vous recherchez des fichiers de configuration, vous pouvez indiquer clairement au protocole TCP/IP l'emplacement de la plupart des fichiers de configuration en utilisant des instructions de définition de données dans les procédures JCL ou en définissant des variables d'environnement. Sinon, vous pouvez laisser le protocole TCP/IP déterminer de manière dynamique l'emplacement des fichiers de configuration, en fonction des ordres de recherche documentés dans le guide *Communications Server IP Configuration Guide* (SC31-8775).

Le composant de configuration de la pile TCP/IP utilise TCPIP.DATA au cours de l'initialisation de la pile TCP/IP afin d'en déterminer le paramètre HOSTNAME. L'ordre de recherche de l'environnement UNIX z/OS permet d'obtenir sa valeur.

**Remarque :** L'utilitaire du programme de résolution de trace permet de déterminer les valeurs TCPIP.DATA qui sont utilisées par le programme de résolution et leur emplacement au moment de la lecture. Pour plus d'informations sur le démarrage dynamique de la trace, consultez le guide *Communications Server IP Diagnosis Guide* (GC31-8782). Quand la trace est active, lancez une commande TSO **NETSTAT HOME** et une commande **netstat -h** de l'interpréteur de commandes UNIX sous z/OS pour afficher les valeurs. Une commande PING de nom d'hôte lancée depuis une commande TSO et de l'interpréteur de commandes UNIX sous z/OS affiche également l'activité de tous les serveurs DNS qui devraient être configurés.

---

## Ordres de recherche dans l'environnement UNIX sous z/OS

Le tableau ou fichier particulier recherché correspond à un fichier MVS ou à un fichier de système hiérarchique, en fonction des paramètres de configuration du programme de résolution et de la présence de ces fichiers sur le système.

## Fichiers de configuration du programme de résolution de base :

Le fichier de configuration du programme de résolution de base contient des instructions TCPIP.DATA. Outre les directives du programme de résolution, il est référencé pour déterminer, entre autres, le préfixe du fichier (valeur de l'instruction DATASETPREFIX) à utiliser lors de la tentative d'accès à certains des fichiers de configuration spécifiés dans cette section.

L'ordre de recherche permettant d'accéder au fichier de configuration du programme de résolution de base est le suivant :

### 1. GLOBALTCPIPDATA

Si ce fichier est défini, la valeur de l'instruction de configuration GLOBALTCPIPDATA du programme de résolution est utilisée (voir aussi «Présentation du programme de résolution», à la page 101). La recherche se poursuit pour trouver un autre fichier de configuration. La recherche s'arrête avec le fichier trouvé suivant.

### 2. La valeur de la variable d'environnement RESOLVER\_CONFIG.

La valeur utilisée est celle de la variable d'environnement. La recherche échoue si le fichier n'existe pas ou s'il se trouve dans un autre emplacement.

### 3. /etc/resolv.conf

### 4. Carte //SYSTCPD DD

Le fichier utilisé est celui qui est attribué au nom DD SYSTCPD. Dans l'environnement UNIX sous z/OS, un processus enfant n'a pas accès au SYSTCPD DD. En effet, l'attribution SYSTCPD n'est pas héritée du processus père sur le processus parallèle de traitement() ou les appels de fonction de commande exec.

### 5. userid.TCPIP.DATA

userid désigne l'ID utilisateur qui est associé à l'environnement de sécurité en cours (espace adresse ou tâche/unité d'exécution).

### 6. jobname.TCPIP.DATA

jobname correspond au nom indiqué dans l'instruction JCL JOB pour les travaux par lots ou le nom de la procédure pour une procédure démarrée.

### 7. SYS1.TCPPARMS(TCPDATA)

### 8. DEFAULTTCPIPDATA

Si ce fichier est défini, la valeur de l'instruction de configuration DEFAULTTCPIPDATA du programme de résolution est utilisée (voir aussi «Présentation du programme de résolution», à la page 101).

### 9. TCPIP.TCPIP.DATA

## Tables de conversion :

Les tables de conversion (EBCDIC en ASCII et ASCII en EBCDIC) permettent de déterminer les fichiers de conversion à utiliser. L'ordre de recherche permettant d'accéder à ce fichier de configuration est le suivant. L'ordre de recherche s'arrête au premier fichier trouvé :

### 1. La valeur de la variable d'environnement X\_XLATE. La valeur de la variable d'environnement correspond au nom de la table de conversion créée par la commande TSO CONVXLAT.

### 2. userid.STANDARD.TCPXLBIN

userid désigne l'ID utilisateur qui est associé à l'environnement de sécurité en cours (espace adresse ou tâche/unité d'exécution).

3. **jobname.STANDARD.TCPXLBIN**  
jobname correspond au nom indiqué dans l'instruction JCL JOB pour les travaux par lots ou le nom de la procédure pour une procédure démarrée.
4. **hlq.STANDARD.TCPXLBIN**  
hlq représente la valeur de l'instruction DATASETPREFIX spécifiée dans le fichier de configuration du programme de résolution de base (le cas échéant) ; sinon, hlq correspond au protocole TCPIP par défaut.
5. Si aucune table n'est trouvée, le programme de résolution utilise une table codée en dur par défaut qui est identique à la table de la liste du membre de fichier SEZATCPX(STANDARD).

## Tables de système hôte local :

Par défaut, le programme de résolution tente tout d'abord d'utiliser n'importe quel serveur de noms de domaine configuré pour les demandes de résolution. Si la demande de résolution n'est pas satisfaite, les tables de système hôte local sont utilisées. Le comportement du programme de résolution est contrôlé par les instructions TCPIP.DATA.

Les instructions du programme de résolution TCPIP.DATA définissent si et comment des serveurs de noms de domaine doivent être utilisés. L'instruction LOOKUP TCPIP.DATA permet également de contrôler l'utilisation des serveurs de noms de domaine et des tables de système hôte local. Pour plus d'informations sur les instructions TCPIP.DATA, consultez l'ouvrage *Communications Server IP Configuration Reference* (SC31-8776).

Le programme de résolution se sert de l'ordre de recherche unique Ipv4 pour trouver des informations de nom de site sans restrictions pour les appels API getnetbyname. L'ordre de recherche unique Ipv4 pour des informations de nom de site est le suivant. La recherche s'arrête au premier fichier trouvé :

1. La valeur de la variable d'environnement **X\_SITE**.  
La valeur de la variable d'environnement correspond au nom du fichier d'informations HOSTS.SITEINFO créé par la commande TSO **MAKESITE**.
2. La valeur de la variable d'environnement **X\_ADDR**.  
La valeur de la variable d'environnement correspond au nom du fichier d'informations HOSTS.ADDRINFO créé par la commande TSO **MAKESITE**.
3. **/etc/hosts**
4. **userid.HOSTS.SITEINFO**  
userid désigne l'ID utilisateur qui est associé à l'environnement de sécurité en cours (espace adresse ou tâche/unité d'exécution).
5. **jobname.HOSTS.SITEINFO**  
jobname correspond au nom indiqué dans l'instruction JCL JOB pour les travaux par lots ou le nom de la procédure pour une procédure démarrée.
6. **hlq.HOSTS.SITEINFO**  
hlq représente la valeur de l'instruction DATASETPREFIX spécifiée dans le fichier de configuration du programme de résolution de base (le cas échéant) ; sinon, hlq correspond au protocole TCPIP par défaut.



---

## Application à Developer for System z

Comme mentionné précédemment, Developer for System z dépend de la validité du nom d'hôte du TCP/IP quand il est initialisé. Cela implique que les différents fichiers de configuration TCP/IP et du programme de résolution soient configurés correctement.

L'exemple ci-dessous présente certaines tâches de la configuration de TCP/IP et du programme de résolution. Il ne couvre pas l'ensemble de la configuration, il souligne uniquement certains aspects clés qui peuvent s'appliquer à votre site.

1. Dans le JCL ci-dessous, nous voyons que le protocole TCP/IP va se servir de SYS1.TCPPARMS(TCPDATA) pour déterminer le nom d'hôte de la pile.

```
//TCPIP    PROC  PARM='CTRACE(CTIEZB00)',PROF=TCPPROF,DATA=TCPDATA
//*
//* TCP/IP NETWORK
//*
//TCPIP    EXEC  PGM=EZBTCP,REGION=0M,TIME=1440,PARM=&PARMS
//PROFILE  DD   DISP=SHR,DSN=SYS1.TCPPARMS(&PROF)
//SYSTCPD  DD   DISP=SHR,DSN=SYS1.TCPPARMS(&DATA)
//SYSPRINT DD   SYSOUT=*,DCB=(RECFM=VB,LRECL=132,BLKSIZE=136)
//ALGPRINT DD   SYSOUT=*,DCB=(RECFM=VB,LRECL=132,BLKSIZE=136)
//CFGPRINT DD   SYSOUT=*,DCB=(RECFM=VB,LRECL=132,BLKSIZE=136)
//SYSOUT   DD   SYSOUT=*,DCB=(RECFM=VB,LRECL=132,BLKSIZE=136)
//CEEDUMP  DD   SYSOUT=*,DCB=(RECFM=VB,LRECL=132,BLKSIZE=136)
//SYSERROR DD   SYSOUT=*
```

2. SYS1.TCPPARMS(TCPDATA) indique que nous voulons le nom du système comme nom d'hôte et que nous n'utilisons pas de serveur de noms de domaine (DNS) ; tous les noms seront résolus via la consultation du tableau du site.

```
; HOSTNAME indique le nom d'hôte TCP de ce système. S'il n'est pas
; spécifié, le paramètre HOSTNAME par défaut sera le nom de noeud spécifié
; dans le membre IEFSSNxx PARMLIB.
;
; HOSTNAME
;
; DOMAINORIGIN indique l'origine du domaine qui sera ajoutée
; aux noms d'hôte transmis au programme de résolution. Si un nom d'hôte contient
; des points, le paramètre DOMAINORIGIN ne sera pas ajouté au
; nom d'hôte.
;
DOMAINORIGIN  RALEIGH.IBM.COM
;
; NSINTERADDR indique l'adresse IP du serveur de noms.
; LOOPBACK (14.0.0.0) indique votre serveur de noms local. Si vous n'utilisez
; pas de serveur de noms, ne codez pas d'instruction NSINTERADDR.
; (Mettez en commentaire la ligne NSINTERADDR ci-dessous). Cette action va
; résoudre tous les noms via la consultation du tableau de site.
;
; NSINTERADDR  14.0.0.0
;
; TRACE RESOLVER va créer une trace complète des requêtes à destination
; et des réponses provenant du serveur de noms ou des tableaux de site
; à écrire dans la console de l'utilisateur. Cette commande s'applique à
; des fins de débogage uniquement.
;
; TRACE RESOLVER
```

3. Dans le langage JCL du programme de résolution, l'instruction SETUP DD n'est pas utilisée. Comme mentionné dans «Présentation du programme de résolution», à la page 101, cela signifie que GLOBALTCPIPDATA et les autres variables ne seront pas utilisées.

```
//RESOLVER PROC PARMS='CTRACE(CTIRES00)'
//*
//* IP NAME RESOLVER – START WITH SUB=MSTR
//*
//RESOLVER EXEC PGM=EZBREINI,REGION=0M,TIME=1440,PARM=&PARMS
//*SETUP DD DISP=SHR,DSN=USER.PROCLIB(RESSETUP),FREE=CLOSE
```

4. Supposons que la variable d'environnement RESOLVER\_CONFIG n'est pas définie, le tableau 13 nous montre que le programme de résolution va tenter d'utiliser /etc/resolv.conf comme fichier de configuration de base.

```
TCPIPJOBNAME TCPIP
DomainOrigin RALEIGH.IBM.COM
HostName CDFMVS08
```

Comme mentionné dans «Ordres de recherche dans l'environnement UNIX sous z/OS», à la page 102, le fichier de configuration de base contient des instructions TCPIP.DATA. Si le nom du système est CDFMVS08 (TCPDATA indique que le nom du système est utilisé comme nom d'hôte), nous pouvons voir que /etc/resolv.conf est en synchronisation avec SYS1.TCPPARMS(TCPDATA). Aucune définition de système de nom de domaine n'existe, par conséquent la consultation du tableau de site sera utilisée.

5. Le tableau 13 nous indique également que nous n'avons rien à faire pour utiliser la table de conversion ASCII-EBCDIC par défaut.
6. En supposant que la commande TSO **MAKESITE** n'est pas utilisée (possibilité de créer les variables X\_SITE et X\_ADDR), /etc/hosts sera le tableau de site désigné pour la consultation du nom.

```
# Resolver /etc/hosts file cdfmvs08
9.42.112.75 cdfmvs08 # CDFMVS08 Host
9.42.112.75 cdfmvs08.raleigh.ibm.com # CDFMVS08 Host
127.0.0.1 localhost
```

Le contenu minimal de ce fichier comprend des informations sur le système actuel. Dans l'exemple ci-dessus, nous avons défini cdfmvs08 et cdfmvs08.raleigh.ibm.com comme un nom valide pour l'adresse IP de notre système z/OS.

Si nous utilisons un serveur de noms de domaine (DNS), le DNS contiendrait des informations /etc/hosts, et /etc/resolv.conf et SYS1.TCPPARMS(TCPDATA) auraient des instructions identifiant le DNS sur notre système.

Pour éviter toute confusion, nous vous recommandons de conserver les fichiers de configuration TCP/IP et du programme de résolution en synchronisation les uns avec les autres.

Tableau 13. Définitions locales disponibles pour le programme de résolution

Description de type de fichier	API concernée(s)	Fichiers candidats
Fichiers de configuration du programme de résolution de base	Toutes les API	<ol style="list-style-type: none"> <li>1. GLOBALTCPIPDATA</li> <li>2. Variable d'environnement RESOLVER_CONFIG</li> <li>3. /etc/resolv.conf</li> <li>4. SYSTCPD DD-name</li> <li>5. userid.TCPIP.DATA</li> <li>6. jobname.TCPIP.DATA</li> <li>7. SYS1.TCPPARMS(TCPDATA)</li> <li>8. DEFAULTTCPIPDATA</li> <li>9. TCPIP.TCPIP.DATA</li> </ol>

Tableau 13. Définitions locales disponibles pour le programme de résolution (suite)

Description de type de fichier	API concernée(s)	Fichiers candidats
Tables de conversion	Toutes les API	<ol style="list-style-type: none"> <li>1. Variable d'environnement X_XLATE</li> <li>2. userid.STANDARD.TCPXLBIN</li> <li>3. jobname.STANDARD.TCPXLBIN</li> <li>4. hlq.STANDARD.TCPXLBIN</li> <li>5. Table de conversion fournie par le programme de résolution, membre STANDARD dans SEZATCPX</li> </ol>
Tables de système hôte local	endhostent endnetent getaddrinfo gethostbyaddr gethostbyname gethostent GetHostNumber GetHostResol GetHostString getnameinfo getnetbyaddr getnetbyname getnetent IsLocalHost Resolve sethostent setnetent	IPv4 <ol style="list-style-type: none"> <li>1. Variable d'environnement X_SITE</li> <li>2. Variable d'environnement X_ADDR</li> <li>3. /etc/hosts</li> <li>4. userid.HOSTS.xxxxINFO</li> <li>5. jobname.HOSTS.xxxxINFO</li> <li>6. hlq.HOSTS.xxxxINFO</li> </ol> IPv6 <ol style="list-style-type: none"> <li>1. GLOBALIPNODES</li> <li>2. Variable d'environnement RESOLVER_IPNODES</li> <li>3. userid.ETC.IPNODES</li> <li>4. jobname.ETC.IPNODES</li> <li>5. hlq.ETC.IPNODES</li> <li>6. DEFAULTIPNODES</li> <li>7. /etc/ipnodes</li> </ol>

**Remarque :** Le tableau ci-dessus est une copie partielle d'un tableau situé dans le guide *Communications Server IP Configuration Guide (SC31-8775)*. Consultez ce manuel pour voir le tableau complet.



---

## Annexe D. Configuration d'INETD

Cette annexe vous aide à résoudre certains incidents fréquents qui peuvent survenir lors de la configuration d'INETD, ou au cours de la vérification et/ou de la modification d'une configuration existante.

Le démon INETD fournit une gestion des services d'un réseau IP. Il réduit la charge du système en appelant d'autres démons uniquement lorsqu'ils sont nécessaires et en fournissant en interne des services internet simples (comme écho). INETD consulte le fichier de configuration `inetd.conf` afin de déterminer quels services supplémentaires proposer. `ETC.SERVICES` sert à lier les services aux ports.

---

### `inetd.conf`

Les services qui dépendent de INETD sont définis dans `inetd.conf`, qui est lu par INETD au moment du démarrage. Le nom et emplacement par défaut de `inetd.conf` est `/etc/inetd.conf`. Un exemple de fichier `inetd.conf` se trouve dans `/samples/inetd.conf`.

La syntaxe suivante s'applique aux entrées `inetd.conf`:

- Les commentaires commencent avec un dièse (#) ou un point-virgule (;) et se poursuivent jusqu'à la fin de la ligne
- Les entrées sont sensibles à la casse
- Les entrées sont sensibles au champ mais pas à la colonne
- Les champs sont séparés par un espace ou un caractère de tabulation
- Les entrées peuvent s'étendre sur plusieurs lignes, selon cette syntaxe supplémentaire
  - La séparation doit se situer entre deux mots distincts (séparés par un espace ou un caractère de tabulation)
  - La ligne de continuation doit démarrer par un espace ou un caractère de tabulation
  - Aucun commentaire ne peut être incorporé dans la suite

Chaque entrée se compose de 7 champs à position fixe, selon le format :

```
service_name socket_type protocol wait_flag userid server_program server  
_program_arguments
```

#### **[ip\_address:]service\_name**

`ip_address` est une adresse IP locale, suivie par un double point (:). Si elle est indiquée, l'adresse est utilisée à la place d'`INADDR_ANY` ou de la valeur par défaut en cours. Pour demander spécifiquement `INADDR_ANY`, utilisez `"*."`. Si `ip_address` (ou un double point) est indiquée sans autre entrée sur la ligne, elle devient la nouvelle valeur par défaut pour les lignes suivantes jusqu'à ce qu'une nouvelle valeur par défaut soit indiquée. `service_name` est un nom de service identifié, ou défini par l'utilisateur. Le nom indiqué doit correspondre l'un des noms de serveur définis dans `ETC.SERVICES`.

#### **socket\_type**

`stream` ou `dgram`, pour indiquer que respectivement un flux ou un datagramme est utilisé pour le service.

**protocol[,sndbuf=n[,rcvbuf=n]]**

protocol peut être tcp[4|6] ou udp[4|6], et est utilisé pour qualifier plus avant le nom de service. Le nom de service et le protocole doivent tous les deux correspondre à une entrée dans ETC.SERVICES, mis à part que le "4" ou "6" ne devrait pas être inclus dans l'entrée ETC.SERVICES.

sndbuf et rcvbuf indiquent la taille des mémoires tampon d'envoi et de réception. La taille, représentée par n, peut être exprimée en octets, ou un "k" ou un "m" peuvent être nécessaires pour indiquer respectivement des kilooctets ou des mégaoctets. sndbuf et rcvbuf peuvent être utilisés dans n'importe quel ordre.

**wait\_flag[.max]**

wait ou nowait.wait indique que le démon dispose d'une seule unité d'exécution et qu'une nouvelle requête ne sera pas traitée tant que la précédente n'aura pas été terminée. Si nowait est indiqué, INETD émet une acceptation lorsqu'une demande de connexion est reçue sur une prise pour flot de données. Si wait est indiqué, il est de la responsabilité du serveur d'émettre une acceptation, s'il s'agit d'une prise pour flot de données.

max est le nombre maximum d'utilisateurs autorisés à effectuer une demande de service dans un intervalle de 60 secondes. La valeur par défaut est 40. En cas de dépassement, le port du service est arrêté.

**userid[.group]**

userid est l'ID utilisateur que le démon en processus parallèle de traitement doit utiliser pour son exécution. Cet ID utilisateur peut être différent de celui d'INETD. Les autorisations affectées à cet ID utilisateur dépendent des besoins du service. L'ID utilisateur d'INETD nécessite une autorisation BPX.DAEMON pour faire passer le traitement par duplication à cet ID utilisateur.

La valeur facultative group, qui est séparée de l'ID utilisateur par un point (.), permet au serveur de s'exécuter avec un ID groupe différent de celui par défaut pour cet ID utilisateur.

**server\_program**

server\_program est le chemin d'accès complet du service. Par exemple :  
/usr/sbin/rlogind est le chemin d'accès complet de la commande rlogind.

**server\_program\_arguments**

20 arguments au maximum. Le premier argument est le nom du serveur.

---

## ETC.SERVICES

INETD utilise ETC.SERVICES pour mapper les numéros de port et les protocoles vers les services qu'ils doivent prendre en charge. Il peut s'agir d'un fichier MVS ou z/OS UNIX. Un exemple est fourni dans SEZAINST(SERVICES), qui est également disponible comme /usr/lpp/tcpip/samples/services. L'ordre de recherche pour ETC.SERVICES dépend de la méthode de démarrage d'INETD ; z/OS UNIX ou MVS natif.

La syntaxe suivante s'applique à la spécification d'informations de maintenance :

- Un fichier MVS ETC.SERVICES doit être fixe ou à bloc fixe avec un LRECL entre 56 et 256

- Un fichier HFS ETC.SERVICES peut avoir une longueur de ligne maximale de 256
- Les éléments sur une ligne sont séparés par des espaces ou des caractères de tabulation
- Chaque service figure en liste sur une seule ligne
- Un nom de service doit démarrer en première position sur une ligne
- La longueur maximale pour un nom de service et pour un nom d'alias est de 32 caractères
- Un maximum de 35 alias sera reconnu
- Les noms de service et d'alias sont sensibles à la casse
- Les commentaires commencent avec un dièse (#) ou un point-virgule (;) et se poursuivent jusqu'à la fin de la ligne

Chaque entrée se compose de 4 champs à position fixe, selon le format :

`service_name port_number/protocol aliases`

**service\_name**

Indique un nom de service identifié, ou défini par l'utilisateur.

**port\_number**

Indique le numéro de port de socket utilisé pour le service

**protocol**

Indique le protocole de transport utilisé pour le service. Les valeurs valides sont tcp et udp

**aliases**

Indique une liste de noms de service non officiels

## Ordre de recherche dans l'environnement z/OS UNIX

L'ordre de recherche permettant d'accéder à ETC.SERVICES dans z/OS UNIX est le suivant. La recherche s'arrête au premier fichier trouvé :

1. **/etc/services**
2. **userid.ETC.SERVICES**  
userid est l'ID utilisateur qui est utilisé pour démarrer INETD.
3. **hlq.ETC.SERVICES**  
hlq représente la valeur de l'instruction DATASETPREFIX spécifiée dans le fichier de configuration du programme de résolution de base (le cas échéant) ; sinon, hlq correspond au protocole TCPIP par défaut.

## Ordre de recherche dans l'environnement MVS natif

L'ordre de recherche permettant d'accéder à ETC.SERVICES dans le système MVS natif est le suivant. La recherche s'arrête au premier fichier trouvé :

1. **//SERVICES DD card**  
Le fichier alloué à l'instruction de définition de données SERVICES est utilisé.
2. **userid.ETC.SERVICES**  
userid est l'ID utilisateur qui est utilisé pour démarrer INETD.
3. **jobname.ETC.SERVICES**  
jobname correspond au nom indiqué dans l'instruction JCL JOB pour les travaux par lots ou le nom de la procédure pour une procédure démarrée.
4. **hlq.ETC.SERVICES**

hlq représente la valeur de l'instruction DATASETPREFIX spécifiée dans le fichier de configuration du programme de résolution de base (le cas échéant) ; sinon, hlq correspond au protocole TCPIP par défaut.

**Remarque :** Le démarrage d'INETD par l'intermédiaire de BPXPATCH ne résulte pas dans l'utilisation de l'ordre de recherche du système MVS natif, dans la mesure où BPXBATCH exécute la commande de démarrage dans l'environnement z/OS UNIX. L'ordre de recherche du système MVS natif est utilisé uniquement lors du démarrage d'un module de chargement MVS, comme SEZALOAD(FTP).

---

## Définitions de ports PROFILE.TCPIP

Ne confondez pas les définitions PORT (ou PORTRANGE) de PROFILE.TCPIP avec les ports définis dans ETC.SERVICES dans la mesure où ces définitions ont des buts différents. Les ports définis dans PROFILE.TCPIP sont utilisés par le protocole TCPIP pour voir si le port est réservé pour un service donné. ETC.SERVICES est utilisé par INETD pour mapper un port à un service.

Lorsqu'INETD reçoit une demande sur un port contrôlé, il produit un processus enfant parallèle (avec le service demandé), qui est appelé inetdx, où inetd est le nom de travail pour INETD (selon la méthode de démarrage) et où x est un chiffre.

Cela complique la réservation de ports, ainsi dans le cas où un port contrôlé INETD est réservé dans PROFILE.TCPIP, il est conseillé d'utiliser le nom de la procédure JCL qui est démarrée pour l'espace d'adresse de noyau z/OS UNIX afin de permettre à presque tout processus de se lier au port. Ce nom est généralement OMVS, sauf si un nom différent est explicitement indiqué dans le paramètre STARTUP\_PROC du membre parmlib BPXPRMxx.

La liste suivante explique comment déterminer le nom de travail, étant donné l'environnement dans lequel l'application est exécutée.

- Les applications exécutées à partir d'un lot utilisent le nom du travail par lots.
- Les applications lancées à partir de la console opérateur MVS utilisent le nom de la procédure démarrée (STC) comme nom de travail.
- Les applications exécutées à partir d'un ID utilisateur TSO utilisent cet ID utilisateur comme nom de travail.
- Les applications exécutées à partir de l'interpréteur de commandes de z/OS présentent normalement un nom de travail qui est l'ID utilisateur connecté plus un suffixe d'un caractère.
- Les utilisateurs autorisés peuvent exécuter des applications à partir de l'interpréteur de commandes de z/OS et utilisent la variable d'environnement \_BPX\_JOBNAME pour définir le nom de travail. Dans ce cas, la valeur indiquée pour la variable d'environnement est le nom de travail.
- Le nom de la procédure de langage JCL démarrée pour l'espace d'adresse de noyau des services système UNIX peut être utilisé pour permettre à n'importe quel demandeur du socket API bind() (sauf pour les utilisateurs de l'API en langage Pascal) de se lier à ce port. Ce nom est généralement OMVS, sauf si un nom différent est explicitement indiqué dans le paramètre STARTUP\_PROC du membre parmlib BPXPRMxx.
- Les applications z/OS UNIX démarrées par INETD utilisent le nom de travail du serveur INETD.



**Remarque :** Bien qu'il n'est pas conseillé de procéder de la sorte, les ports définis dans ETC.SERVICES peuvent être différents du numéro de port réservé pour le service dans PROFILE.TCPIP.

---

## /etc/inetd.pid

Le processus INETD crée un fichier temporaire, /etc/inetd.pid, qui contient l'ID processus (PID) du démon INETD donc l'exécution est en cours. Cette valeur de PID est utilisée pour identifier les enregistrements du journal système qui ont le processus INETD pour origine, et pour fournir la valeur de PID aux commandes qui en nécessitent une, comme arrêter. Elle est également utilisée comme mécanisme de verrouillage afin d'empêcher d'avoir plus de 1 processus INETD actif.

---

## Démarrage

La mise en oeuvre z/OS UNIX d'INETD est située par défaut dans /usr/sbin/inetd et prend en charge deux paramètres de démarrage facultatif et ne dépendant pas de la position :

`/usr/sbin/inetd [-d] [inetd.conf]`

**-d** Option de débogage. La sortie de débogage est inscrite dans stderr, qui peut être routé vers un fichier par le démon syslogd. Pour plus d'informations sur syslogd, voir *Communications Server IP Configuration Guide* (SC31-8775). Veuillez noter que lorsqu'il est démarré de cette manière, INETD ne génère pas de processus enfant parallèle pour démarrer un service.

### inetd.conf

Fichier de configuration. La valeur par défaut est /etc/inetd.conf

Il est conseillé de démarrer INETD au moment de l'IPL. La façon la plus courante de procéder est de le démarrer à partir de /etc/rc ou de /etc/inittab (à partir de z/OS 1.8 seulement). Il peut également être démarré à partir d'un travail ou d'une tâche démarrée en utilisant de BPXBATCH ou à partir d'une session de l'interpréteur de commandes d'un utilisateur ayant les droits d'accès correspondants.

## /etc/rc

Lorsqu'il est démarré à partir du script de shell d'initialisation de z/OS UNIX, /etc/rc, INETD utilise l'ordre de recherche de z/OS UNIX pour trouver ETC.SERVICES. Un exemple de fichier /etc/rc est fourni comme /samples/rc. Les exemples de commande suivants peuvent être utilisés pour démarrer INETD :

```
# Start INETD
_BPX_JOBNAME='INETD' /usr/sbin/inetd /etc/inetd.conf &
sleep 5
```

## /etc/inittab

z/OS 1.8 et au dessus met à disposition une méthode alternative, /etc/inittab, pour l'émission des commandes lors de l'initialisation de z/OS UNIX. /etc/inittab autorise la définition du paramètre respawn, qui redémarre automatiquement le processus lorsqu'il se termine (un WTOR est envoyé à l'opérateur pour un second redémarrage dans les 15 minutes). Lorsqu'il est démarré à partir de /etc/inittab, INETD utilise l'ordre de recherche de z/OS UNIX pour trouver ETC.SERVICES. Un exemple de fichier /etc/inittab est fourni comme /samples/inittab. L'exemple de commande suivant peut être utilisé pour démarrer INETD :

```
# Start INETD
inetd::respfrk:/usr/sbin/inetd /etc/inetd.conf
```

**Remarque :** Soyez conscient que le paramètre respfrk utilisé dans l'exemple transfèrera l'attribut respawn à tous les traitements par duplication, y compris RSE. Lorsque le client ferme la connexion, respawn la redémarre. Le serveur RSE s'arrêtera à nouveau plus tard, en raison du dépassement du délai d'attente. Pour plus d'informations concernant inittab, voir *UNIX System Services Planning (GA22-7800)*.

## BPXBATCH

La méthode de démarrage BPXBATCH fonctionne pour les travaux des utilisateurs et du programme STC. Veuillez noter qu'INETD est un processus d'arrière-plan, de sorte que l'étape BPXBATCH du démarrage d'INETD se termine en quelques secondes après le démarrage. Lorsqu'il est démarré par BPXBATCH, INETD utilise l'ordre de recherche de z/OS UNIX pour trouver ETC.SERVICES. Le langage JCL de la liste de la figure 11 est un exemple de procédure de démarrage d'INETD (L'étape KILL retire un processus INETD actif, s'il y en a un) :

```
//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 and STDENV are defaulted to /dev/null
//* INETD daemon output can be controlled by syslogd settings
//*
```

Figure 11. Langage JCL de démarrage d'INETD

**Remarque :** STDIN, STDOUT et STDERR doivent être des fichiers z/OS UNIX lorsqu'ils sont attribués. STDENV peut être soit un fichier MVS soit un fichier z/OS UNIX. Pour plus d'informations sur BPXBATCH, voir *UNIX System Services Command Reference (SA22-7802)*.

**Remarque :** inetd.conf peut être un fichier ou un membre MVS quand INETD est démarré par BPXBATCH. Pour ce cela, codez l'instruction PARM comme dans l'exemple suivant (utilisez des apostrophes simples (') uniquement)) :

```
// PARM='PGM /usr/sbin/inetd //'SYS1.TCPPARMS(INETCONF)'' &PRM'
```

## Session de Shell

Lorsqu'il est démarré à partir d'une session de shell, INETD utilise l'ordre de recherche de z/OS UNIX pour trouver ETC.SERVICES. Les exemples de commandes suivants peuvent être utilisés (par quelqu'un disposant des droits d'accès nécessaires) pour arrêter et démarrer INETD (# désigne l'invite z/OS UNIX) :

```
1. # ps -e | grep inetd
    7 ?          0:00 /usr/sbin/inetd
2. # kill 7
3. # _BPX_JOBNAME='INETD' /usr/sbin/inetd
```

**Remarque :** Cette méthode n'est pas recommandée pour le démarrage initial, /etc/rc ou /etc/inittab sont plus appropriées dans la mesure où elles sont exécutées lors de l'initialisation de z/OS UNIX.

---

## Sécurité

INETD est un processus z/OS UNIX et nécessite donc des définitions OMVS valides dans le logiciel de sécurité pour l'ID utilisateur associé à INETD. UID, HOME et PROGRAM doivent être définis pour l'ID utilisateur, ainsi que le GID pour le groupe par défaut de l'utilisateur. Si INETD est démarré par /etc/rc ou par /etc/inittab, l'ID utilisateur est hérité du noyau z/OS UNIX, il s'agit de OMVSKERN par défaut.

```
ADDGROUP OMVSGRP OMVS(GID(1))
ADDUSER OMVSKERN DFLTGRP(OMVSGRP) NOPASSWORD +
        OMVS(UID(0) HOME('/') PROGRAM('/bin/sh'))
```

INETD est un démon qui nécessite l'accès à des fonctions comme setuid(). L'ID utilisateur utilisé pour démarrer INETD nécessite ainsi des droits d'accès READ au profil BPX.DAEMON dans la classe FACILITY. Dans le cas où ce profil n'est pas défini, un UID 0 est obligatoire.

```
PERMIT BPX.DAEMON CLASS(FACILITY) ACCESS(READ) ID(OMVSKERN)
```

L'ID utilisateur INETD nécessite également des droits EXECUTE pour le programme inetd (/usr/sbin/inetd), des droits READ à vos fichiers inetd.conf et ETC.SERVICES et des droits WRITE à /etc/inetd.pid. Si vous désirez exécuter INETD sans UID 0, vous pouvez donner des droits d'accès CONTROL au profil SUPERUSER.FILESYS dans la classe UNIXPRIV afin de fournir les permissions nécessaires pour les fichiers z/OS UNIX.

Les programmes qui nécessitent des droits d'accès de démon doivent être contrôlés par programme si BPX.DAEMON est défini dans la classe FACILITY. Cela est déjà fait pour le programme INETD par défaut (/usr/sbin/inetd), mais doit être défini manuellement si vous utilisez une copie ou une version personnalisée. Utilisez la commande **extattr +p** pour rendre un fichier z/OS UNIX contrôlé par programme. Utilisez la classe RACF PROGRAM pour rendre un module chargeable MVS contrôlé par programme.

Les programmeurs système qui ont besoin de redémarrer INETD depuis leur session de shell démarreront INETD à l'aide de leurs autorisations. Ils doivent ainsi disposer de la même liste de permissions que les ID utilisateur INETD normaux. Ils doivent de plus disposer de permissions de lister et d'arrêter les processus INETD. Cela peut être accompli de plusieurs façons.

- UID 0

Cette solution n'est pas recommandée pour les ID utilisateur "humains" dans la mesure où il n'y a pas de restrictions liées à z/OS UNIX.

- Droits d'accès READ au profil BPX.SUPERUSER dans la classe FACILITY  
Permettent à l'utilisateur d'obtenir un UID 0 par l'intermédiaire de la commande **su**. Il s'agit de la configuration recommandée.
- Droits d'accès aux profils individuels qui couvrent les autorisations obligatoires.
  - Droits d'accès READ à SUPERUSER.PROCESS.GETPSENT dans la classe UNIXPRIV (pour la commande **ps**)
  - Droits d'accès READ à SUPERUSER.PROCESS.KILL dans la classe UNIXPRIV (pour la commande **kill**)
  - Droits d'accès READ à BPX.JOBNAME dans la classe FACILITY (pour la variable d'environnement **\_BPX\_JOBNAME**)

Pour plus d'informations sur les commandes **extattr** et **su**, voir *UNIX System Services Command Reference (SA22-7802)*. Pour plus d'informations concernant la classe UNIXPRIV et les profils BPX.\* dans la classe FACILITY, voir *UNIX System Services Planning (GA22-7800)*. Pour plus d'informations sur les définitions de segments OMVS et la classe PROGRAM, voir *Security Server RACF Security Administrator's Guide (SA22-7683)*.

---

## Exigences de Developer for System z

Developer for System z dépend d'INETD pour la configuration de la connexion client-hôte. Il impose des exigences supplémentaires à la configuration d'INETD décrite ci-avant.

### INETD

Les paramètres environnementaux d'INETD, qui sont transmis lors du démarrage du processus, ainsi que les permissions pour l'ID utilisateur d'INETD doivent être configurés convenablement pour qu'INETD puisse démarrer le serveur RSE.

- Si INETD est lancé par JCL à l'aide de BPXBATCH, la taille de la région doit être 0.
- Si INETD est démarré à partir d'une session de shell TSO/OMVS, la taille de la région TSO doit être 2096128 ou plus.
- Si INETD est démarré par `/etc/rc` ou par `/etc/inittab`, la taille de la région de SYS1.PROCLIB(BPX0INIT) est utilisée, sa valeur par défaut étant 0.
- Droits d'accès READ et EXECUTE aux répertoires d'installation de Developer for System z, qui est `/usr/lpp/wd4z/*` par défaut

### Démon RSE

Le démon RSE qui est démarré par INETD lorsqu'un client se connecte au port 4035 est utilisé pour procéder à l'authentification, démarrer le serveur RSE, et renvoyer le numéro de port pour la suite de la communication en retour vers le client. Pour cela, l'ID utilisateur qui est affecté au démon RSE (dans `inetd.conf`) nécessite les autorisations suivantes :

- Définitions OMVS valides dans le logiciel de sécurité ; UID, HOME et PROGRAM doivent être définis, ainsi que le GID pour le groupe par défaut de l'utilisateur
- Droits d'accès READ au profil BPX.DAEMON dans la classe FACILITY
- Des droits d'accès READ et EXECUTE aux répertoires d'installation de Developer for System z, qui est `/usr/lpp/wd4z/*` par défaut
- Des droits d'accès READ et EXECUTE au répertoire de configuration de Developer for System z, `/usr/lpp/wd4z/rse/lib/` par défaut, mais il est conseillé d'utiliser un répertoire différent, comme `/etc/wd4z/`

---

## Annexe E. Configuration du protocole SSL

Cette annexe vous aide à résoudre certains incidents fréquents qui peuvent survenir lors de la configuration du protocole SSL (Secure Socket Layer), ou au cours de la vérification et/ou de la modification d'une configuration existante.

Une communication sécurisée vous assure que votre partenaire de communication est bien qui il prétend être, et que la transmission des informations se fait d'une manière qui rend difficile toute interception et lecture des données par des tiers. Le protocole SSL fournit cette capacité dans un réseau TCP/IP. Il fonctionne par l'emploi de certificats numériques pour vous identifier et d'un protocole à clé publique pour chiffrer la communication. Pour plus d'informations sur les certificats numériques et le protocole à clé publique utilisés par SSL, voir *Security Server RACF Security Administrator's Guide (SA22-7683)*.

Les actions nécessaires pour la configuration des communications SSL pour Developer for System z varient largement d'un site à l'autre, selon les besoins exacts, la méthode de communication RSE employée, et ce qui est déjà disponible au niveau du site.

Dans la présente annexe, nous allons cloner les définitions RSE en cours, de manière à avoir une seconde connexion qui utilisera le protocole SSL. Les méthodes de connexion REXEC et par démon seront toutes les deux configurées pour SSL. Nous créerons également nos propres certificats de sécurité destinés à être utilisés par les différents composants de la connexion RSE. Tout cela s'effectue selon les étapes suivantes :

1. Clonez la configuration RSE existante
2. Déterminez quel(s) fichier(s) de clés employer
3. Créez un stock de clés avec keytool
4. Créez une base de données de clés (démon uniquement) soit avec RACF soit avec gskkyman
5. Activez le protocole SSL en mettant ssl.properties à jour
6. Testez la connexion

**Remarque :** Pour plus d'informations sur l'utilisation d'un certificat signé par une autorité de certification de confiance ou la configuration de votre propre autorité de certification, voir le livre blanc *Setting up SSL for RSE Communication (SC23-5816)* dans la bibliothèque internet de Developer for System z, <http://www-306.ibm.com/software/awdtools/devzseries/library/>.

Une convention d'attribution de nom uniforme est utilisée dans cette annexe :

- Certificat : wd4zrse
- Stockage de clés et de certificats : wd4zssl.\*
- Mot de passe : rsessl

La plupart des tâches décrites ci-après nécessitent des actions de votre part dans z/OS UNIX. Vous pouvez les effectuer en lançant la commande TSO **OMVS**. Utilisez la commande **exit** pour retourner à TSO.

---

## Clonage de la configuration RSE existante

Dans cette étape, une nouvelle instance du serveur RSE et du démon RSE est créée pour être exécutée en parallèle avec celle(s) qui existe(nt) déjà. De cette façon, les tests du protocole SSL ne perturberont pas les opérations normales. Comme il est conseillé dans l'«Enregistrement du fichier de configuration rsed.envvars dans un autre répertoire», à la page 38, les exemples de commandes ci-dessous supposent que les fichiers de configuration se trouvent dans /etc/wd4z/

```
$ 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 for System z RSE using SSL

ajout d'un service rsessl, en utilisant le port 4047
# oedit /etc/inetd.conf

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

ajout d'un démon rsessl, en utilisant le répertoire de configuration /etc/wd4z/ssl
# ps -e | grep inetd
7 ?          0:00 /usr/sbin/inetd
# kill 7
# _BPX_JOBNAME='INETD' /usr/sbin/inetd
# exit
$ netstat | grep 4047
INETD4  00016619 0.0.0.0..4047          0.0.0.0..0          Listen
```

Les commandes indiquées ci-dessus créent un sous-répertoire appelé ssl et le remplissent avec les fichiers de configuration obligatoires. Il n'y a pas (encore) besoin de procéder à des modifications de la configuration. Il est possible de partager le répertoire d'installation et les composants MVS dans la mesure où ils ne sont pas spécifiques au protocole SSL. Toutefois, un nouveau démon (rsessl) doit être défini, qui utilise les nouveaux fichiers de configuration. Le port 4047 est affecté au nouveau démon.

Pour plus d'informations concernant les actions présentées ci-avant, voir le Chapitre 4, «Activation des composants de Developer for System z z/OS UNIX», à la page 37.

---

## Détermination des fichiers de clés à employer

Les certificats d'identité et les clés de chiffrement/déchiffrement utilisés par le protocole SSL sont stockés dans un fichier de clés. Différentes implémentations de ce fichier de clés existent, selon le type d'application.

Le démon RSE est une application du système SSL, qui utilise un fichier de base de données de clés. Cette base de données de clés peut être un fichier physique créé par gskkyman ou un fichier de clés géré par votre logiciel de sécurité (RACF

par exemple). Le serveur RSE (qui est démarré par le démon ou par REXEC) est une application Java SSL, qui utilise un fichier de stock de clés créé par keytool. A l'heure actuelle, RACF ne présente pas de prise en charge prête à l'emploi pour Java SSL.

Ainsi, pour procéder à une connexion par l'intermédiaire de REXEC, nous avons uniquement besoin du fichier de stock de clés :

- stock de clés (keytool)

Pour réaliser la connexion par l'intermédiaire du démon, nous avons besoin à la fois du stock de clés et du fichier de la base de données de clés :

- stock de clés (keytool)
- base de données de clés (RACF ou gskkyman)

Pour plus d'informations sur RACF et sur les certificats numériques, voir *Security Server RACF Security Administrator's Guide (SA22-7683)*. Pour la documentation gskkyman, voir *Cryptographic System SSL Programming (SC24-5901)*. La documentation de keytool est disponible à l'adresse suivante : <http://java.sun.com/j2se/1.4.2/docs/tooldocs/solaris/keytool.html>.

---

## Création d'un stock de clés avec keytool

"keytool -genkey" génère un paire de clés (une clé publique et une clé privée associée). La clé publique est ensuite incorporée dans un certificat d'auto-signature X.509 v1, qui est stocké comme une chaîne de certificats à un seul élément. Cette chaîne de certificats ainsi que la clé privée sont stockées comme une entrée (identifiée par un alias) dans un (nouveau) fichier de stock de clés.

**Remarque :** Java doit être inclus dans vos répertoires de recherches de commandes. L'instruction suivante peut être nécessaire pour pourvoir exécuter keytool (/usr/lpp/java/J1.4 est le répertoire dans lequel Java est installé) : PATH=\$PATH:/usr/lpp/java/J1.4/bin

Toutes les informations peuvent être passées comme paramètres, mais en raison des limitations de longueur de la ligne de commande, une certaine interactivité est nécessaire.

```
$ 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
```



Le certificat d'auto-signature créé ci-avant est valide pendant environ 10 ans (sans compter les jours intercalaires des années bissextiles). Il est stocké dans /etc/wd4z/ssl/wd4zssl.jks avec l'alias wd4zrse. Son mot de passe (rsessl) est identique au mot de passe du stock de clés, ce qui est un élément préalable pour RSE.

Le résultat peut être vérifié par l'intermédiaire de l'option -list :

```
$ 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
```

---

## Création d'une base de données de clés (démon uniquement)

Comme il a été mentionné plus tôt, le démon est une application du système SSL, qui utilise un fichier de base de données de clés. Il peut s'agir soit d'un fichier physique créé par gskkyman soit d'un jeu de clés RACF. Les jeux de clés RACF sont la méthode favorite de gestion des clés privées et des certificats pour le système SSL.

**Remarque :** Le système SSL utilise ICFS (Integrated Cryptographic Service Facility) si celui-ci est disponible. ICSF met à disposition un support de chiffrement matériel qui est utilisé à la place des algorithmes logiciels du système SSL. Pour plus d'informations à ce sujet, voir *Cryptographic System SSL Programming (SC24-5901)*.

## Création d'un jeu de clés avec RACF

Ne suivez pas cette étape si vous utilisez gskkyman pour le système SSL.

La commande **RACDCERT** installe et maintient les clés privées et les certificats dans RACF. RACF prend en charge la gestion en groupe de multiples clés privées et certificats. Ces groupes sont des jeux de clés.

Pour plus de détails concernant la commande **RACDCERT**, voir *Security Server RACF Command Language Reference (SA22-7687)*.

```
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(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))
```



L'exemple ci-avant commence par la création des profils nécessaires et par l'autorisation d'accès de l'ID utilisateur OMVSKERN aux jeux de clés. L'ID utilisateur utilisé doit correspondre à celui codé dans /etc/inetd.conf pour le démon RSE du protocole SSL. L'étape suivante est la création d'un nouveau certificat auto-signé avec l'intitulé wd4zrse. Aucun mot de passe n'est nécessaire. Ce certificat est alors ajouté au jeu de clés nouvellement créé (wd4zssl.racf). Exactement comme avec le certificat, aucun mot de passe n'est nécessaire pour le jeu de clés. La durée de vie du certificat correspond à celle créé avec keytool.

Le résultat peut être vérifié par l'intermédiaire de l'option list :

```
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<
```

## Création d'une base de données de clés avec gskkyman

Ne suivez pas cette étape si vous utilisez RACF pour le système SSL.

gskkyman est un programme z/OS UNIX basé sur le shell, piloté par menus, qui crée, remplit et gère un fichier z/OS UNIX qui contient les clés privées, les demandes de certificats et les certificats. Ce fichier z/OS UNIX est une base de données de clés.

**Remarque :** Les instructions suivantes peuvent être nécessaires pour configurer l'environnement pour gskkyman. Pour plus d'informations à ce sujet, voir System SSL Programming (SC24-5901).

```
PATH=$PATH:/usr/lpp/gskssl/bin
export NLSPATH=/usr/lpp/gskssl/lib/nls/msg/En_US.IBM-1047/%N:$NLSPATH
export STEPLIB=$STEPLIB:SYS1.SIEALNKE
```

```
$ gskkyman
```

```
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

```

L'exemple ci-avant commence par la création d'une base de données de clés appelée `wd4zssl.kdb` avec le mot de passe `rsessl`. Une fois que la base de données existe, elle est remplie par la création d'un nouveau certificat, auto-signé, stocké sous le l'intitulé `wd4zrse` et avec le même mot de passe (`rsessl`) que celui utilisé pour la base de données de clés (il s'agit d'un élément prérequis par RSE).

`gskkyman` attribue un masque de contrôle des données de droits (très sûr) de 600 (seul le propriétaire a accès) à la base de données de clés. Mis à part le cas où le démon utilise le même ID utilisateur que le créateur de la base de données de clés, les droits doivent être définis d'une manière moins restrictive. Les masques 640 (le propriétaire a des droits d'accès en lecture/écriture, le groupe du propriétaire a des droits d'accès en lecture) ou 644 (le propriétaire a des droits d'accès en lecture/écriture, tout le monde a des droits d'accès en lecture) sont utilisables pour la commande **`chmod`**.

Ce résultat peut être vérifié en sélectionnant l'option **Show certificate information** dans le sous-menu **Manage keys and certificates** :

```

$ 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): rssl

#### 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

---

## Activation du protocole SSL par la mise à jour de ssl.properties

Maintenant que les certificats sont en place, RSE peut commencer à utiliser le protocole SSL. En fonction des définitions choisies lors des étapes précédentes, différentes valeurs doivent être fournies dans ssl.properties.

```
$ oedit ssl.properties
```

```
enable_ssl=true
```

Les valeurs valides sont true et false (valeur par défaut).

```
daemon_keydb_file=wd4zssl.racf
```

Nom de base de données de clés gskkyman ou nom de jeu de clés RACF.  
Nécessaire uniquement pour l'emploi d'un démon.

```
daemon_keydb_password=
```

Mot de passe pour la base de données de clés gskkyman ou blanc pour un jeu de clés RACF. Nécessaire uniquement pour l'emploi d'un démon.

```
daemon_key_label=wd4zrse
```

Intitulé gskkyman/RACF utilisé, s'il n'est pas défini comme la valeur par défaut. Doit être mis en commentaire si la valeur par défaut est utilisée.  
Nécessaire uniquement pour l'emploi d'un démon.

**server\_keystore\_file=wd4zssl.jks**  
nom de stock de clés de keytool.

**server\_keystore\_password=rssl**  
mot de passe de stock de clés de keytool.

---

## Test de la connexion

La configuration de l'hôte SSL est maintenant terminée, et peut être testée par la connexion au client Developer for System z. Dans la mesure où nous avons créé une nouvelle configuration pour l'utilisation avec SSL (par clonage d'une configuration existante), une nouvelle connexion doit être définie, à l'aide des caractéristiques suivantes :

- REXEC : avec le chemin d'accès /etc/wd4z/ssl
- démon : avec le port 4047

**Remarque :** Pour exécuter une application du système SSL (connexion par démon), SYS1.SIEALNKE doit être dans LINKLIST ou dans STEPLIB. Si vous préférez la méthode STEPLIB, ajoutez l'instruction suivante à la fin de rsed.envvars. Soyez toutefois conscient que l'utilisation de STEPLIB dans z/OS UNIX présente un impact négatif sur les performances, voir «Eviter l'emploi de STEPLIB», à la page 73.

- Si la dernière directive STEPLIB définie plus tôt dans rsed.envvars est identique à STEPLIB=NONE  
STEPLIB=SYS1.SIEALNKE
- Si la dernière directive STEPLIB définie plus tôt dans rsed.envvars n'est pas identique à STEPLIB=NONE  
STEPLIB=\$STEPLIB:SYS1.SIEALNKE

A la connexion, l'hôte et le client démarrent avec un protocole d'établissement de liaison afin de configurer un chemin d'accès sécurisé. L'échange de certificats fait partie de ce protocole d'établissement de liaison. Si le client Developer for System z ne reconnaît pas le certificat de l'hôte, il demande à l'utilisateur si le certificat est digne de confiance.

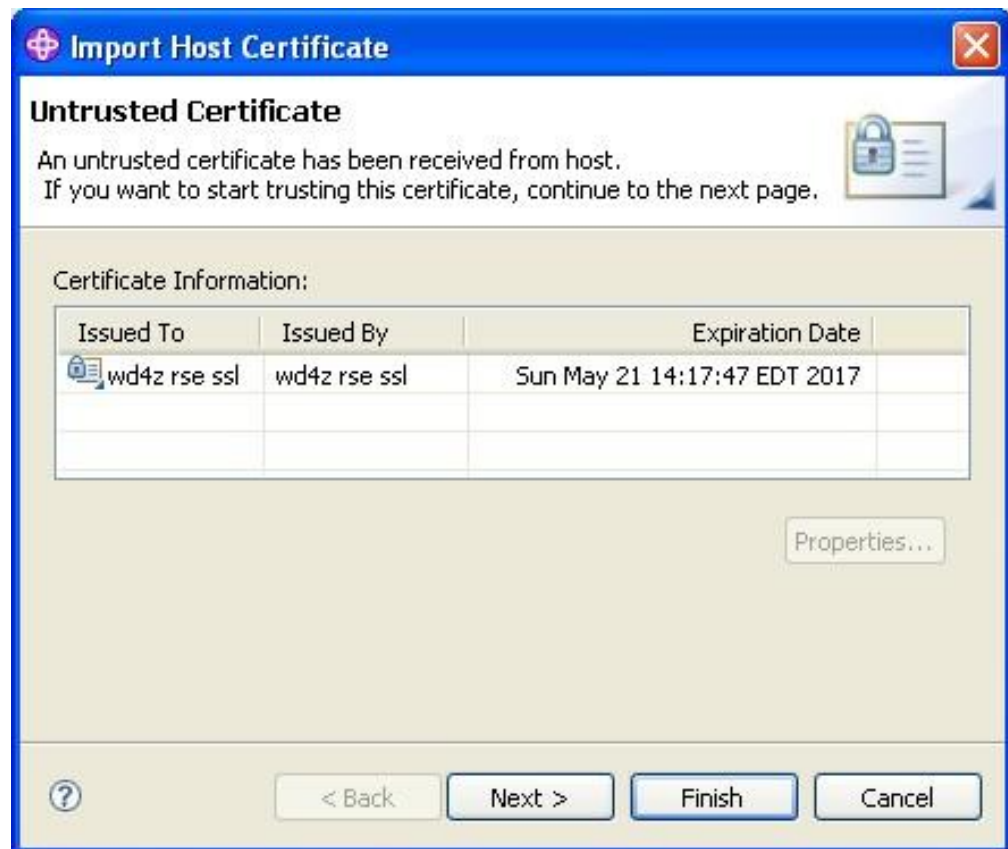


Figure 12. Importation du certificat hôte

En cliquant sur le bouton Finish, l'utilisateur peut accepter le certificat comme étant sécurisé, après quoi l'initialisation de la connexion se poursuit.

**Remarque :** La connexion du démon utilise 2 emplacements de certificats (Système SSL et Java SSL), ce qui donne 2 certificats différents et donc 2 confirmations.

Une fois que le client connaît le certificat, ce dialogue n'est plus affiché. La liste des certificats de confiance peut être gérée en sélectionnant **Windows > Préférences... > Système distant > SSL**, ce qui provoque l'affichage de la boîte de dialogue suivante :

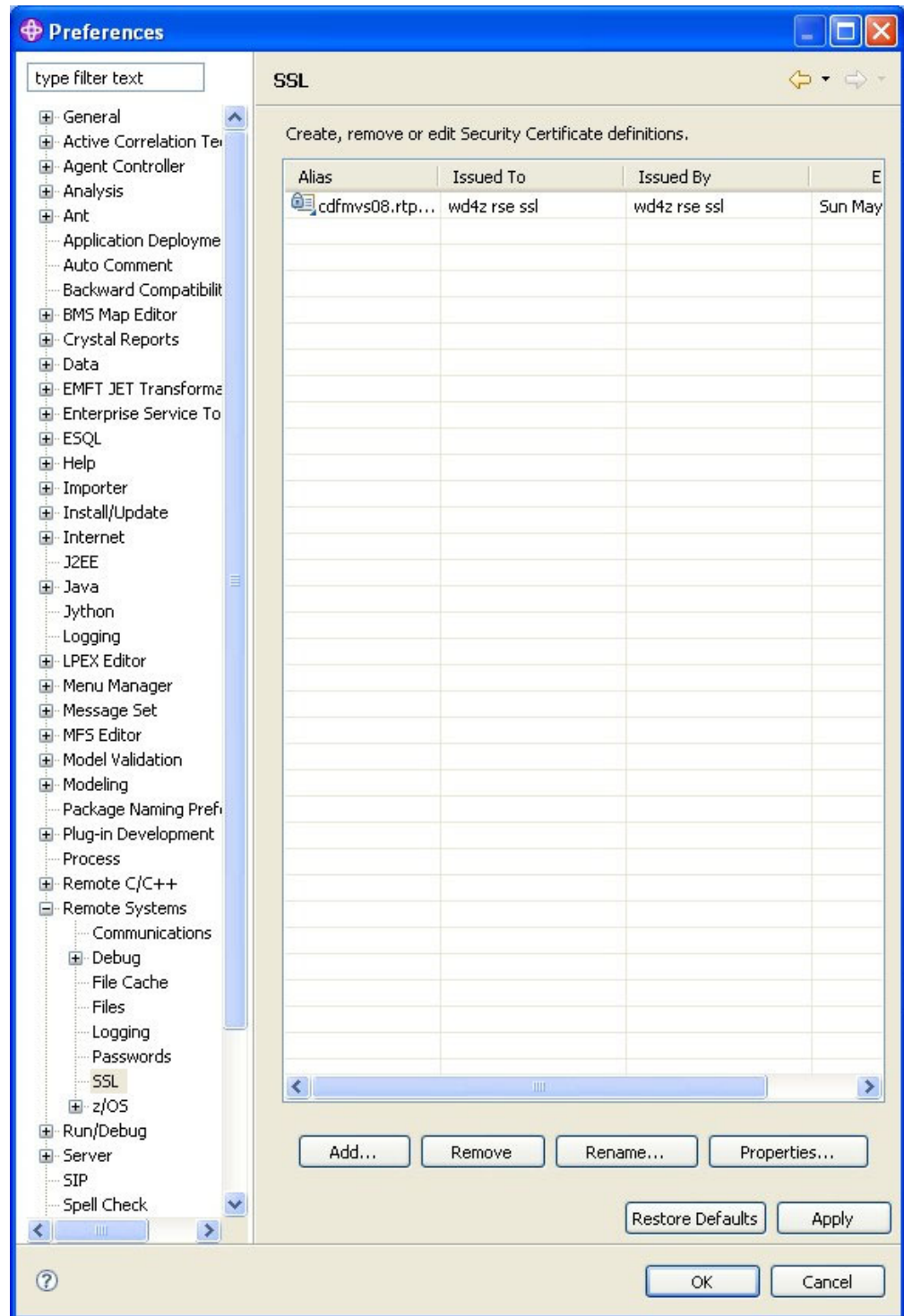


Figure 13. Préférences



En cas d'échec des communications SSL le client renvoie un message d'erreur. Plus d'informations sont disponibles dans les différents fichiers journaux (`home/.eclipse/RSE/USERID/*` et `/tmp/rsedaemon.log`), comme il est indiqué dans «Journalisation du serveur RSE», à la page 86.



---

## Annexe F. Configuration APPC

Cette annexe vous aide à résoudre certains incidents fréquents qui peuvent survenir lors de la configuration d'APPC (Advanced Program-to-Program Communication), ou au cours de la vérification et/ou de la modification d'une configuration existante.

Pour plus d'informations sur la gestion des APPC et sur les membres parmlib étudiés ci-dessous, consultez les ouvrages *MVS Planning APPC/MVS Management* (SA22-7599) et *MVS Initialization and Tuning Reference* (SA22-7592).

Veuillez noter que l'ensemble de la configuration d'APPC n'est pas abordé ici, mais uniquement certains aspects clés qui peuvent s'appliquer à votre site.

Le membre SYS1.SAMPLIB(ATBALL) contient une liste et des descriptions de tous les (exemples de) membres liés à APPC dans SYS1.SAMPLIB.

---

### Méthode d'accès VSAM

APPC/MVS stocke ses données de configuration dans des membres SYS1.PARMLIB et dans deux fichiers VSAM :

- Le fichier VSAM du programme de transaction (TP) (nom par défaut : SYS1.APPCTP) contient les informations de planification et de sécurité pour les programmes z/OS.
- Le fichier VSAM des informations complémentaires (SI) (nom par défaut : SYS1.APPCSI) contient la traduction des noms de destination symbolique utilisés par les serveurs APPC/MVS et TP locaux de z/OS.

Un TP est un programme d'application qui utilise APPC pour communiquer avec un TP sur le même système ou sur un autre afin d'accéder à des ressources. La configuration d'APPC pour Developer for System z active un nouveau TP appelé FEKFRSRV, qui est désigné comme le service Commandes TSO.

L'exemple de travail présenté dans la figure 14 est une concaténation des exemples de membres SYS1.SAMPLIB(ATBTPVSM) et SYS1.SAMPLIB(ATBSIVSM), et peut être utilisé pour définir les méthodes d'accès VSAM d'APPC.

```

//APPCVSAM JOB <job parameters>
//*
/* CAUTION: This is neither a JCL procedure nor a complete job.
/* Before using this sample, you will have to make the following
/* modifications:
/* 1. Change the job parameters to meet your system requirements.
/* 2. Change ***** to the volume that will hold the 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))
/*

```

Figure 14. Langage JCL pour créer des VSAM d'APPC

## VTAM

APPC est une implémentation du protocole architecture SNA LU 6.2. L'architecture SNA fournit des formats et des protocoles qui définissent une gamme de composants SNA physiques et logiques, comme l'unité logique (LU). LU 6.2 est un type d'unité logique qui est spécialement conçue pour gérer les communications entre les programmes d'application.

Pour utiliser SNA sur MVS, vous devez installer et configurer la méthode d'accès virtuelle en télétraitement VTAM (Virtual Telecommunications Access Method). VTAM doit être active avant que les tâches système APPC puissent être utilisées.

La partie spécifique à APPC de la configuration de VTAM se compose de trois étapes :

1. Définition du nom de mode utilisé (par exemple APPCHOST) pour VTAM à l'aide de SYS1.SAMPLIB(ATBLJOB) pour assembler et procéder à l'édition de liens de SYS1.SAMPLIB(ATBLMODE) dans votre SYS1.VTAMLIB. Pour plus de détails, voir le membre SYS1.SAMPLIB(ATBLMODE).
2. Définition d'APPC/MVS comme une application VTAM en copiant l'exemple de membre SYS1.SAMPLIB(ATBAPPL) dans un fichier dans la concaténation SYS1.VTAMLST. Pour plus de détails, voir le membre SYS1.SAMPLIB(ATBAPPL).

3. Utilisation de la commande de console **v net,act,id=atbappl** pour activer l'application nouvellement définie (où net est identique au nom de votre programme STC VTAM). Utilisez la commande de console **d net,appls** pour vérifier que l'application est active. Ajoutez le nom du membre à SYS1.VTAMLST(ATCCONxx) si vous voulez qu'il soit actif lorsque VTAM démarre.

L'ACBNAME de MVSLU01 utilisé dans l'exemple de membre SYS1.SAMPLIB(ATBAPPL) peut être modifié pour se conformer aux normes du site, mais doit correspondre aux définitions dans le membre SYS1.PARMLIB(APPCPMxx).

```
MVSLU01 APPL  ACBNAME=MVSLU01,      C
                APPC=YES,             C
                AUTOSES=0,             C
                DDRAINL=NALLOW,        C
                DLOGMOD=APPCHOST,       C
                DMINWNL=5,             C
                DMINWNR=5,             C
                DRESPL=NALLOW,         C
                DSESLIM=10,            C
                LMDENT=19,             C
                MODETAB=LOGMODES,       C
                PARSESS=YES,           C
                SECACPT=CONV,          C
                SRBEXIT=YES,           C
                VPACING=1              C
```

Figure 15. SYS1.SAMPLIB(ATBAPPL)

Pour plus d'informations concernant la configuration de VTAM, voir *Communications Server bookshelf (F1A1BK61 for z/OS 1.7)*.

---

## SYS1.PARMLIB(APPCPMxx)

Pour activer et prendre en charge le flux de conversations entre les systèmes, les sites doivent définir des unités logiques entre lesquelles les sessions peuvent être liées. Un site a besoin de définir au moins une unité logique avant le traitement APPC/MVS, même quand le traitement APPC demeure sur un seul système. Les unités logiques correspondent à certaines des définitions créées dans SYS1.PARMLIB(APPCPMxx).

Le service Commandes TSO requiert de configurer APPC de façon à disposer d'une unité logique qui puisse traiter à la fois les demandes entrantes et sortantes.

La définition d'unité logique doit être ajoutée au membre SYS1.PARMLIB(APPCPMxx) et inclure les paramètres BASE et SCHED(ASCH). Le membre APPCPMxx spécifie également le profil de transaction (TP) et les fichiers VSAM d'informations complémentaires (SI) qui seront utilisés.

La figure 16 est un exemple du membre SYS1.PARMLIB(APPCPMxx) que vous pouvez utiliser pour le service Commandes TSO.

```
LUADD ACBNAME(MVSLU01)
      BASE
      SCHED(ASCH)
      TPDATA(SYS1.APPCTP)
      SIDEINFO DATASET(SYS1.APPCSI)
```

Figure 16. SYS1.PARMLIB(APPCPMxx)

Lorsqu'un système comporte plusieurs noms d'unité logique, vous devrez peut-être apporter des modifications en fonction de l'unité logique sélectionnée par le système comme l'unité logique de base (BASE). L'unité logique de base du système est déterminée par les conditions suivantes :

1. L'unité logique de base du système est représentée par la dernière instruction LUADD qui contient à la fois les paramètres NOSCHED et BASE. Ce type d'unité logique permet de traiter les demandes sortantes lorsque aucun planificateur de transactions n'est actif.
2. Si aucune instruction LUADD contient à la fois NOSCHED et BASE, l'unité logique de base du système est représentée par la dernière instruction LUADD qui contient le paramètre BASE et spécifie ASCH comme le planificateur de transaction APPC/MVS. Cette action s'effectue par le codage de SCHED(ASCH) ou le non codage du paramètre SCHED (ASCH est la valeur par défaut de SCHED).

Si votre système comporte une unité logique contenant les paramètres BASE et NOSCHED, elle sera utilisée comme l'unité logique de base mais le Service Commandes TSO ne fonctionnera pas, car cette unité ne possède pas de planificateur de transactions pour traiter les demandes de transaction FEKFRSRV. Si cette unité logique ne peut pas être modifiée pour supprimer le paramètre NOSCHED, la variable d'environnement `rsed.envvars _FEKFSCMD_PARTNER_LU` peut être définie avec l'unité logique qui possède les paramètres BASE et SCHED(ASCH), par exemple :  
`_FEKFSCMD_PARTNER_LU=MVSLU01`

Pour plus d'informations sur `rsed.envvars`, voir «Personnalisation de `rsed.envvars`, le fichier de configuration de l'Explorateur de systèmes éloignés RSE», à la page 39.

---

## SYS1.PARMLIB(ASCHPMxx)

Le planificateur de transactions APPC/MVS (le nom par défaut est ASCH) lance et planifie des programmes de transaction en réponse aux demandes entrantes de conversations. Le membre `SYS1.PARMLIB(ASCHPMxx)` contrôle son fonctionnement, avec les définitions de classe de transaction par exemple.

La classe de transaction APPC utilisée pour le service Commandes TSO doit comporter suffisamment de demandeurs APPC pour en autoriser un par utilisateur de Developer for System z.

**Remarque :** Il existe une différence entre les demandeurs APPC et z/OS (JES). Quand un client Developer for System z se connecte à l'hôte, le serveur de commandes TSO est lancé à l'aide du demandeur APPC. Developer for System z utilise un demandeur z/OS (JES) quand un projet est généré, une vérification de la syntaxe est effectuée ou quand un travail est envoyé.

Le service Commandes TSO nécessite également que les spécifications par défaut soient indiquées aux sections `OPTIONS` et `TPDEFAULT`.

La figure 17, à la page 135 est un exemple du membre `SYS1.PARMLIB(ASCHPMxx)` que vous pouvez utiliser pour le service Commandes TSO.

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

```
OPTIONS  
  DEFAULT(A)
```

```
TPDEFAULT  
  REGION(2M)  
  TIME(5)  
  MSGLEVEL(1,1)  
  OUTCLASS(X)
```

Figure 17. SYS1.PARMLIB(ASCHPMxx)

**Remarque :** Pour les besoins du débogage, il se peut que le point service IBM vous demande d'augmenter la valeur de MSGLIMIT, de sorte qu'une plus grande quantité de sortie soit écrite dans le fichier journal.

---

## Activation des modifications d'APPC

Les modifications de configuration présentées dans les étapes ci-avant peuvent maintenant être activées. Il est possible de procéder de différentes manières, selon les circonstances.

- APPC n'est pas actif pour le moment. Saisissez les commandes de console suivantes pour lancer APPC/MVS (où xx représente les 2 derniers caractères du membre SYS1.PARMLIB associé) :

1. S APPC,SUB=MSTR,APPC=xx
2. S ASCH,SUB=MSTR,ASCH=xx

Ajoutez ces commandes à SYS1.PARMLIB(COMMNDxx) pour les lancer au démarrage du système.

- APPC est déjà actif. APPC peut recharger les membres SYS1.PARMLIB de manière dynamique par l'intermédiaire d'une commande de console SET (où xx représente les 2 derniers caractères du membre SYS1.PARMLIB associé) :
1. SET APPC=xx
  2. SET ASCH=xx

Les commandes de console **D APPC** et **D ASCH** peuvent être employées pour vérifier la configuration d'APPC. Pour plus d'informations sur les commandes mentionnées, voir *MVS System Commands (SA22-7627)*.

---

## Définition de la transaction du service Commandes TSO

Une fois qu'APPC/MVS est actif, le service Commandes TSO de Developer for System z peut être défini, voir «(Facultatif) Définition d'une transaction APPC pour le service Commandes TSO», à la page 26.

La façon de définir la transaction APPC décrite est par personnalisation et soumission de hlq.SFEKSAMP(FEKAPPC), où hlq est identique au qualificatif de haut niveau utilisé lors de l'installation de Developer for System z (la valeur par défaut est FEK).

La transaction APPC peut également être définie en mode interactif par l'intermédiaire de l'interface ISPF APPC, qui est décrit dans le livre blanc. Ce livre

blanc décrit également comment configurer la transaction APPC afin de collecter des informations statistiques spécifiques des utilisateurs.

Le livre blanc *APPC and WebSphere Developer for System z (SC23-5885)* est disponible auprès de la bibliothèque internet de Developer for System z, <http://www-306.ibm.com/software/awdtools/devzseries/library/>

**Remarque :** Le programme transactionnel (TP) en langage JCL qui est utilisé par APPC pour lancer le service Commandes TSO a changé dans Developer for System z version 7.1. Si vous suivez les instructions du livre blanc pour définir le TP, vous devez ajouter le mot-clé NESTMACS à la ligne PARM, par exemple :

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



---

## Glossaire

### A

**Action de verrouillage :** Verrouille un membre

**Attribut bidirectionnel :** Type de texte, orientation du texte, Permutation numérique et Permutation symétrique.

### B

**Base de données :** Collection d'éléments de données liés entre eux ou indépendants, stockés ensemble, destinée à être utilisée dans une ou plusieurs applications.

**Bibliothèque de chargement :** Bibliothèque contenant des modules de chargement.

**Bidirectionnel (bidi) :** Caractérise des scripts, tels que l'arabe et l'hébreu, qui s'exécutent généralement de droite à gauche, à l'exception des nombres, qui s'exécutent de gauche à droite. La définition de ce terme est extraite du glossaire Localization Industry Standards Association (LISA).

### C

**Compiler :**

1. Dans les langages Integrated Language Environment (ILE), traduire des instructions source en modules qui peuvent ensuite être associés en programmes ou programmes de service.
2. Traduire toute ou une partie d'un programme exprimé en langage haut niveau en un programme exprimé en langage intermédiaire, un langage d'assemblage ou un langage machine.

**Conteneur :**

1. Dans CoOperative Development Environment/400, objet système qui contient et organise des fichiers source. Par exemple, un conteneur peut être une bibliothèque i5/OS ou un fichier partitionné pour MVS.
2. Dans J2EE, entité qui fournit la gestion du cycle de vie, la sécurité, le déploiement et des services d'exécution pour les composants. (Sun) Chaque type de conteneur (EJB, Web, JSP, servlet, applet et client d'application) fournit également des services spécifiques aux composants.
3. Dans Backup Recovery and Media Services, objet physique utilisé pour stocker ou déplacer des supports, tels qu'une case, un chemin ou une armoire.

4. Dans un serveur Virtual Tape Server (VTS), réceptacle dans lequel un ou plusieurs volumes logiques exportés (LVOL) peuvent être stockés. Un volume empilé qui contient un ou plusieurs LVOL et qui réside en dehors d'une bibliothèque VTS est considéré comme le conteneur de ces volumes.
5. Lieu de stockage physique des données. Par exemple, un fichier, un répertoire ou un périphérique.
6. Colonne ou rang utilisé pour la disposition d'un portlet ou d'un autre conteneur sur une page.
7. Élément de l'interface utilisateur qui contient des objets. Dans le gestionnaire de dossier, objet qui contient les autres dossiers ou documents.

### D

**Déboguer :** Détecter, diagnostiquer et éliminer les erreurs des programmes.

**Demande de génération :** Demande du client pour effectuer une transaction de génération.

**Désinstallation en mode silencieux :** Processus de désinstallation qui n'envoie pas les messages vers la console mais stocke les messages et erreurs dans des fichiers journaux après que la commande d'installation a été appelée.

### F

**Fichier :** Unité principale de stockage et d'extraction des données, qui est constituée d'une collection de données disposée selon une des structures imposées et décrites par les données de contrôle auxquelles le système a accès.

**Fichier de réponses :**

1. Fichier qui contient un ensemble de réponses prédéfinies aux questions envoyées par un programme afin d'éviter d'entrer ces valeurs une par une.
2. Fichier ASCII qui peut être personnalisé au moyen des données de configuration pour automatiser une installation. Les données de configuration sont généralement entrées lors d'une installation interactive alors qu'un fichier de réponses permet d'effectuer l'installation sans aucune intervention.

### I

**ID action :** Identificateur numérique d'une action entre 0 et 999

**Interpréteur de commandes :** Interface logicielle entre les utilisateurs et le système d'exploitation qui interprète les commandes et les interactions utilisateur et les communique au système d'exploitation. Un ordinateur peut contenir plusieurs couches d'interpréteur de commandes (shell) pour différents niveaux d'interaction utilisateur.

**Interactive System Productivity Facility (ISPF) :** Logiciel sous licence IBM servant d'éditeur plein écran et de gestionnaire de boîte de dialogue. Utilisé dans l'écriture de programmes d'application, il permet de générer des panneaux d'écran standard et des boîtes de dialogue interactives entre le programmeur et l'utilisateur final. ISPF est constitué de quatre composants principaux : DM, PDF, SCLM et C/S. Le composant DM (Dialog Manager) est le gestionnaire qui fournit des services pour les boîtes de dialogue et les utilisateurs finals. Le composant PDF (Program Development Facility) offre des services d'aide au développeur de boîtes de dialogue ou d'applications. Le composant SCLM (Software Configuration Library Manager) offre aux développeurs d'applications des services destinés à leurs bibliothèques de développement d'applications. Le composant C/S (Client/Server), qui permet d'exécuter ISPF sur un poste de travail programmable, d'afficher les panneaux au moyen de la fonction d'affichage sur le système d'exploitation de votre poste de travail et d'intégrer des outils et données de poste de travail au moyen des outils et des données de l'hôte.

**Interpréteur :** Programme qui traduit et exécute successivement toutes les instructions en langage de programmation haut niveau.

**Installation en mode silencieux :** Installation qui n'envoie pas les messages vers la console mais stocke les messages et les erreurs dans des fichiers journaux. Une installation en mode silencieux peut également utiliser des fichiers de réponses pour entrer les données.

**Instance de référentiels :** Projet ou composant existant dans un SCM.

**Isomorphe :** A chaque élément composé (c'est-à-dire, contenant d'autres éléments) du document d'instance XML lancé à partir de la racine correspond un seul et unique élément de groupe COBOL dont la profondeur d'imbrication est identique à la profondeur d'imbrication de son équivalent XML. Chaque élément non composé (à savoir, ne contenant pas d'autres éléments) dans le document d'instance XML, en partant du haut, comporte une seule donnée élémentaire COBOL correspondante dont la profondeur d'imbrication est identique à la profondeur d'imbrication de son équivalent XML et dont l'adresse mémoire lors de l'exécution peut être identifiée de manière unique.

## L

**Liste des tâches :** Liste des procédures qui peuvent être exécutées par un seul flux de contrôle.

## M

**Mémoire tampon des erreurs :** Partie de mémoire servant à contenir provisoirement les données de sortie des erreurs.

## N

**Nom d'interpréteur de commandes :** Nom de l'interface de l'interpréteur de commandes.

**Non isomorphe :** Mappage simple d'éléments COBOL et d'éléments XML faisant partie de documents XML et de groupes COBOL de forme non identique (non isomorphe). Un mappage non isomorphe peut également être créé entre des éléments non isomorphes de structures isomorphes.

## P

**Passerelle :**

1. Composant intermédiaire qui relie Internet aux environnements intranet lors des appels de service Web.
2. Logiciel qui fournit des services entre les points d'arrêt final et le reste de l'environnement Tivoli.
3. Composant de Voice over Internet Protocol constituant un pont entre VoIP et les environnements commutés par circuit.
4. Périphérique ou programme utilisé pour la connexion de réseaux ou systèmes à d'autres architectures réseau. Ces systèmes peuvent présenter des caractéristiques différentes, telles que des protocoles de communication différents, une architecture réseau ou des stratégies de sécurité différentes, la passerelle réalisant alors leur traduction et leur connexion.

**Perspective :** Groupe de vues présentant les divers aspects des ressources d'un plan de travail. L'utilisateur du plan de travail peut naviguer entre les perspectives en fonction de la tâche qu'il traite et peut personnaliser la disposition des vues et des éditeurs d'une perspective.

**Perspective Systèmes distants :** Offre une interface permettant de gérer des systèmes distants par l'intermédiaire de conventions similaires à ISPF

## R

**RAM :** Repository Access Manager

### Référentiel :

1. Zone de stockage des données. Chaque référentiel comporte un nom et un type d'élément métier associé. Par défaut, son nom sera le même que celui de l'élément métier. Par exemple, un référentiel de factures sera appelé Factures. Il existe deux types de référentiels d'information : local (spécifique au processus) et global (réutilisable).
2. Fichier VSAM dans lequel sont stockés les états des processus du BTS. Lorsqu'un processus ne s'exécute pas sous le contrôle du BTS, son état (et les états des tâches qui le composent) sont protégés en écriture dans un fichier de référentiel. Les états de tous les processus d'un type de processus donné (et les instances de leurs tâches) sont stockés dans le même fichier de référentiel. Les enregistrements des types multiprocessus peuvent être écrits dans ce même référentiel.
3. Zone de stockage permanente du code source et des autres ressources d'application. Dans un environnement de programmation en équipe, un référentiel partagé permet à plusieurs utilisateurs d'accéder en même temps aux ressources de l'application.
4. Collection d'informations sur les gestionnaires de file d'attente qui sont membres d'un cluster. Ces informations comprennent les noms des gestionnaires de files d'attente, leurs emplacements, leurs canaux, les files d'attente qu'ils hébergent, etc.

## S

**Script de shell :** Fichier contenant des commandes qui peuvent être interprétées par l'interpréteur de commandes. L'utilisateur entre le nom du fichier script sur la ligne de commande de l'interpréteur et l'interpréteur exécute ensuite les commandes du script.

**Section de liaison :** Section de la division des données d'une unité activée (programme ou méthode appelé(e)) qui décrit les éléments de données disponibles à partir d'une unité d'activation (programme ou méthode). L'unité activée et l'unité d'activation peuvent toutes deux se référer à ces éléments de données.

### Serveur d'applications :

1. Programme qui traite toutes les opérations d'une application qui s'exécutent entre des ordinateurs dotés d'un navigateur et les applications dorsales ou bases de données de l'entreprise. Une classe spécifique de serveurs d'applications Java prend en charge la norme J2EE. Le code J2EE peut être facilement porté entre ces différents serveurs. Ils peuvent supporter des JSP et des servlets destinés

au contenu Web dynamique et des EJB pour les transactions et l'accès aux bases de données.

2. Cible d'une demande émise à partir d'une application distante. Dans l'environnement DB2, la fonction de serveur d'applications est remplie par la fonction de données réparties et permet d'accéder aux données DB2 à partir d'applications distantes.
3. Programme serveur dans un réseau réparti qui fournit l'environnement d'exécution d'un programme d'application.
4. Cible d'une demande émise par un demandeur d'application. Le système de gestion de base de données du site du serveur de l'application fournit les données demandées.
5. Logiciel qui traite les communications avec le client lorsque celui-ci demande un actif et les requêtes du Content Manager.

**Session de débogage :** Tâches de débogage exécutées entre l'heure à laquelle le développeur lance le débogage, et l'heure à laquelle il sort de l'application.

**Sidedeck :** Bibliothèque qui publie les fonctions d'un programme DLL. Les noms des entrées et des modules sont stockés dans la bibliothèque après la compilation du code source.

**Système distant :** Tout autre système du réseau avec lequel votre système peut communiquer.

**Système de fichiers distant :** Système de fichiers résidant sur un serveur ou un système d'exploitation séparé.

## T

**Transaction de génération :** Travail démarré sur MVS pour effectuer des générations après qu'une demande de génération a été reçue du client.

## U

**URL :** Uniform Resource Locator

## V

**Vue Console de sortie :** Affiche les données de sortie d'un processus et vous permet d'entrer à partir du clavier les données d'un processus.

**Vue de sortie :** Affiche les messages, les paramètres et résultats associés aux objets sur lesquels vous travaillez

**Vue Définition de données :** Affiche une image locale des bases de données, ainsi que des objets qu'elles contiennent. Elle fournit également les fonctions nécessaires pour manipuler ces objets et les exporter vers une base de données distante.

**Vue du navigateur :** Fournit une vue hiérarchique des ressources du plan de travail.

**Vue Référentiels :** Affiche les emplacements des référentiels CVS qui ont été ajoutés à votre plan de travail.

**Vue Serveurs :** Présente une liste de tous les serveurs et des configurations qui leur sont associées.

---

## Remarques

La présente documentation a été rédigée pour des produits et services proposés aux Etats-Unis.

IBM peut ne pas offrir les produits, services, ou fonctions traités dans ce document dans d'autres pays. Pour plus de détails, référez-vous aux documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial IBM. Toute référence à un produit, logiciel ou service IBM n'implique pas que seul ce produit, logiciel ou service IBM puisse être utilisé. Tout autre élément fonctionnellement équivalent peut être utilisé, s'il n'enfreint aucun droit d'IBM. Toutefois, il appartient à l'utilisateur d'évaluer et de vérifier le fonctionnement de produits, logiciels ou services non expressément référencés par IBM.

IBM peut détenir des brevets ou des demandes de brevet couvrant les produits mentionnés dans la présente documentation. La remise de ce document ne vous donne aucun droit de licence sur ces brevets ou demandes de brevet. Si vous désirez recevoir des informations concernant l'acquisition de licences, veuillez en faire la demande par écrit à l'adresse suivante :

*IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.*

Les informations sur les licences concernant les produits utilisant un jeu de caractères double octet peuvent être obtenues par écrit à l'adresse suivante :

*IBM World Trade Asia Corporation  
Licensing  
2-31 Roppongi 3-chome, Minato-ku  
Tokyo 106, Japan*

**Le paragraphe suivant ne s'applique ni au Royaume-Uni, ni à aucun pays dans lequel il serait contraire aux lois locales :** LE PRESENT DOCUMENT EST LIVRE "EN L'ETAT" SANS AUCUNE GARANTIE EXPLICITE OU IMPLICITE. IBM DECLINE NOTAMMENT TOUTE RESPONSABILITE RELATIVE A CES INFORMATIONS EN CAS DE CONTREFACON AINSI QU'EN CAS DE DEFAULT D'APTITUDE A L'EXECUTION D'UN TRAVAIL DONNE. Certaines juridictions n'autorisent pas l'exclusion des garanties implicites, auquel cas l'exclusion ci-dessus ne vous sera pas applicable.

Le présent document peut contenir des inexactitudes ou des coquilles. Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. IBM peut, à tout moment et sans préavis, modifier les produits et logiciels décrits dans ce document.

Les références à des sites Web non IBM sont fournies à titre d'information uniquement et n'impliquent en aucun cas une adhésion aux données qu'ils contiennent. Les éléments figurant sur ces sites Web ne font pas partie des éléments du présent produit IBM et l'utilisation de ces sites relève de votre seule responsabilité.

IBM pourra utiliser ou diffuser, de toute manière qu'elle jugera appropriée et sans aucune obligation de sa part, tout ou partie des informations qui lui seront fournies.

Les licenciés souhaitant obtenir des informations permettant : (i) l'échange des données entre des logiciels créés de façon indépendante et d'autres logiciels (dont celui-ci), et (ii) l'utilisation mutuelle des données ainsi échangées, doivent adresser leur demande à :

*IBM Corporation  
P.O. Box 12195, Dept. TL3B/B503/B313  
3039 Cornwallis Rd.  
Research Triangle Park, NC 27709-2195  
U.S.A.*

Ces informations peuvent être soumises à des conditions particulières, prévoyant notamment le paiement d'une redevance.

Le logiciel sous licence décrit dans cette documentation et tous les éléments sous licence disponibles s'y rapportant sont fournis par IBM conformément aux dispositions de l'IBM Customer Agreement, des Conditions internationales d'utilisation des logiciels IBM ou de tout autre accord équivalent.

Les données de performance indiquées dans ce document ont été déterminées dans un environnement contrôlé. Par conséquent, les résultats peuvent varier de manière significative selon l'environnement d'exploitation utilisé. Certaines mesures évaluées sur des systèmes en cours de développement ne sont pas garanties sur tous les systèmes disponibles. En outre, elles peuvent résulter d'extrapolations. Les résultats peuvent donc varier. Il incombe aux utilisateurs de ce document de vérifier si ces données sont applicables à leur environnement d'exploitation.

Les informations concernant des produits non IBM ont été obtenues auprès des fournisseurs de ces produits, par l'intermédiaire d'annonces publiques ou via d'autres sources disponibles. IBM n'a pas testé ces produits et ne peut confirmer l'exactitude de leurs performances ni leur compatibilité. Elle n'accepte aucune réclamation concernant des produits non IBM. Toute question concernant les performances de produits non IBM doit être adressée aux fournisseurs de ces produits.

Toute instruction relative aux intentions d'IBM pour ses opérations à venir est susceptible d'être modifiée ou annulée sans préavis, et doit être considérée uniquement comme un objectif.

Le présent document peut contenir des exemples de données et de rapports utilisés couramment dans l'environnement professionnel. Ces exemples mentionnent des noms fictifs de personnes, de sociétés, de marques ou de produits à des fins illustratives ou explicatives uniquement. Toute ressemblance avec des noms de personnes, de sociétés ou des données réelles serait purement fortuite.

#### LICENCE DE COPYRIGHT :

Le présent logiciel contient des exemples de programmes d'application en langage source destinés à illustrer les techniques de programmation sur différentes plateformes d'exploitation. Vous avez le droit de copier, de modifier et de distribuer ces exemples de programmes sous quelque forme que ce soit et sans paiement d'aucune redevance à IBM, à des fins de développement, d'utilisation, de vente ou de distribution de programmes d'application conformes aux interfaces de programmation IBM. Ces exemples de programmes n'ont pas été rigoureusement testés dans toutes les conditions. Par conséquent, IBM ne peut garantir expressément ou implicitement la fiabilité, la maintenabilité ou le fonctionnement de ces programmes. Vous avez le droit de copier, de modifier et de distribuer ces exemples de programmes sous quelque forme que ce soit et sans paiement d'aucune redevance à IBM, à des fins de développement, d'utilisation, de vente ou de distribution de programmes d'application conformes aux interfaces de programmation IBM.

Toute copie totale ou partielle de ces programmes exemples et des oeuvres qui en sont dérivées doit comprendre une notice de copyright libellée comme suit :

© (nom de votre société) (année). Les segments de code sont dérivés des Programmes exemples d'IBM Corp. © Copyright IBM Corp. \_Saisissez l'année ou les années\_. All rights reserved.

---

## Marques et logos

Les termes qui suivent sont des marques d'International Business Machines Corporation aux Etats-Unis et/ou dans certains autres pays :

- CICS
- CICSplex
- DB2
- IBM
- IMS
- MVS
- OS/390
- RACF
- Rational
- System z
- VTAM
- WebSphere
- z/OS
- zSeries

Java ainsi que toutes les marques et logos incluant Java sont des marques de Sun Microsystems, Inc. aux Etats-Unis et/ou dans certains autres pays.

Microsoft, Windows, Windows NT, et le logo Windows sont des marques de Microsoft aux Etats-Unis et/ou dans certains autres pays.

UNIX est une marque enregistrée de The Open Group aux Etats-Unis et/ou dans certains autres pays.

Les autres noms de sociétés, de produits et de services peuvent appartenir à des tiers.









Numéro de programme : 5724-T07

SC11-6285-00

