

IBM Rational Developer para System z



Guia de Configuração do Host

Versão 7.1

IBM Rational Developer para System z



Guia de Configuração do Host

Versão 7.1

Nota

Antes de utilizar este documento, leia as informações gerais em “Avisos” na página 127.

Primeira Edição (Setembro de 2007)

Esta edição se aplica ao IBM Rational Developer para System z Versão 7.1 (número de programa 5724-T07) e a todos os releases e modificações subseqüentes até que o contrário seja indicado em novas edições.

Solicite as publicações pelo telefone ou fax. O IBM Software Manufacturing Solutions recebe os pedidos de publicações entre 8h30 e 19h, horário padrão na costa leste dos EUA. O número de telefone é (800) 879-2755. O número de fax é (800) 445-9269. O fax deve ser enviado para: Publications, 3rd floor.

Você também pode solicitar as publicações através de um representante IBM ou da filial da IBM que atende em sua região. As publicações não são guardadas no endereço abaixo.

A IBM agradece pelo seu comentário. Você pode enviar os comentários pelo correio ao seguinte endereço:

IBM Brasil - Centro de Traduções
Rodovia SP 101 km 09
CEP 13185-900
Hortolândia, SP

Ao enviar informações à IBM, você concede à IBM o direito não-exclusivo de utilizar ou distribuir as informações da forma que julgar apropriada, sem incorrer em qualquer obrigação para com o Cliente.

Nota para Usuários do Governo dos Estados Unidos - Uso, duplicação ou divulgação restritos pelo documento GSA ADP Schedule Contract com a IBM Corp.

© Copyright International Business Machines Corporation 2005, 2007. Todos os direitos reservados.

Índice

Figuras	vii
--------------------------	------------

Tabelas	ix
--------------------------	-----------

Sobre Este Manual	xi
------------------------------------	-----------

Quem Deve Ler este Manual.	xi
------------------------------------	----

Capítulo 1. Instalando e Configurando os Componentes do Host 1

Considerações de Pré-instalação	2
Considerações de Pré-configuração	3
Configuração Requerida de Produtos e Software de Requisito	3
Considerações de ID do Usuário.	4
Considerações do Servidor.	5
Permissões Requeridas para Implementar as Tarefas de Configuração	6
Considerações de Pré-implementação	6
IBM Rational Developer para System z, FMID HHOP710	6
IBM CARMA, FMID HCMA710	7

Capítulo 2. Alterações na Instalação e na Configuração 9

Alterações entre a Versão 7.0 e a Versão 7.1	9
IBM Rational Developer para System z, FMID HHOP710	9
IBM CARMA, FMID HCMA710	9
Alterações entre a Versão 6.0.1 e a Versão 7.0	10
IBM WebSphere Developer for System z, FMID HHOP700	10
IBM CARMA, FMID HCMA700	10
Fazendo Backup de Arquivos Configurados Anteriormente	11

Capítulo 3. Ativando os Componentes MVS do Developer for System z 15

Configurar MAXASSIZE em SYS1.PARMLIB(BPXPRMxx).	15
hlq.SFEKLOAD Autorizada por APF	15
Customizar o FEJJCNFG, o Arquivo de Configuração do JES Job Monitor	16
Customizar o JCL de Inicialização do JES Job Monitor	18
Rastrear o JES Job Monitor	18
Executar o JES Job Monitor como uma STC.	18
Permissões do Servidor	20
Verificação do JCL de inicialização do JES Job Monitor	20
Acesso e Segurança do Spool do JES	20
Acesso Condicional ao Spool	20
Comandos Disponíveis	21
Limitando o Acesso aos Arquivos do Spool.	21

Customizar ELAXF*, Procedimentos de Construção Remota.	21
(Opcional) Definir uma Transação APPC para o Serviço de Comandos do TSO	23
Preparação	23
Implementação	25
(Opcional) Customizar ELAXM*, Membros do Procedimento Armazenado do DB2	26
(Opcional) Customizar o Suporte à Linguagem Bidirecional (BIDI) do CICS	27
(Opcional) Customizar o ADM (Application Deployment Manager).	28
Repositório do CRD	29
Região de Conexão Primária do CICS	30
Manipulador de Mensagem de Pipeline	30
(Opcional) Regiões de Conexão Não-primária do CICS	30

Capítulo 4. Ativando Componentes do z/OS UNIX para o Developer for System z 33

Salvando o Arquivo de Configuração rsed.envvars em um Outro Diretório	34
Customizar rsed.envvars, o Arquivo de Configuração para RSE	35
(Opcional) Definindo o PORTRANGE Disponível para RSE	40
(Opcional) Definindo Parâmetros Adicionais de Inicialização Java com _RSE_*OPTS	41
Configuração do Daemon INETD e REXEC/SSH do RSE	42
Configuração do Daemon INETD RSE	43
Configuração do INETD REXEC (ou SSH)	44
Customizar ISPF.conf, o Arquivo de Configuração do ISPF.	45
Verificar a Configuração do Servidor RSE	46
Disponibilidade de Porta	47
Conexão REXEC.	47
Script de Shell REXEC/SSH	48
Conexão do Daemon RSE	48
Conexão do JES Job Monitor.	49
Conexão do Serviço Comandos do TSO (Utilizando SCLMDT)	49
Conexão do Serviço Comandos do TSO (Utilizando APPC)	50
(Opcional) Customizar ssl.properties, Configuração do RSE SSL	51
(Opcional) Customizar rsecomm.properties, Configuração do Rastrear RSE	51
(Opcional) Customizar projectcfg.properties, Configuração de Projetos do Host	52
(Opcional) Customizar FMEXT.properties, Integração do File Manager	53

Capítulo 5. (Opcional) Ativando o IBM CARMA	55
Customizando os Componentes MVS do CARMA	55
Customizando os Componentes do z/OS UNIX para o CARMA	56
(Opcional) Ativando os RAM (Repository Access Managers) de Amostra	58
Ativando o RAM do SCLM	58
Ativando o PDS RAM	58

Capítulo 6. (Opcional) Ativando o IBM Software Configuration and Library Manager (SCLM) Developer Toolkit	61
--	-----------

Capítulo 7. Considerações do Cliente do Developer for System z	63
---	-----------

Capítulo 8. Considerações de Desempenho	65
Evite o Uso de STEPLIB	65
Aprimorar o Acesso às Bibliotecas do Sistema	65
Bibliotecas de Tempo de Execução do LE (Language Environment)	65
Desenvolvimento do Aplicativo	66
Aprimorando o Desempenho da Verificação de Segurança	66
Compartilhamento de Classe entre JVMs	67
Ativar Compartilhamento de Classe	67
Limites de Tamanho do Cache	67
Segurança do Cache	68
SYS1.PARMLIB(BPXPRMxx)	68
Espaço em Disco	68
Utilitários de Gerenciamento do Cache	68
Tamanho de Heap Java Fixo	69
Gerenciamento de Carga de Trabalho	70

Apêndice A. Executando Várias Instâncias do Developer for System z	71
Arquivos de Configuração Diferentes de Níveis de Software Idênticos	71
Todas as Outras Situações	72

Apêndice B. Resolvendo Problemas de Configuração	75
Localização dos Arquivos de Log	75
Criação de Logs do JES Job Monitor	76
Criação de Logs de Transações APPC (Serviço de Comandos do TSO)	76
Criação de Logs do RSE	76
Criação de Log IVP fekfivpc	77
Criação de Log de Integração do Fault Analyzer	77
Criação de Log de Integração do File Manager	77
Criação de Logs do CARMA	78
Arquivos de Dump	78
Dumps do MVS	78
Dumps Java	78
Locais de Dump do z/OS UNIX	80

Autorização do Controle de Programas para Programas RSE	80
Portas TCP/IP Reservadas	81
Tamanho do Espaço de Endereço	82
Requisitos do INETD	82
Limitações Definidas em SYS1.PARMLIB(BPXPRMxx)	82
Limitações Armazenadas no Perfil de Segurança	82
Limitações Impostas por Saídas do Sistema	83
Rastreamento de Feedback de Erro	83
Transação APPC e Serviço de Comandos do TSO	84
Informações Diversas	85
Limites do Sistema	85
Problemas Conhecidos	86
Emulador de Conexão do Host	87
Entrando em Contato com o Suporte IBM	87

Apêndice C. Configurando o TCP/IP	89
Dependência do Nome do Host	89
Compreendendo os Resolvedores	89
Compreendendo as Ordens de Procura das Informações de Configuração	90
Ordens de Procura Utilizadas no Ambiente z/OS UNIX	90
Arquivos de Base da Configuração do Resolvedor	90
Tabelas de Conversão	91
Tabelas do Host Local	91
Aplicando Isto no Developer for System z	92

Apêndice D. Configurando o INETD	97
inetd.conf	97
ETC.SERVICES	98
Ordem de Procura Utilizada no Ambiente z/OS UNIX	99
Ordem de Procura Utilizada no Ambiente MVS Ambiente	99
Definições de Portas PROFILE.TCPIP	100
/etc/inetd.pid	101
Inicialização	101
/etc/rc	101
/etc/inittab	101
BPXBATCH	102
Sessão de Shell	102
Segurança	103
Requisitos do Developer for System z	104
INETD	104
Daemon do RSE	104

Apêndice E. Configurando o SSL	105
Clonar a Configuração RSE Existente	106
Determinar o(s) Arquivo(s) de Chaves a Ser(em) Utilizado(s)	106
Criar um Armazenamento de Chaves com keytool	107
Criar um Banco de Dados de Chaves (daemon apenas)	108
Criar um Anel de Chave com o RACF	108
Criar um Banco de Dados de Chaves com gskkyman	109
Ativar o SSL, Atualizando ssl.properties	113

Testar a Conexão	114
----------------------------	-----

Apêndice F. Configurando o APPC 117

VSAM.	117
VTAM.	118
SYS1.PARMLIB(APPCPMxx)	119
SYS1.PARMLIB(ASCHPMxx)	120
Ativando Alterações do APPC.	121

Definindo a Transação de Serviço Comandos do TSO	121
---	-----

Glossário . 123

Avisos . 127

Marcas Registradas e Marcas de Serviço	129
--	-----

Figuras

1. FEJCNFG, arquivo de configuração do JES Job Monitor 16
2. JCL do JES Job Monitor 19
3. Painéis do REXX para o APPC ISPF 24
4. rsed.envvars – arquivo de configuração RSE 36
5. ISPF.conf - Arquivo de Configuração do ISPF 45
6. ssl.properties – Arquivo de configuração SSL 51
7. rsecomm.properties – Arquivo de configuração da criação de log. 52
8. projectcfg.properties – Arquivo de configuração de projetos baseados em host 53
9. FMIEXT.properties – Arquivo de configuração do File Manager 54
10. CRASRV.properties – Arquivo de configuração CARMA 57
11. JCL de Inicialização do INETD. 102
12. Importar Certificado do Host 115
13. Preferências 116
14. JCL para Criar VSAMs do APPC 118
15. SYS1.SAMPLIB(ATBAPPL) 119
16. SYS1.PARMLIB(APPCPMxx) 119
17. SYS1.PARMLIB(ASCHPMxx) 120

Tabelas

1. Matriz de Instalação e Configuração do Developer for System z	1	6. Procedimento ELAXF* de amostra	22
2. Visão geral de membros do MVS customizado	11	7. Lista de verificação de transações APPC	24
3. Visão geral dos arquivos z/OS UNIX customizados	13	8. Membros do procedimento armazenado DB2 do ELAXM* de amostra	26
4. Customização em bibliotecas não-Developer for System z	14	9. Arquivos de Configuração Opcionais	34
5. Comandos do Console do JES Job Monitor	21	10. Lista de verificação do cliente do Developer for System z	63
		11. Definições locais disponíveis para o resolvidor	94

Sobre Este Manual

Este manual discute a configuração das funções do IBM Rational Developer para System z. Ele inclui instruções sobre como configurar os servidores IBM Rational Developer para System z no sistema de host z/OS.

De agora em diante, os seguintes nomes serão utilizados neste manual:

- O *IBM Rational Developer para System z* é chamado de *Developer for System z*
- O *IBM Rational Developer para System z Common Access Repository Manager* é chamado de *Common Access Repository Manager*, algumas vezes abreviado como *CARMA*
- O *IBM SCLM (Software Configuration and Library Manager) Developer Toolkit* é chamado de *SCLM Developer Toolkit*, algumas vezes abreviado como *SCLMDT*

Nota: As informações de configuração encontradas neste documento se referem ao IBM Rational Developer para System z Versão 7.1.

Para o IBM WebSphere Developer for System z, o IBM WebSphere Developer for zSeries e o WebSphere Studio Enterprise Developer, utilize as informações de configuração encontradas no Guia de Configuração do Host e Diretórios do Programa desses releases.

Quem Deve Ler este Manual

Este manual se destina a programadores de sistema instalando e configurando o IBM Rational Developer para System z Versão 7.1 no sistema de host z/OS. Para utilizar este manual, você deve estar familiarizado com os sistemas host z/OS UNIX e MVS.

Capítulo 1. Instalando e Configurando os Componentes do Host

Para cada uma das seguintes funções, instale os FMIDs necessários. Para obter informações de instalação sobre os vários FMIDs, consulte o diretório de programa correspondente ao FMID que você está instalando.

Tabela 1. Matriz de Instalação e Configuração do Developer for System z

Se você necessitar desta função do IBM Rational Developer para System z	Deverá instalar estes FMIDs	E encontrará aqui, informações de instalação e configuração
<ul style="list-style-type: none">• Conectividade do Host• Conectividade JES• Compilação Remota (Remote Compile)• Feedback de Erro da Compilação Remota• Depuração Remota• Procedimentos Armazenados do DB2• Suporte a Tela MFS IMS• Suporte a Mapas BMS no CICS• Suporte à linguagem bidirecional (BIDI) do CICS• ADM (Application Deployment Manager)• Integração do File Manager• Integração do Fault Analyzer	HHOP710, HSD3310*	<ul style="list-style-type: none">• <i>Program Directory for IBM Rational Developer para System z</i> (GI11-8298-00)• <i>IBM Rational Developer para System z Host Configuration Guide</i> (SC31-6930)• <i>IBM Rational Developer para System z Host Planning Guide</i> (GI11-8296-00) opcional <ul style="list-style-type: none">• <i>Program Directory for IBM Software Configuration and Library Manager (SCLM) Developer Toolkit</i> (GI11-8302-00)• <i>SCLM Developer Toolkit Installation and Customization Guide</i> (SC23-8504)
<ul style="list-style-type: none">• CARMA (Common Software Configuration Management Access)	HCMA710, HHOP710**	<ul style="list-style-type: none">• <i>Program Directory for IBM Rational Developer para System z Common Access Repository Manager</i> (GI11-8299-00) opcional <ul style="list-style-type: none">• <i>Program Directory for IBM Rational Developer para System z</i> (GI11-8299-00)• <i>IBM Rational Developer para System z Host Configuration Guide</i> (SC31-6930)• <i>IBM Rational Developer para System z Host Planning Guide</i> (GI11-8296-00)
<ul style="list-style-type: none">• SCLM (Software Configuration and Library Manager) Developer Toolkit	HSD3310	<ul style="list-style-type: none">• <i>Program Directory for IBM SCLM Developer Toolkit</i> (GI11-8302-00)• <i>SCLM Developer Toolkit Installation and Customization Guide</i> (SC23-8504)

(*) O Developer for System z requer uma conexão com o serviço Comandos do TSO no z/OS. Essa conexão é fornecida por um dos seguintes:

1. Instalação e configuração do SCLM Developer Toolkit, FMID HSD3310 (o padrão)
2. Instalação e configuração de uma transação APPC

(**) CARMA requer uma interface baseada em host. O HHOP710 oferece essa função ou você pode utilizar uma interface baseada em host integrada customizada e omitir a instalação do HHOP710

Considerações de Pré-instalação

- Se você for um usuário da versão anterior do IBM WebSphere Developer for System z, IBM WebSphere Developer for zSeries ou WebSphere Studio Enterprise Developer, é recomendável salvar os arquivos customizados relacionados antes de instalar o upgrade para IBM Rational Developer para System z Versão 7.1. Consulte “Fazendo Backup de Arquivos Configurados Anteriormente” na página 11 para obter uma visão geral dos arquivos que precisam ser customizados.
- Leia o Apêndice A, “Executando Várias Instâncias do Developer for System z”, na página 71 se você planeja executar várias instâncias do Developer for System z.
- Na versão 7.1, o Developer for System z oferece dois métodos de acesso para o serviço Comandos do TSO, que executa comandos TSO/ISPF, implicitamente pelo produto ou explicitamente pelo usuário.

- Chamando uma função do SCLM Developer Toolkit. Esse método requer a instalação e configuração do SCLMDT, FMID HSD3310, que é fornecido com o Developer for System z. Esse método é o padrão utilizado nos membros de amostra.

As etapas de customização SCLMDT a seguir devem ser desempenhadas e estão descritas no *SCLM Developer Toolkit Installation and Customization Guide* (SC23-8504):

- Crie a estrutura de diretórios e os arquivos de configuração
- Customize ISPF.conf

Nota:

- O cliente baseado no Eclipse não é necessário para suportar essa função do SCLMDT
- O produto SCLM não é necessário para suportar essa função do SCLMDT
- Uma transação APPC (como em releases anteriores à versão 7.1). Esse método requer a definição e configuração do APPC, que faz parte do z/OS.
- O módulo de carregamento do bidi foi alterado na versão 7.0 (nome e codificação) e requer a remoção dos módulos mais antigos e a configuração do novo.
- O código Bidi que foi criado com releases anteriores (antes da versão 7.0) deve ser RECOMPILADO para utilizar o novo módulo FEJBDTRX.

Para obter instruções detalhadas sobre a instalação do SMP/E para cada um dos FMIDs, consulte o diretório de programas apropriado, conforme listado em Tabela 1 na página 1.

Considerações de Pré-configuração

O Developer for System z possui uma lista de softwares obrigatórios que devem ser instalados e estar em funcionamento para que o produto funcione. Há também uma lista de software de co-requisito para suportar recursos específicos do Developer for System z. Esses requisitos devem ser instalados e estar em funcionamento no tempo de execução para que o recurso correspondente funcione conforme projetado.

Consulte o *IBM Rational Developer para System z Host Planning Guide (GI11-8296-00)* para obter uma lista de pré-requisitos e co-requisitos para a sua versão do Developer for System z.

Atenção: Java 64 bits NÃO é suportado.

Configuração Requerida de Produtos e Software de Requisito

Consulte o programador de sistemas do MVS, o administrador de segurança e o administrador TCP/IP para verificar se os produtos e o software de requisito estão instalados, testados e funcionando.

- Na versão 7.1, o Developer for System z oferece dois métodos para execução de comandos TSO/ISPF, implicitamente pelo produto ou explicitamente pelo usuário.
 - Chamando uma função do SCLM Developer Toolkit. Esse método requer a instalação e configuração do SCLMDT, FMID HSD3310, que é fornecido com o Developer for System z. Esse método é o padrão utilizado nos membros de amostra.

As etapas de customização SCLMDT a seguir devem ser desempenhadas e estão descritas no *Guia de Instalação e Customização do SCLM Developer Toolkit (SC23-8504)*:

- Crie a estrutura de diretórios e os arquivos de configuração
- Customize ISPF.conf

Nota:

- O cliente baseado no Eclipse não é necessário para suportar essa função do SCLMDT
- O produto SCLM não é necessário para suportar essa função do SCLMDT
- Uma transação APPC (como em releases anteriores à versão 7.1). Esse método requer a instalação e configuração do APPC, que faz parte do z/OS.
- A biblioteca de classes C/C++ DLL CBC.SCLBDLL e as bibliotecas de tempo de execução LE (Language Environment), CEE.SCEERUN e CEE.SCEERUN2, devem estar em LINKLIST.
- Todos os usuários do z/OS UNIX devem possuir acesso de LEITURA e de EXECUÇÃO aos diretórios Java.
- O Developer for System z depende de INETD para configuração da conexão de host do cliente. Consulte Apêndice D, “Configurando o INETD”, na página 97 para obter mais informações sobre o REXEC e o SSH.
- Ações remotas (baseadas em host) para subprojetos do z/OS UNIX requerem que o REXEC ou o SSH esteja ativo no host.
- O Developer for System z depende de o TCP/IP ter o nome do host correto quando for inicializado. Isto significa que o TCP/IP diferente e os arquivos de configuração do Resolver devem ser configurados corretamente. Para obter

informações de customização do TCP/IP e do Resolver, consulte o Apêndice C, "Configurando o TCP/IP", na página 89 e "TCPIP.DATA configuration statements" no *Communications Server: IP Configuration Reference* (SC31-8776).

Você pode testar sua configuração TCP/IP com o comando **HOMETEST** do TSO. Consulte "TSO Commands" no *Communications Server: IP System Administrator's Commands* (SC31-8781) para obter mais informações sobre este comando.

Saída de exemplo do comando **HOMETEST**:

Executando o Testador de Configuração TCP/IP IBM MVS TCP/IP CS V1R7

O arquivo de parâmetros de configuração de FTP utilizado será "SYS1.TCPPARMS(FTPDATA)"

O Nome do Host TCP é: CDFMVS08

Utilizando o Servidor de Nomes para Resolver CDFMVS08

Os endereços IP a seguir correspondem ao Nome do Host TCP: CDFMVS08
9.42.112.75

Os endereços IP a seguir são os endereços IP HOME definidos no PROFILE.TCPIP:
9.42.112.75
127.0.0.1

Todos os endereços IP do CDFMVS08 estão na lista HOME!

O Hometest foi bem-sucedido - todos os Testes Passaram!

Considerações de ID do Usuário

O ID do usuário de um usuário do Developer for System z deve ter, pelo menos, os seguintes atributos:

- Acesso ao TSO (com um tamanho de região normal).
- Um segmento OMVS definido para o sistema de segurança (por exemplo, RACF), ambos para o ID do usuário e seu grupo padrão. Este segmento OMVS pode ser um individual ou um padrão do sistema. Consulte *Security Server RACF Security Administrator's Guide* (SA22-7683) para obter mais informações sobre o segmento OMVS padrão.
 - Um diretório inicial (com acesso de GRAVAÇÃO, LEITURA e EXECUÇÃO) alocado para o usuário e identificado no segmento de OMVS.
 - O campo PROGRAM no segmento OMVS deve ser /bin/sh ou outro shell USS z/OS UNIX válido, tal como /bin/tcsh.
 - O campo ASSIZEMAX não deve ser definido, para esses padrões do sistema serão utilizados.
 - O ID do usuário não requer UID 0.

Exemplo (comando **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
MMAPAREMA= NONE

- O grupo padrão do ID do usuário requer um GID.

Exemplo (comando **LISTGRP group NORACF OMVS**):

```
GROUP group
```

```
OMVS INFORMATION
```

```
-----
```

```
GID= 0000003243
```

- Acesso de LEITURA e EXECUÇÃO aos diretórios de instalação e configuração do Developer for System z, padrão `/usr/lpp/wd4z/*`.

Considerações do Servidor

Os serviços de host a seguir, e, portanto, suas portas, devem estar disponíveis para conexão do cliente. Todas as outras portas utilizadas pelo Developer for System z possuem tráfego apenas do host. Isso significa que, para o Developer for System z, apenas as portas listadas devem estar definidas para o firewall que está protegendo o host.

- Serviço INETD para iniciar o servidor RSE. O Developer for System z suporta várias maneiras de fazer isso:
 - Daemon RSE, porta padrão 4035
 - REXEC, porta padrão 512
 - SSH, porta padrão 22
- Servidor RSE para comunicação de host do cliente. Utiliza qualquer porta disponível por padrão, mas pode estar limitada a um intervalo especificado. A comunicação nesta porta pode ser criptografada utilizando SSL.
- Serviço Telnet TN3270 para o Emulador de Conexão do Host, porta padrão 23. A comunicação nesta porta pode ser criptografada utilizando SSL (porta padrão 992).

Nota: Clientes anteriores (versão 7.0 e mais antiga) comunicam-se diretamente com o JES Job Monitor, porta padrão 6715.

INETD é um processo do servidor z/OS UNIX necessário para configurar conexões de host do cliente do Developer for System z. As configurações ambientais do INETD, informadas ao iniciar um processo, e as permissões para o ID do usuário do INETD, devem ser definidas adequadamente para que o INETD inicie o servidor RSE.

- Se INETD for iniciado pelo JCL utilizando BPXBATCH, o tamanho da região deve ser 0.
- Se o INETD for iniciado a partir de uma sessão TSO/OMVS, o tamanho da região TSO deve ser 2096128 ou maior.
- Acesso de LEITURA e EXECUÇÃO aos diretórios de instalação do Developer for System z, padrão `/usr/lpp/wd4z/*`.
- Permissão BPX.DAEMON e possivelmente UID 0.

Consulte Apêndice D, “Configurando o INETD”, na página 97 para obter mais informações sobre permissões do INETD.

O RSE (Remote Systems Explorer) é o componente do Developer for System z que fornece os principais serviços, como de conexão do cliente ao host.

- Cada conexão de host do cliente possui um servidor RSE privado.
- Além das portas documentadas neste manual para comunicação com outros servidores do Developer for System z, o servidor RSE requer várias portas para uso interno.

Permissões Requeridas para Implementar as Tarefas de Configuração

A configuração do Developer for System z requer mais do que as permissões típicas do programador de sistemas, portanto é esperada uma necessidade mínima de assistência. A lista a seguir destaca as áreas mais importantes:

- Permissão de atualização para conjuntos de dados de sistemas (como SYS1.PARMLIB)
- Permissão para emissão de comandos do console
- Permissão para criação de diretórios do z/OS UNIX
- Permissão para atualização de diretório do z/OS UNIX /etc
- Permissão para configuração do bit de controle de programa z/OS UNIX
- Permissão para definir novas portas TCP/IP
- Permissão BPX.DAEMON e possivelmente UID 0 como (re)iniciar o INETD
- Permissões de gerenciamento do WLM/RSM
- Permissões de gerenciamento do APPC para criação de um programa de transação
- Permissões do administrador de segurança para consulta e criação de perfis de segurança
- (opcional) Permissões de administrador do DB2 para criação de um procedimento armazenado
- (opcional) Permissões de administrador do CICS para definição de programas para o CICS

Considerações de Pré-implementação

O Developer for System z e o CARMA (Common Access Repository Manager) suportam clonagem e instalação em um sistema diferente, evitando a necessidade de uma instalação SMP/E em cada sistema.

Os seguintes conjuntos de dados, diretórios e arquivos são obrigatórios para implementação em outros sistemas. Se você copiou um arquivo em um local diferente para customização, então esse arquivo deve substituir sua contraparte nas listas a seguir.

Nota: As listas a seguir não abrangem as necessidades de implementação do software obrigatório e de co-requisito.

IBM Rational Developer para System z, FMID HHOP710

- hlq.SFEKLOAD(*)
- hlq.SFEKPROC(*)
- hlq.SFEKSAMP(FEJJCNFG)
- hlq.SFEKSAMP(FEJJJCL)
- hlq.SFEKSAMP(ELAXF*)
- /usr/lpp/wd4z/*
- optional parts
 - hlq.SFEKSAMP(FEJTSO) - envio TSO
 - hlq.SFEKSAMP(ELAXM*) - procedimento armazenado do DB2
 - Definições CICS criadas pelo hlq.SFEKSAMP(ADNPCCSD) - Conexão primária ADM

- Definições CICS criadas pelo hlq.SFEKSAMP(ADNARCSD) - Conexão não-primária ADM
- VSAM criado pelo hlq.SFEKSAMP(ADNVSAM) - Servidor CRD ADM
- Módulo de carregamento criado pelo hlq.SFEKSAMP(ADNCMSGH) - Manipulador do Pipeline ADM
- Transação APPC criada pelo hlq.SFEKSAMP(FEKAPPCC) - Serviço Comandos do TSO baseado no APPC
- /etc/SCLMDT/CONFIG/ISPF.conf - inclui o Servidor de Comandos do TSO no ambiente ISPF do SCLM Developer Toolkit

Nota: hlq e /usr/lpp/wd4z são os qualificadores de alto nível (padrão FEK) e caminho utilizados durante a instalação do produto.

IBM CARMA, FMID HCMA710

- hlq.SCRALOAD(*)
- hlq.SCRACLIST(CRASUBMT)
- VSAMs criados pelos seguintes membros hlq.SCRASAMP
 - VSAM de mensagem hlq.SCRASAMP(CRA\$VMSG)
 - VSAM de configuração hlq.SCRASAMP(CRA\$VDEF)
 - VSAM de informações customizadas hlq.SCRASAMP(CRA\$SVSTR)

Nota: hlq é o qualificador de alto nível (padrão CRA) utilizado durante a instalação do produto.

Capítulo 2. Alterações na Instalação e na Configuração

Esta seção destaca as alterações na instalação e na configuração de releases anteriores do produto.

Alterações entre a Versão 7.0 e a Versão 7.1

IBM Rational Developer para System z, FMID HHOP710

- **Incluído:** Opção de configuração - comandos do TSO/ISPF via uma transação APPC ou via SCLM Developer Toolkit
- **Incluído:** Os seguintes membros customizáveis são novos
 - samplib ELAXFADT
 - samplib ADNCMSGH
 - /usr/lpp/wd4z/rse/lib/FMIEXT.properties
- **Alterado:** Os seguintes membros foram movidos
 - SFEKDLL(FEJBDTRX) -> SFEKLOAD(FEJBDTRX)
- **Alterado:** Os seguintes membros customizáveis foram alterados
 - samplib FEKFAPPCC
 - /usr/lpp/wd4z/rse/lib/rsed.envvars
 - /usr/lpp/wd4z/rse/lib/setup.env.zseries
 - /usr/lpp/wd4z/rse/lib/server.zseries

IBM CARMA, FMID HCMA710

- **Alterado:** A criação de log foi gravada na instrução CARMALOG DD
- **Incluído:** Os seguintes membros customizáveis são novos
 - samplib CRA#ECOB
 - samplib CRA#EPDS
 - samplib CRA#ERAM
 - samplib CRA#ESLM
- **Renomeado:** Os seguintes membros customizáveis foram renomeados
 - samplib CRAREPR -> CRA\$VDEF
 - samplib CRAMREPR -> CRA\$VMMSG
 - samplib CRASREPR -> CRA\$VSTR
 - samplib CRASALX -> CRA#ASLM
 - samplib CRACOBJ1 -> CRA#CCB1
 - samplib CRACOBJ2 -> CRA#CCB2
 - samplib CRACLICM -> CRA#CCLT
 - samplib CRARAMCS -> CRA#CPDS
 - samplib CRARAMCM -> CRA#CRAM
 - samplib CRATREPR -> CRA#VPDS
 - samplib CRALREPR -> CRA#VSLM
 - samplib CRACLIRN -> CRA#XCLT
- **Alterado:** Os seguintes membros customizáveis foram alterados

Alterações entre a Versão 6.0.1 e a Versão 7.0

IBM WebSphere Developer for System z, FMID HHOP700

- **Alterado:** O IBM WebSphere Developer for System z, FMID HHOP700, combina agora as funções de 5 produtos separados em 1 para reduzir o trabalho da instalação
 - v601 FMID H001600, IBM WebSphere Developer for zSeries RSE + ICU
 - v601 FMID H002600, IBM WebSphere Developer for zSeries JES Job Monitor
 - v601 FMID HEDS500, IBM WebSphere Developer for zSeries Opções para z/OS (nem todas as funções estão incluídas no HHOP700)
 - v601 FMID HBDI601, IBM WebSphere Developer for zSeries suporte ao bidi
 - (novo) versão 700, IBM Application Deployment Manager
- **Alterado:** As tarefas de instalação são renomeadas para que as etapas a serem executadas estejam em ordem alfabética
- **Alterado:** A instalação do SMP/E agora configura os bits de controle de programa em z/OS UNIX
- **Alterado:** A codificação do servidor REXX é movida de samplib para um novo conjunto de dados, SFEKPROC
- **Alterado:** O bidi utiliza um novo módulo de carregamento, todas as referências aos antigos devem ser removidas
- **Alterado:** O parâmetro JESNAME não é mais utilizado no arquivo de configuração do JES Job Monitor (samplib FEJJCNFG)
- **Alterado:** Os seguintes membros não precisam mais de customização

/usr/lpp/wd4z/rse/lib/setup.env.zseries

- **Incluído:** Os seguintes membros são novos
 - samplib ELAXFCPC
 - samplib ELAXFCPP
 - samplib ELAXFMFS
 - /usr/lpp/wd4z/rse/lib/projectcfg.properties
- **Renomeado:** Os seguintes membros customizáveis foram renomeados
 - samplib FEKFRDIS -> FEKAPPCL
 - samplib FEKFRDEL -> FEKAPPCX
 - samplib FEKFRTAD -> FEKAPPCC
- **Removido:** Os seguintes membros não são mais enviados
 - samplib ELAXFLNK
 - samplib FEJIVJCL
 - loadlib FEJBDTRN
 - loadlib FEJBDTRE

IBM CARMA, FMID HCMA700

- **Alterado:** As tarefas de instalação são renomeadas para que as etapas a serem executadas estejam em ordem alfabética
- **Alterado:** Os seguintes membros não precisam mais de customização
 - /usr/lpp/wd4z/rse/lib/rsed.envvars

Fazendo Backup de Arquivos Configurados Anteriormente

Antes de instalar o Developer for System z versão 7.1, se você for um usuário anterior do Developer for System z, é recomendável salvar os arquivos customizados relacionados. Leia o Apêndice A, “Executando Várias Instâncias do Developer for System z”, na página 71 antes de iniciar a customização se você planeja executar várias instâncias do Developer for System z.

A Tabela 2 e a Tabela 3 na página 13 fornecem uma visão geral dos arquivos que podem ter sido customizados para o Developer for System z e CARMA versão 6.0.1 e posterior. A Tabela 4 na página 14 lista as customizações de conjuntos de dados e produtos e software obrigatórios e de co-requisito que ocorrem durante a instalação de um Developer for System z e CARMA versão 6.0.1 (e posterior).

Tabela 2. Visão geral de membros do MVS customizado

Membro	Local padrão v6.0.1	Local padrão v7.0/v7.1	Objetivo
FEJJCNFG	hlq.SFEJSAMP (hlq = FEJ)	hlq.SFEKSAMP (hlq = FEK)	Arquivo de configuração para JES Job Monitor
FEJJJCL	hlq.SFEJSAMP (hlq = FEJ)	hlq.SFEKSAMP (hlq = FEK)	JCL para JES Job Monitor
FEJTSO	hlq.SFEJSAMP (hlq = FEJ)	hlq.SFEKSAMP (hlq = FEK)	JCL para emissões TSO
FEKAPPCC	não existe	hlq.SFEKSAMP (hlq = FEK)	JCL para criação de uma transação APPC
FEKAPPCL	não existe	hlq.SFEKSAMP (hlq = FEK)	JCL para exibição de uma transação APPC
FEKAPPCX	não existe	hlq.SFEKSAMP (hlq = FEK)	JCL para exclusão de uma transação APPC
FEKFRTAD	hlq.SFEKSAMP (hlq = FEK)	(novo nome de membro FEKAPPCC)	JCL para criação de uma transação APPC
FEKFRDIS	hlq.SFEKSAMP (hlq = FEK)	(novo nome de membro FEKAPPCL)	JCL para exibição de uma transação APPC
FEKFRDEL	hlq.SFEKSAMP (hlq = FEK)	(novo nome de membro FEKAPPLX)	JCL para exclusão de uma transação APPC
ELAXF*	hlq.SCCUSAMP (hlq = CCU)	hlq.SFEKSAMP (hlq = FEK)	JCL para construções de projetos remotos, etc.

Tabela 2. Visão geral de membros do MVS customizado (continuação)

Membro	Local padrão v6.0.1	Local padrão v7.0/v7.1	Objetivo
ELAXMSAM	hlq.SCCUSAMP (hlq = CCU)	hlq.SFEKSAMP (hlq = FEK)	Procedimento JCL do espaço de endereços WLM para o Stored Procedure Builder para PL/I e COBOL
ELAXMJCL	hlq.SCCUSAMP (hlq = CCU)	hlq.SFEKSAMP (hlq = FEK)	JCL para definição do Stored Procedure Builder para PL/I e COBOL para o DB2
ADNVSAM	não existe	hlq.SFEKSAMP (hlq = FEK)	JCL para definir o repositório ADM CRD
ADNPCCSD	não existe	hlq.SFEKSAMP (hlq = FEK)	JCL para ativação do servidor CRD na região CICS primária
ADNCMSGH	não existe	hlq.SFEKSAMP (hlq = FEK) (não existe na versão 7.0)	JCL para compilar o Manipulador de Mensagem de Pipeline
ADNTMSGH	não existe	hlq.SFEKSAMP (hlq = FEK)	Código-fonte de amostra para o Manipulador de Mensagem de Pipeline
ADNARCSD	não existe	hlq.SFEKSAMP (hlq = FEK)	JCL para ativação do servidor CRD em regiões CICS não-primárias
CRASUBMT	hlq.SCRACLST (hlq = CRA)	hlq.SCRACLST (hlq = CRA)	CLIST para emissão do JCL para um servidor CARMA
CRAMREPR	hlq.SCRASAMP (hlq = CRA)	hlq.SCRASAMP (hlq = CRA) (novo nome de membro na versão 7.1 CRA\$VMSG)	JCL para criação do VSAM de mensagens do CARMA

Tabela 2. Visão geral de membros do MVS customizado (continuação)

Membro	Local padrão v6.0.1	Local padrão v7.0/v7.1	Objetivo
CRAREPR	hlq.SCRASAM (hlq = CRA)	hlq.SCRASAMP (hlq = CRA) (novo nome de membro na versão 7.1 CRA\$VDEF)	JCL para criação do VSAM de configuração do CARMA
CRASREPR	hlq.SCRASAM (hlq = CRA)	hlq.SCRASAMP (hlq = CRA) (novo nome de membro na versão 7.1 CRA\$VSTR)	JCL para criação do VSAM de informações customizadas do CARMA
CRALREPR	hlq.SCRASAM (hlq = CRA)	hlq.SCRASAMP (hlq = CRA) (novo nome de membro na versão 7.1 CRA#VSLM)	JCL para criação do VSAM de mensagens de RAM do SCLM
CRASALX	hlq.SCRASAM (hlq = CRA)	hlq.SCRASAMP (hlq = CRA) (novo nome de membro na versão 7.1 CRA#ASLM)	JCL para criação dos conjuntos de dados de RAM do SCLM
CRATREPR	hlq.SCRASAM (hlq = CRA)	hlq.SCRASAMP (hlq = CRA) (novo nome de membro na versão 7.1 CRA#VPDS)	JCL para criação do VSAM de mensagens de RAM do PDS

Nota: O hlq é o qualificador de alto nível utilizado durante a instalação do produto. Os padrões IBM fornecidos para o hlq estão listados, mas talvez não se apliquem ao seu site.

Tabela 3. Visão geral dos arquivos z/OS UNIX customizados

Arquivo	Local padrão v6.0.1	Local padrão versão 7.0/versão 7.1	Objetivo
rsed.envvars	/usr/lpp/wd4z/rse/lib/	/usr/lpp/wd4z/rse/lib/	Arquivo de configuração RSE
rsecomm.properties	/usr/lpp/wd4z/rse/lib/	/usr/lpp/wd4z/rse/lib/	Arquivo de configuração de rastreamento RSE
ssl.properties	/usr/lpp/wd4z/rse/lib/	/usr/lpp/wd4z/rse/lib/	Arquivo de configuração SSL RSE
setup.env.zseries	/usr/lpp/wd4z/rse/lib/	(não mais customizado)	Exportar variáveis de ambiente RSE
projectcfg.properties	(não existe)	/usr/lpp/wd4z/rse/lib/	Arquivo de configuração de projetos do host
FMIEXT.properties	(não existe)	/usr/lpp/wd4z/rse/lib/ (não existe na versão 7.0)	Arquivo de configuração do File Manager
CRASRV.properties	/usr/lpp/wd4z/rse/lib/	/usr/lpp/wd4z/rse/lib/	Arquivo de configuração CARMA

Tabela 3. Visão geral dos arquivos z/OS UNIX customizados (continuação)

Arquivo	Local padrão v6.0.1	Local padrão versão 7.0/versão 7.1	Objetivo
rexsub	/usr/lpp/wd4z/rse/lib/	(não mais customizado)	REXX do z/OS UNIX do CARMA para execução do MVS CRASUBMT CLIST

Nota: /usr/lpp/wd4z é o caminho utilizado durante a instalação dos produtos. O padrão IBM fornecido é mostrado, mas talvez não se aplique ao seu site.

Tabela 4. Customização em bibliotecas não-Developer for System z

Membro/ Arquivo	Local padrão	Customização necessária
BPXPRMxx	SYS1.PARMLIB	Configure MAXASIZE como 2147483647 ou maior
PROGxx	SYS1.PARMLIB	hlq.SFEKLOAD autorizado por APF
ASCHPMxx	SYS1.PARMLIB	Definir uma classe de transação APPC para o serviço de comandos do TSO
services	/etc/	Definir daemon RSE
inetd.conf	/etc/	Definir daemon RSE
ISPF.conf	/etc/SCLMDT/ CONFIG/	Defina o local do Servidor de Comandos do TSO
/	APPC	Definir uma transação APPC para o serviço de comandos do TSO
/	WLM	Associar o programa de transação APPC a um grupo de desempenho do TSO
/	WLM	Designar um ambiente de aplicativos para o procedimento armazenado do DB2

Capítulo 3. Ativando os Componentes MVS do Developer for System z

Antes de instalar a versão 7.1, se você for um usuário anterior do WebSphere Developer for System z (FMID: HHOP700), é recomendável salvar a customização relacionada, conforme descrito em “Fazendo Backup de Arquivos Configurados Anteriormente” na página 11.

Todas as referências a hlq neste capítulo se referem ao qualificador de alto nível utilizado durante a instalação do Developer for System z. O padrão da instalação é FEK, mas isto talvez não se aplique ao seu site.

Nota: A biblioteca de classes C/C++ DLL CBC.SCLBDLL e as bibliotecas de tempo de execução LE (Language Environment) CEE.SCEERUN e CEE.SCEERUN2 devem estar em LINKLIST.

O requisito de LINKLIST pode ser ignorado incluindo-se uma instrução STEPLIB em rsed.envvars; consulte “Customizar rsed.envvars, o Arquivo de Configuração para RSE” na página 35.

Configurar MAXASSIZE em SYS1.PARMLIB(BPXPRMxx)

MAXASSIZE define o tamanho da região do espaço (processo) de endereços. Configure MAXASSIZE em SYS1.PARMLIB(BPXPRMxx) como 2147483647 ou maior.

Este valor pode ser verificado e configurado dinamicamente (até o próximo IPL) com os seguintes comandos do console, conforme descrito em *MVS System Commands* (GC28-1781).

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

Consulte “Tamanho do Espaço de Endereço” na página 82 para obter mais informações sobre outros locais em que tamanhos de espaço do endereço possam ser definidos ou limitados.

hlq.SFEKLOAD Autorizada por APF

Para que o JES Job Monitor acesse os arquivos do spool do JES, a biblioteca de carregamento hlq.SFEKLOAD deve ser autorizada por APF.

As autorizações APF são definidas em SYS1.PARMLIB(PROGxx), se seu site seguiu as recomendações da IBM. Consulte *MVS Initialization and Tuning Reference* (SC28-1752) para obter informações adicionais sobre a definição de autorizações APF.

Para finalidades de testes, as autorizações APF também podem ser fornecidas dinamicamente. Estas autorizações durarão até o próximo IPL do sistema. O comando do console necessário se parecerá com este:

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

Consulte *MVS System Commands (GC28-1781)* para obter informações adicionais sobre o comando **SETPROG**.

Customizar o FEJJCNFG, o Arquivo de Configuração do JES Job Monitor

O JES Job Monitor é o componente do Developer for System z que administra toda a conectividade do JES. Um arquivo de configuração é utilizado para customizar determinados aspectos para atender os padrões do seu site.

Nota: É recomendável copiar o arquivo de configuração de amostra para um novo conjunto de dados e customizar essa cópia para evitar sobrescrevê-la ao realizar manutenção.

O arquivo de configuração sample está localizado em:
hlq.SFEKSAMP(FEJJCNFG)

O arquivo consiste de um conjunto de definições de variáveis de ambiente. Linhas de comentários iniciam com o sinal de libras (#). O arquivo de configuração de amostra está listado em Figura 1.

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

Figura 1. FEJJCNFG, arquivo de configuração do JES Job Monitor

As seguintes definições são requeridas:

SERV_PORT

O número da porta para o servidor de host do JES Job Monitor. A porta padrão é 6715. Altere conforme desejado, no entanto, o cliente e o servidor devem ser configurados com o mesmo número de porta. Se você alterar o número da porta do servidor, todos os usuários do cliente do Developer for System z também devem alterar a porta do JES Job Monitor para este sistema na Visualização de Sistemas Remotos.

Nota: Antes de selecionar uma porta, verifique se ela está disponível em seu sistema com os comandos do TSO **NETSTAT** e **NETSTAT PORTL**. Consulte “Portas TCP/IP Reservadas” na página 81 para obter informações adicionais.

Nota: Ao utilizar um cliente da versão 7.1 ou posterior, toda a comunicação nesta porta é confinada para a máquina host.

CODEPAGE

A página de códigos da estação de trabalho. O padrão é UTF-8. A página de códigos da estação de trabalho é definida como UTF-8 e geralmente não deve ser alterada. Poderá ser necessário alterar o UTF-8 para corresponder

à página de códigos da estação de trabalho, se você tiver dificuldades com os caracteres NLS, como o símbolo da moeda.

HOST_CODEPAGE

A página de códigos do host. O padrão é IBM-1047. Altere conforme necessário.

TZ Seleção de fuso horário. O padrão é EST5EDT. O fuso horário padrão é UTC +5 horas (horário de verão do horário padrão na costa leste dos EUA). Altere isso para representar seu fuso horário. Informações adicionais podem ser encontradas em *UNIX System Services File System Interface Reference (SA22-7808)*.

LISTEN_QUEUE_LENGTH

O comprimento da fila de atendimento do TCP/IP. O padrão é 5. Não altere esse valor, a menos que orientado pelo IBM Support Center.

MAX_DATASETS

O padrão é 32. Este é o número máximo de conjuntos de dados de saída em spool que o JES Job Monitor retornará ao cliente (por exemplo, SYSOUT, SYSPRINT, SYS00001, etc.).

MAX_THREADS

O padrão é 200. Número máximo de usuários que podem utilizar um Monitor de Tarefas do JES por vez. Aumentar este número pode exigir o aumento do tamanho do espaço de endereços do JES Job Monitor.

TIMEOUT_INTERVAL

O padrão é 1200. Controla a frequência com que o servidor verifica e elimina encadeamentos ao atingir o tempo limite (consulte **TIMEOUT**). O valor de **TIMEOUT_INTERVAL** especifica o número de segundos entre as verificações.

TIMEOUT

O padrão é 3600. O período de tempo, em segundos, antes que um encadeamento seja eliminado devido à falta de interação com o cliente. O valor máximo é 2147483647.

AUTHMETHOD

O padrão é RACF, o que significa que a interface de segurança padrão é utilizada. Isto não deve ser alterado, mesmo se você utilizar um produto diferente de RACF.

As definições a seguir são opcionais. Se omitidas, valores padrão serão utilizados conforme especificados abaixo:

LIMIT_VIEW=USERID

Define qual saída o usuário pode visualizar. O padrão (**LIMIT_VIEW=NOLIMIT**) permite que o usuário visualize toda a saída do JES. Especificar **USERID** limita a visualização na saída possuída pelo usuário.

Nota: As únicas configurações válidas são **USERID** e **NOLIMIT**.

CONCHAR

Especifica o caractere de comando do console do JES. **CONCHAR** tem **CONCHAR=\$** como padrão para JES2, ou **CONCHAR=*** para JES3. Se sua instalação definiu um caractere de comando diferente, especifique-o como o valor para **CONCHAR**.

SUBMITMETHOD=TSO

Enviar por meio do TSO. O padrão (**SUBMITMETHOD=JES**) envia jobs

diretamente para o JES. Especificar `SUBMITMETHOD=TSO` faz com que o job seja enviado por meio do comando **SUBMIT** do TSO. Este método permite que o TSO seja encerrado para ser chamado; entretanto, ele possui uma desvantagem de desempenho e por este motivo não é recomendado.

Nota: As únicas configurações válidas são TSO e JES.

Nota: Se `SUBMITMETHOD=TSO` for especificado, então `TSO_TEMPLATE` também deve ser definido.

TSO_TEMPLATE=hlq.SFEKSAMP(FEJTSO)

JCL wrapper para envio do job por meio do TSO. Não há valor padrão. Esta instrução faz referência ao nome completo do membro do JCL a ser utilizado como um wrapper para o envio TSO. Consulte a instrução `SUBMITMETHOD` para obter informações adicionais.

Nota: Um job do wrapper de amostra é fornecido em `hlq.SFEKSAMP(FEJTSO)`. Consulte este membro para obter informações adicionais sobre a customização necessária.

Nota: `TSO_TEMPLATE` não possui nenhum efeito a menos que `SUBMITMETHOD=TSO` também seja especificado.

Nota: A definição `JESNAME` não é mais necessária. Desde a versão 7.0, o JES Job Monitor detecta automaticamente se o JES primário é JES2 ou JES3.

Customizar o JCL de Inicialização do JES Job Monitor

O JES Job Monitor é o componente do Developer for System z que administra toda a conectividade do JES. Para fazer isto, um servidor deve estar ativo no sistema (como job do usuário ou STC). Siga as etapas de customização do JCL localizadas em `hlq.SFEKSAMP(FEJJJCL)` para criar o servidor JES Job Monitor de acordo com os padrões do seu site.

Nota: É recomendável copiar o JCL de amostra para um novo conjunto de dados e customizar a cópia para evitar sobrescrevê-la ao realizar manutenção.

Rastreio do JES Job Monitor

Se for necessário ativar o rastreio do JES Job Monitor para diagnóstico, inclua `"-TV"` na linha `PARM`. O rastreio causará diminuição no desempenho e deverá ser realizado somente sob a orientação do IBM Support Center.

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

O rastreio também poderá ser controlado por comandos do console. Assumindo que `JMON` é o nome de tarefa utilizado, o primeiro comando de console listado a seguir ativará o rastreio e o segundo o desativará.

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

Executar o JES Job Monitor como uma STC

O JES Job Monitor pode ser executado como um job de usuário ou como tarefa iniciada (STC). As tarefas a seguir precisam ser implementadas para definir `JMON` como uma STC:

1. O JCL não precisa ter uma placa JOB (preferencialmente não). A maioria das STCs inicia com uma placa PROC, como no exemplo em Figura 2.

```
//JMON      PROC HLQ=FEK,
//          PRM=
//*
//* WD/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
```

Figura 2. JCL do JES Job Monitor

2. O JCL deve residir em uma biblioteca de procedimentos do sistema (SYS1.PROCLIB é a padrão IBM).
Para uma fácil referência, é recomendável que o nome do membro corresponda ao nome do procedimento (JMON na amostra acima).
3. É recomendável que as STCs tenham um ID de usuário dedicado. Por questão de segurança, esse ID do usuário deve ser 'protegido', definindo a palavra-chave NOPASSWORD (em RACF). Isso significa que o RACF falhará em qualquer tentativa de logon que exija uma senha, mas sem revogar o ID do usuário.

Utilize o seguinte comando de amostra para criar tal ID do usuário, em que userid é o ID do usuário em questão, groupid é o grupo padrão e uid é o ID do UNIX.

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

4. As STCs devem ser definidas no software de segurança (ou seja, RACF). Há diferentes formas de definição de uma STC, mas a utilização da classe STARTED é recomendada. Para definir a classe STARTED, o administrador de segurança emitiria os seguintes comandos RACF;

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

Para incluir o JES Job Monitor como uma STC, a seguir você possui os comandos RACF necessários, em que jmon é o nome do membro JCL e userid é o ID de usuário cujas as autoridades devem ser utilizadas.

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

Nota: Incluindo a palavra-chave TRUSTED(YES) no campo STDATA [STDATA(USER(userid) TRUSTED(YES)], é possível evitar a definição de permissões individuais necessárias para o ID do usuário STC. O RACF ignora verificações de segurança de conjunto de dados para STCs confiáveis. No entanto, antes de fazer isso, garanta que esse ID do usuário não seja usado de forma inadequada, protegendo-o conforme descrito acima.

Consulte *Security Server RACF Security Administrator's Guide (SA22-7683)* para obter mais informações sobre tarefas iniciadas e segurança.

- Assim que as tarefas anteriores forem concluídas, o JES Job Monitor poderá ser iniciado e parado como uma STC.

O operador do sistema pode emitir os seguintes comandos no console (em que JMON é o nome da STC do JES Job Monitor):

- Start JMON
- stoP JMON
- Display A,JMON

Se você possuir a autoridade adequada, você poderá emitir estes comandos a partir do SDSF, entretanto, os comandos deverão ser precedidos por uma barra ('/'). Consulte *MVS System Commands (SA22-7627)* para obter informações adicionais sobre os comandos do console.

Permissões do Servidor

O ID do usuário designado ao servidor do JES Job Monitor precisa de acesso de LEITURA à biblioteca de carregamento do Developer for System z: hlq.SFEKLOAD.

Verificação do JCL de inicialização do JES Job Monitor

Inicie o job ou a STC do usuário definida nas etapas acima. Se a tarefa terminar com o código de retorno 66, então hlq.SFEKLOAD não será autorizado por APF.

Acesso e Segurança do Spool do JES

Acesso Condicional ao Spool

Para permitir que os usuários executem operações por meio do JES Job Monitor, é necessário conceder a eles autoridade de acesso à classe OPERCMDS. Isto pode ser realizado condicionalmente, para que os direitos de acesso dos usuários tenham efeito somente quando eles estiverem utilizando o JES Job Monitor. Para utilizar esta verificação de acesso condicional, você deve ter a classe CONSOLE ativa e um console definido chamado JMON (JMON é o único nome válido).

Por exemplo, o administrador de segurança emitiria os seguintes comandos RACF:

- SETOPTS CLASSACT(CONSOLE)
- RDEFINE CONSOLE JMON UACC(READ)
- SETOPTS RACLIST(CONSOLE) REFRESH

Utilize os seguintes comandos RACF para permitir que os usuários emitam comandos do JES2 somente por meio do JES Job Monitor, em que id é o ID do usuário ou ID do grupo de usuários do Developer for System z:

- RDEFINE OPERCMDS JES2.** UACC(NONE)
- PERMIT JES2.** CLASS(OPERCMDS) ID(id) ACCESS(CONTROL) WHEN(CONSOLE(JMON))
- SETOPTS RACLIST(OPERCMDS) REFRESH
- SETOPTS GENERIC(OPERCMDS) REFRESH

CUIDADO:

A definição de comandos JES com acesso universal NONE em seu software de segurança pode impactar outros aplicativos e operações. Teste isto antes de ativá-lo em um sistema de produção.

Comandos Disponíveis

O JES Job Monitor não oferece aos usuários do Developer for System z acesso total ao spool do JES. Apenas os comandos listados em Tabela 5 estão disponíveis. Os comandos são emitidos selecionando a opção apropriado na estrutura de menus do cliente (sem prompt de comandos). O escopo dos comandos é limitado pelas técnicas descritas a seguir.

Tabela 5. Comandos do Console do JES Job Monitor

Comando	JES2	JES3
Suspender	\$Hjobid	*F,J=jobid,H
Liberação	\$Ajobid	*F,J=jobid,R
Cancelar	\$Cjobid	*F,J=jobid,C
Limpar	\$Cjobid,P	*F,J=jobid,C

Sem autorização para esses comandos do console, os usuários ainda poderão enviar tarefas e ler a saída da tarefa, se tiverem recebido autorização de acesso aos arquivos do spool.

Limitando o Acesso aos Arquivos do Spool

Para limitar os usuários aos seus próprios jobs no spool do JES, defina a instrução "LIMIT_VIEW=USERID" no arquivo de configuração JES Job Monitor (FEJJCNFG). Se precisarem de acesso a uma maior quantidade de jobs, mas não todos, utilize os recursos padrão de proteção de arquivos do spool do produto de segurança, como a classe JESSPOOL no RACF.

Ao definir a proteção adicional, tenha em mente que o JES Job Monitor utiliza a SAPI (SYSOUT application program interface) para acessar o spool. Isto significa que o usuário precisa, pelo menos, do acesso UPDATE aos arquivos do spool, mesmo para a funcionalidade de procura. Este requisito não é aplicado se você executa o z/OS v1.7 (z/OS 1.8 para JES3) ou posterior. Aqui, a permissão de LEITURA é suficiente para a funcionalidade de procura.

Consulte *Security Server RACF Security Administrator's Guide (SA22-7683)* para obter informações adicionais sobre a proteção de arquivos do spool do JES.

Customizar ELAXF*, Procedimentos de Construção Remota

O Developer for System z fornece procedimentos JCL de amostra que podem ser utilizados para a geração do JCL, construções remotas de projetos e recursos de verificação de sintaxe remota dos mapas BMS do CICS, telas IMS MFS e programas COBOL, PL/I, Assembler, C e C/C++.

Estes procedimentos permitem que as instalações apliquem seus próprios padrões. Isto também assegurará que os desenvolvedores utilizem os mesmos procedimentos com as mesmas opções e níveis do compilador.

O SMP/E instala estes procedimentos JCL de amostra no hlq.SFEKSAMP. Se planeja utilizar esses procedimentos, você deve:

1. Copiar todos os procedimentos chamados ELAXF* para uma biblioteca de procedimentos de sistema (como SYS1.PROCLIB) disponível aos usuários.
2. Os procedimentos copiados devem ser customizados para refletir as convenções de nomenclatura utilizadas no sistema de destino. A customização requerida é documentada em cada procedimento do JCL.

Os procedimentos de amostra a serem copiados e customizados estão listados na Tabela 6.

Se os procedimentos ELAXF* não puderem ser copiados para uma biblioteca de procedimentos de sistema, copie-os para uma biblioteca privada e solicite aos usuários do Developer for System z a inclusão de uma placa JCLLIB (logo após a placa JOB) nas propriedades da tarefa no cliente. Não customize o JCL de amostra no conjunto de dados de instalação, uma vez que a manutenção pode substituir esses membros e desfazer as customizações.

```
//MYJOB    JOB <parâmetros_da_tarefa>
//PROCS JCLLIB ORDER=(hlq.SFEKSAMP)
```

Nota: A Geração de JCL/Construções de projetos remotos e as Operações de Verificação de Sintaxe Remota realizadas a partir de um cliente do Developer for System z assumem que esses procedimentos estão customizados e disponíveis ao usuário.

Os nomes dos procedimentos e os nomes das etapas nos procedimentos correspondem às propriedades padrão que são enviadas com o cliente. Se você decidir alterar o nome do procedimento ou o nome de uma etapa em um procedimento, o arquivo de propriedades correspondente no cliente também deverá ser atualizado. É recomendável que você não altere os nomes dos procedimentos e das etapas.

Nota: O IBM Debug Tool deverá ser solicitado, instalado e configurado para suportar depuração remota de programas assembler, COBOL e PL/I. Consulte o *IBM Rational Developer para System z Host Planning Guide (GI11-8296-00)* para saber qual nível do Debug Tool é necessário para sua versão do Developer for System z. A instalação e customização deste produto não está descrita neste manual.

Tabela 6. Procedimento ELAXF* de amostra

Membro	Objetivo
ELAXFADT	Procedimento de amostra para montagem e depuração de programas assembler de Alto Nível.
ELAXFASM	Procedimento de amostra para montagem de programas assembler de alto nível.
ELAXFBMS	Procedimento de amostra para criação do objeto BMS do CICS e a cópia correspondente, dsect, ou incluir membro.
ELAXFCOC	Procedimento de amostra para realizar compilações COBOL, conversão do CICS integrado e conversão do DB2 integrado.
ELAXFCOP	Procedimento de amostra para realizar o pré-processamento DB2 das instruções EXEC de SQL incorporadas em programas COBOL.
ELAXFCOT	Procedimento de amostra para realizar a conversão do CICS de instruções EXEC CICS incorporadas nos programas COBOL.
ELAXFCPC	Procedimento de amostra para realizar compilações C.
ELAXFCPP	Procedimento de amostra para realizar compilações C++.

Tabela 6. Procedimento ELAXF* de amostra (continuação)

Membro	Objetivo
ELAXFGO	Procedimento de amostra para a etapa IR.
ELAXFLNK	Procedimento de amostra para vincular os programas assembler C/C++, COBOL, PLI e de Alto Nível.
ELAXFMFS	Procedimento de amostra para criar telas IMS MFS.
ELAXFPLP	Procedimento de amostra para realizar o pré-processamento DB2 de instruções EXEC de SQL incorporadas nos programas PLI.
ELAXFPLT	Procedimento de amostra para realizar a conversão do CICS das instruções EXEC CICS incorporadas nos programas PLI.
ELAXFPL1	Procedimento de amostra para realizar compilações PL/I, conversão do CICS integrado e conversão do DB2 integrado.
ELAXFUOP	Procedimento de amostra para a geração da etapa UOPT ao construir programas que executam nos subsistemas CICS ou IMS.

(Opcional) Definir uma Transação APPC para o Serviço de Comandos do TSO

A definição da transação APPC tornou-se opcional na versão 7.1. Por padrão, o Developer for System z agora utiliza o SCLM Developer Toolkit para fornecer o serviço Comandos do TSO. A configuração disso é feita em `rsed.envvars`, descrita em “Customizar `rsed.envvars`, o Arquivo de Configuração para RSE” na página 35.

Nota: O JCL do Programa de Transações, utilizado pelo APPC para iniciar o serviço Comandos do TSO, foi alterado na versão 7.1. Se você definiu anteriormente o serviço Comandos do TSO para capturar a saída ISPEXEC, deve definir uma nova transação APPC ou incluir a palavra-chave NESTMACS na linha PARM, por exemplo:

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

O serviço Comandos do TSO é implementado como um programa de transações APPC, FEKFRSRV e deve ser ativado para que as conexões do Developer for System z sejam bem-sucedidas entre o host e o cliente. O FEKFRSRV atua como um servidor host para a execução de comandos do TSO que são emitidos a partir da estação de trabalho via TCP/IP. A APPC não é requerida na estação de trabalho porque a estação de trabalho se comunica com o FEKFRSRV via TCP/IP. Cada estação de trabalho pode ter uma conexão ativa para vários hosts ao mesmo tempo.

Nota: Se você não estiver familiarizado com o APPC, leia Apêndice F, “Configurando o APPC”, na página 117 antes de continuar com esta seção.

Preparação

- As tarefas a seguir são um pré-requisito e devem ser concluídas antes da configuração do Servidor de Comandos do TSO. Os manuais mencionados descrevem estas tarefas.
 1. Instale, configure e inicie o VTAM no sistema z/OS. Consulte o *Communications Server bookshelf (F1A1BK61 para z/OS 1.7)* para obter mais informações.
 2. Instale, configure e inicie o TCP/IP no sistema z/OS. Consulte o Apêndice C, “Configurando o TCP/IP”, na página 89 e o *Communications Server bookshelf (F1A1BK61 para z/OS 1.7)* para obter informações adicionais.

3. Configure e inicie a APPC e o subsistema ASCH (planejador de transações APPC). Consulte o Apêndice F, “Configurando o APPC”, na página 117 e o *MVS bookshelf (IEA2BK60 para z/OS 1.7)* para obter informações adicionais.
- O REXX em Figura 3 pode ser utilizado para gerenciar o APPC por meio de painéis ISPF.

```
/* REXX -- administração APPC utilizando painéis ISPF */
address ISPEXEC
"LIBDEF ISPMLIB DATASET ID('ICQ.ICQMLIB') STACK"
"LIBDEF ISPLIB 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 ISPLIB"
"LIBDEF ISPSLIB"
"LIBDEF ISPTLIB"
exit
```

Figura 3. Painéis do REXX para o APPC ISPF

Nota: Esteja ciente de que você pode desativar a transação APPC com esta ferramenta; a transação ainda está lá, mas não aceitará nenhuma conexão.

- A definição da transação APPC requer habilidades em diversas áreas do sistema operacional MVS. Consulte administradores experientes utilizando a seguinte lista de verificação antes de continuar.

Tabela 7. Lista de verificação de transações APPC

	Informações necessárias:	
Experiência	<ul style="list-style-type: none"> • Valor Padrão • Onde encontrar a resposta 	Valor
APPC	Nome do conjunto de dados de TPDATA <ul style="list-style-type: none"> • Padrão: SYS1.APPCTP • O valor está listado em SYS1.PARMLIB(APPCPMxx) 	
APPC	Nome da transação a ser utilizada (talvez não exista) <ul style="list-style-type: none"> • Padrão: FEKFRSRV • Transações existentes podem ser consultadas selecionando-se "Administração de Perfis TP" no menu ISPF APPC 	
APPC	Classe de transação APPC a ser utilizada <ul style="list-style-type: none"> • Padrão: A • As classes APPC são definidas em SYS1.PARMLIB(ASCHPMxx) 	
WLM/SRM	Grupo e domínio de desempenho do TSO <ul style="list-style-type: none"> • Sem padrão IBM (site dependente) 	
RACF	Cada usuário do Developer for System z possui acesso a um segmento OMVS (isto é necessário) <ul style="list-style-type: none"> • Sem padrão IBM (site dependente) • O comando RACF do TSO LU userid OMVS exibirá um segmento OMVS pessoal existente 	

Tabela 7. Lista de verificação de transações APPC (continuação)

Experiência	Informações necessárias:	Valor
	<ul style="list-style-type: none"> • Valor Padrão • Onde encontrar a resposta 	
RACF	<p>Cada usuário do Developer for System z deve ter acesso de LEITURA ao hlq.SFEKPROC (FEKFRSRV)</p> <ul style="list-style-type: none"> • Sem padrão IBM (site dependente) • O comando RACF do TSO LD AUTHUSER DATASET('hlq.SFEKPROC.**') exibirá usuários e grupos e seus níveis de acesso aos conjuntos de dados abrangidos pelo perfil do conjunto de dados 	

Consulte *MVS Planning: Workload Management (SA22-7602)* para obter informações adicionais sobre o gerenciamento do WLM/SRM. Consulte *Security Server RACF Security Administrator's Guide (SA22-7683)* para obter mais informações sobre segmentos OMVS e perfis de proteção de conjuntos de dados.

Nota: A classe de transação APPC utilizada deve ter inicializadores APPC suficientes para permitir um inicializador para cada usuário da função Remote Edit-Compile-Debug.

Nota: A transação APPC utiliza o executável REXX FEKFRSRV, localizado em hlq.SFEKPROC. Não altere esta localização se desejar que seja possível que a manutenção SMP/E seja ativada automaticamente.

Implementação

O programador do sistema ou administrador APPC precisa executar as seguintes etapas para configurar o recurso de comando:

1. Definir as informações de planejamento (classe) para o planejador de transações APPC se não estiver utilizando uma classe de transação existente. Inclua uma definição em SYS1.PARMLIB(ASCHPMxx) para a classe a ser utilizada pelo programa de transações FEKFRSRV. Essa classe é utilizada no JCL de amostra hlq.SFEKSAMP(FEKAPPCC). Portanto, a classe em FEKAPPCC deve corresponder à classe definida em SYS1.PARMLIB(ASCHPMxx). Por exemplo:

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

Nota: O serviço Comandos do TSO precisa que as especificações padrão sejam especificadas nas seções OPTIONS e TPDEFAULT de SYS1.PARMLIB(ASCHPMxx). Consulte o Apêndice F, “Configurando o APPC”, na página 117 para obter informações adicionais.

2. Defina a transação do APPC que agirá como um servidor de comandos. Você pode utilizar o JCL de amostra hlq.SFEKSAMP(FEKAPPCC) para definir essa transação. As instruções sobre como customizar este JCL estão localizadas no JCL. O JCL de amostra também é fornecido para exibir, hlq.SFEKSAMP(FEKAPPCL), ou excluir, hlq.SFEKSAMP(FEKAPPCX), a transação.

Nota: Se você tiver alterado o nome do programa de transações (padrão FEKFRSRV), o novo nome deverá ser designado como _FEKFSCMD_TP_NAME_ em rsed.envvars, conforme descrito em “Customizar rsed.envvars, o Arquivo de Configuração para RSE” na página 35.

3. Controlar a prioridade de dispatch do programa de transação hlq.SFEKPROC(FEKFRSRV) associando FEKFRSRV a um grupo de domínio e desempenho no WLM (Workload Manager). Como o FEKFRSRV emite comandos do TSO, ele deve ser designado a um grupo de desempenho TSO.
4. Definir o segmento OMVS padrão do sistema ou um segmento individual para cada usuário que precisa utilizar a função Remote Edit-Compile-Debug.
5. Forneça aos usuários do Developer for System z acesso de LEITURA ao hlq.SFEKPROC(FEKFSERV), o servidor de Comandos do TSO.

Nota: A verificação da configuração será realizada posteriormente, durante a verificação do RSE. Isto porque o RSE implementa a conexão TCP/IP no servidor de comandos do TSO.

(Opcional) Customizar ELAXM*, Membros do Procedimento Armazenado do DB2

A customização a seguir é necessária para incorporar a amostra do procedimento armazenado do DB2 (Stored Procedure Builder para PL/I e COBOL) em seu sistema. Você necessitará da assistência de um administrador WLM e um administrador DB2 para realizar estas tarefas.

Após a aplicação de SMP/E, a biblioteca de amostra hlq.SFEKSAMP e a biblioteca de procedimentos hlq.SFEKPROC contêm os membros do procedimento armazenado do DB2 listados na Tabela 8.

Tabela 8. Membros do procedimento armazenado DB2 do ELAXM de amostra*

Membro	Objetivo
hlq.SFEKSAMP(ELAXMJCL)	JCL de amostra para definição do Stored Procedure Builder para PL/I e COBOL para DB2.
hlq.SFEKSAMP(ELAXMSAM)	Procedimento JCL de amostra do espaço de endereços WLM para o Stored Procedure Builder para PL/I e COBOL.
hlq.SFEKPROC(ELAXMREX)	Código REXX para o Stored Procedure Builder para PL/I e COBOL.

Nota: O procedimento armazenado DB2 utiliza o executável REXX ELAXMREX, localizado em hlq.SFEKPROC. Não altere esta localização se desejar que seja possível que a manutenção SMP/E seja ativada automaticamente.

Nota: Consulte o Apêndice A, “Executando Várias Instâncias do Developer for System z”, na página 71 se quiser renomear os membros ELAXMSAM e ELAXMREX.

1. Copie ELAXMSAM para uma biblioteca de procedimentos (como SYS1.PROCLIB) disponível para os usuários do procedimento armazenado do DB2 e customize o JCL conforme descrito em seus comentários. Certifique-se de que o cartão SYSEXEC DD aponte para a biblioteca na qual o membro ELAXMREX reside (default hlq.SFEKPROC).
2. Utilize os painéis do WLM (workload management) para associar um ambiente de aplicativos ao procedimento JCL do espaço de endereços WLM para o Stored Procedure Builder para PL/I e COBOL. Consulte *MVS Planning: Workload Management (SA22-7602)* para obter informações sobre como fazer isto.

Nota: Você pode criar um novo ambiente de aplicativos no WLM para o Stored Procedure Builder para PL/I e COBOL, ou você pode incluir as definições necessárias em um ambiente existente.

3. Copie ELAXMJCL para uma biblioteca JCL privada, customize a cópia descrita em seus comentários e solicite que um administrador DB2 envie a tarefa. Certifique-se de que a cláusula WLM ENVIRONMENT na instrução CREATE PROCEDURE especifique o nome do procedimento do ambiente WLM que foi definido para o Construtor do Procedimento Armazenado PL/I e COBOL (padrão ELAXMSAM).

(Opcional) Customizar o Suporte à Linguagem Bidirecional (BIDI) do CICS

O componente EST (Enterprise Service Tools) do Developer for System z suporta formatos diferentes das mensagens de interface em árabe e hebraico, assim como a apresentação de dados bidirecionais e a edição em todos os editores e visualizações. Em aplicativos terminais, telas da esquerda para a direita e da direita para a esquerda são suportadas, assim como campos numéricos e campos com orientação oposta à tela.

A funcionalidade e os recursos bidirecionais adicionais incluem:

- O solicitante do serviço EST especifica de forma dinâmica os atributos bidirecionais das mensagens de interface.
- O processamento de dados bidirecional em fluxos de serviços tem como base atributos bidirecionais (tipo de texto, orientação de texto, troca numérica e troca simétrica). Esses atributos podem ser especificados em diferentes estágios da criação de fluxo para ambos os fluxos de interface e do terminal.
- O código de tempo de execução gerado pelo EST inclui a conversão dos dados entre os campos em mensagens que possuem atributos bidirecionais diferentes.

Atenção: O módulo de carregamento (nome e codificação) foi alterado em comparação com releases anteriores (antes da versão 7.0). Se você tiver ativado um release anterior do bidi, o(s) módulo(s) de carregamento antigo(s) deverá(ão) ser removido(s) da concatenação CICS RPL. O(s) local(is) padrão é(são):

- FEJ.SFEJDLL(FEJBDTRN)
- FEJ.SFEJLMD(FEJBDTRE) (opcional, versão não-PDSE do FEJBDTRN para hebraico)

Você precisará de assistência do administrador CICS para realizar as seguintes tarefas:

1. Coloque o programa hlq.SFEKLOAD(FEJBDTRX) na concatenação da RPL do CICS (instrução DD DFHRPL). É recomendável fazer isso incluindo o conjunto de dados da instalação na concatenação para que a manutenção aplicada esteja automaticamente disponível ao CICS.

As transformações bidirecionais do EST são realizadas no ambiente CICS SFR (Service Flow Runtime) pelo módulo FEJBDTRX. O módulo FEJBDTRX é chamado quando são necessárias conversões bidirecionais no código gerado pelo EST (quando mapeado, é gerado entre os campos nas mensagens que possuem atributos bidirecionais diferentes.)

Nota: Se você não concatenar o diretório de instalação, mas copiar o módulo FEJBDTRX para um conjunto de dados novo/existente, lembre-se de que este módulo é uma DLL e DEVE residir em uma biblioteca PDSE.

Nota: Na versão 7.1, o programa hlq.SFEKDLL(FEJBDTRX) movido para uma nova biblioteca de carregamento, hlq.SFEKLOAD(FEJBDTRX).

2. Solicite ao administrador do CICS a configuração da instalação automática como automaticamente ativa.

Se a instalação automática estiver configurada como autoINactive, defina o programa FEJBDTRX como CICS, utilizando o comando CEDA, por exemplo:

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

Além disso, o código gerado pelo EST pode suportar a transformação bidi em ambientes diferentes do CICS SFR (por exemplo, aplicativos em lote). Você pode fazer com que os geradores EST incluam chamadas nas rotinas de conversão bidirecional especificando as opções de transformação bidi apropriadas nos assistentes de geração EST e vinculando os programas gerados com a biblioteca de conversão bidirecional apropriada, hlq.SFEKLOAD.

Atenção: O código bidi que foi criado com releases anteriores (antes da versão 7.0) deve ser RECOMPILADO para utilizar o novo módulo FEJBDTRX.

(Opcional) Customizar o ADM (Application Deployment Manager)

O ADM (Application Deployment Manager) fornece uma abordagem de implementação comum e API de implementação para todos os componentes do Developer for System z.

Além disso, o ADM fornece um cliente e servidor (baseado em host) do CRD (CICS Resource Definition), que fornece as seguintes funções:

1. Permitir que os desenvolvedores de aplicativos definam recursos do CICS de forma limitada, controlada e segura.
 - Os padrões da definição de recurso do CICS são fornecidos pelo administrador do CICS e armazenados no repositório do servidor CRD no host.
 - A Capacidade de Atualização de Atributo de Recurso é controlada pelo administrador do CICS (atualização, proteção, oculto).
 - O CRD é limitado a um pequeno conjunto de recursos geralmente necessários aos desenvolvedores de aplicativos, incluindo DB2Tran, Doctemplate, File, Mapset, Processtype, Program, TDQ, e Transaction.
 - A autorização para criar definições de recurso do CICS é controlada pelo RACF ou outro gerenciador de segurança externo.
2. Impedir o acesso de desenvolvimento do CICS a conjuntos de dados VSAM não-autorizados ou incorretos, fornecendo ao administrador do CICS controle sobre o atributo de nome do conjunto de dados físico nas definições de Arquivo. Essas informações sobre ligação são armazenadas no repositório do CRD no host.
3. Auxílios diversos de desenvolvimento do servidor CRD.
 - Nova cópia para programas e conjuntos de mapas
 - Lista de regiões do CICS
 - Lista DFHRPL
4. Auxílios diversos de desenvolvimento de serviço da Web do servidor CRD.
 - Desempenha varredura de Pipeline para instalar automaticamente as definições URIMAP e WEBSERVICE
 - Fornecer lista de diretórios de recebimento de Pipeline e WSBind
 - Fornecer lista de diretórios de arquivo WSDL

- Fornecer lista de URIs de Terminais

O Developer for System z fornece três transações que são utilizadas pelo servidor CRD ao definir e consultar recursos do CICS. O código-fonte COBOL de amostra é fornecido para permitir customizações específicas do site.

ADMD

Para pedidos que definam o serviço da Web e padrões de recurso do CICS. Geralmente, isso é destinado aos administradores do CICS.

ADMI Para pedidos que definam, instalem ou desinstalem recursos do CICS.

ADMR

Para todos os outros pedidos que recuperem informações ambientais ou sobre recurso do CICS.

Consulte o *IBM Rational Developer para System z Application Deployment Manager User's Guide (SC31-6972)* para obter mais informações sobre o ADM.

As etapas de customização a seguir são necessárias para ativar o servidor CRD.

Nota: Você precisará de assistência do administrador do CICS para desempenhar algumas das tarefas a seguir.

Antes de instalar a versão 7.1, se você for um usuário anterior do servidor CRD, é recomendável salvar a customização relacionada, conforme descrito em “Fazendo Backup de Arquivos Configurados Anteriormente” na página 11.

Copie os membros a serem customizados do diretório de instalação para uma biblioteca pessoal e customize estas cópias para evitar que sejam sobrescritas ao realizar manutenção:

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

opcional (Manipulador de Mensagem de Pipeline)

- hlq.SFEKSAMP(ADNCMSGH)
- hlq.SFEKSAMP(ADNTMSGH)

opcional (atualização CSD para regiões não-primárias)

- hlq.SFEKSAMP(ADNARCSD)

Repositório do CRD

Customize e envie a tarefa ADNVSAM para alocar e inicializar o arquivo VSAM do repositório do servidor CRD. Consulte a documentação em ADNVSAM para obter instruções de customização.

É aconselhável criar um repositório separado para cada região de conexão primária do CICS. O compartilhamento do repositório implica que todas as regiões relacionadas ao CICS utilizarão os mesmos valores armazenados no repositório e que vários espaços de endereço serão gravados no VSAM, que deve ser configurado corretamente para isso.

Nota: A menos que seja notificado o contrário, o repositório do servidor CRD (contendo os valores customizados) pode ser reutilizado nos releases do Developer for System z.

Região de Conexão Primária do CICS

O servidor CRD deve ser definido para a região de conexão primária. Essa é a região que processará pedidos de Serviço da Web do Developer for System z.

- Coloque os módulos de carregamento ADM hlq.SFEKLOAD(ADNCRDS) e hlq.SFEKLOAD(ADNCRDR) na concatenação RPL do CICS (instrução DD DFHRPL) da região de conexão primária do CICS. É recomendável fazer isso incluindo o conjunto de dados da instalação na concatenação para que a manutenção aplicada esteja automaticamente disponível ao CICS
- Customize e envie a tarefa ADNPCCSD para atualizar a CSD (CICS System Definition) para a região de conexão primária do CICS. Consulte a documentação em ADNPCCSD para obter instruções de customização
- Utilize o comando CEDA apropriado para instalar o grupo ADM para esta região, por exemplo:
CEDA INSTALL GROUP(ADNPCRGP)

Manipulador de Mensagem de Pipeline

O manipulador de mensagem de pipeline (ADNTMSGH) é utilizado para processamento do ID do usuário e senha no cabeçalho SOAP. ADNTMSGH é mencionado no arquivo de configuração de pipeline e, portanto, deve ser colocado na concatenação RPL do CICS. ADNTMSGH também é utilizado para definir o ID da transação como ADMD, ADMR ou ADMI, dependendo da operação solicitada. É possível customizar ADNTMSGH para utilizar IDs de transação diferentes.

Utilizando o padrão:

- Coloque o módulo de carregamento hlq.SFEKLOAD(ADNTMSGH) na concatenação RPL do CICS (instrução DD DFHRPL) da região de conexão primária do CICS. É recomendável fazer isso incluindo o conjunto de dados da instalação na concatenação para que a manutenção aplicada esteja automaticamente disponível ao CICS.

Customizando ADNTMSGH:

- Customize o código-fonte (COBOL) de ADNTMSGH para o Manipulador de Mensagem de Pipeline.
- Customize e envie a tarefa ADNCMSGH para compilar a origem ADNTMSGH customizada. Consulte a documentação em ADNCMSGH para obter instruções de customização.
- Coloque o módulo de carregamento ADNTMSGH resultante na concatenação RPL do CICS (instrução DD DFHRPL) da região de conexão primária do CICS.

Nota: Certifique-se de que o módulo de carregamento ADNTMSGH customizado esteja localizado antes de qualquer referência para hlq.SFEKLOAD; caso contrário, o padrão será utilizado.

(Opcional) Regiões de Conexão Não-primária do CICS

O servidor CRD também pode ser utilizado com uma ou mais regiões de conexão não-primárias, que geralmente são AORs (Application Owning Regions).

Nota: Não é necessário desempenhar essas etapas se o CICSplex SM for utilizado para gerenciar o ambiente do CICS.

- Coloque o módulo de carregamento ADM hlq.SFEKLOAD(ADNCRDS) na concatenação RPL do CICS (instrução DD DFHRPL) dessas regiões de conexão

não-primária. É recomendável fazer isso incluindo o conjunto de dados da instalação na concatenação para que a manutenção aplicada esteja automaticamente disponível ao CICS.

- Customize e envie a tarefa ADNARCSD para atualização do CSD para outras regiões de conexão não-primárias. Consulte a documentação em ADNARCSD para obter instruções de customização
- Utilize o comando CEDA apropriado para instalar o grupo ADM para estas regiões, por exemplo:
CEDA INSTALL GROUP(ADNARRGP)

Capítulo 4. Ativando Componentes do z/OS UNIX para o Developer for System z

Antes de instalar a versão 7.1, se você for um usuário anterior do WebSphere Developer for System z (FMID: HHOP700), é recomendável salvar a customização relacionada, descrita e “Fazendo Backup de Arquivos Configurados Anteriormente” na página 11.

Se não estiver familiarizado com o z/OS UNIX, é aconselhável solicitar assistência de um administrador z/OS UNIX ou outro administrador UNIX experiente para desempenhar as tarefas listadas neste capítulo.

Os comandos z/OS UNIX necessários para desempenhar as tarefas listadas são brevemente descritos para sua conveniência. A menos que observado de outra forma, consulte *UNIX System Services Command Reference (SA22-7802)* para obter informações adicionais sobre estes comandos.

As tarefas descritas abaixo esperam que você esteja ativo no z/OS UNIX. Isso pode ser feito emitindo o comando do TSO **OMVS**. Utilize o comando **exit** para retornar ao TSO.

O MVS fornece a possibilidade de editar arquivos z/OS UNIX utilizando o ISPF por meio do comando **OEDIT**. Este comando pode ser utilizado no TSO e no OMVS.

A maioria dos arquivos z/OS UNIX possui a permissão de gravação restrita ao proprietário do arquivo. Essa restrição pode ser ignorada de várias maneiras.

- **UID 0**

Isso não é recomendado para IDs de usuário “humanos”, pois não há restrições relacionadas ao z/OS UNIX.

- **Acesso de LEITURA ao perfil BPX.SUPERUSER na classe FACILITY**

Permite que o usuário se torne o UID 0 por meio do comando **su**. Essa é a configuração recomendada.

- **Acesso de ATUALIZAÇÃO ao perfil SUPERUSER.FILESYS na classe UNIXPRIV**

Permite que o usuário leia/grave qualquer arquivo e leia ou procure qualquer diretório. O acesso de **CONTROLE** (ou superior) a esse perfil de segurança inclui gravação em qualquer diretório à lista de permissões.

Consulte *UNIX System Services Planning (GA22-7800)* para saber mais sobre a segurança do z/OS UNIX.

Todas as instruções de caminho `/usr/lpp/wd4z/` neste capítulo se referem ao caminho utilizado durante a instalação do Developer for System z. O padrão é `/usr/lpp/wd4z/`, mas isto pode não se aplicar ao seu site.

Nota: O Developer for System z depende de o TCP/IP ter o nome do host correto quando for inicializado. Isto significa que o TCP/IP diferente e os arquivos de configuração do Resolver devem ser configurados corretamente. Consulte Apêndice C, “Configurando o TCP/IP”, na página 89 para obter mais informações sobre a customização necessária.

Você pode testar sua configuração TCP/IP com o comando **HOMETEST** do TSO. *Communications Server: IP System Administrator's Commands (SC31-8781)* para obter mais informações sobre esse comando.

Nota: O Developer for System z depende de INETD para configuração da conexão de host do cliente. Consulte Apêndice D, “Configurando o INETD”, na página 97 para obter mais informações sobre o INETD.

Nota: Ações remotas (baseadas em host) para subprojetos do z/OS UNIX requerem que o REXEC ou o SSH esteja ativo no host.

Nota: Java de 31 bits é necessário e todos os usuários do z/OS UNIX devem possuir acesso de EXECUÇÃO e de LEITURA aos diretórios Java HFS.

Atenção: Java 64 bits NÃO é suportado.

Você pode localizar as seguintes publicações úteis sobre noções do 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)*

Salvando o Arquivo de Configuração rsed.envvars em um Outro Diretório

É recomendável copiar o arquivo `/usr/lpp/wd4z/rse/lib/rsed.envvars` para um novo diretório (tal como o `/etc/wd4z/`) e customizar a cópia para evitar sobrescrever sua customização ao aplicar a manutenção. Porém, ao fazer isto, você também deve copiar os seguintes arquivos do diretório de instalação (o padrão é `/usr/lpp/wd4z/rse/lib/`) para o novo local:

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

Nota: Embora não haja customização necessária para os arquivos `*.zseries`, é importante substituir as versões anteriores pelas atuais, para que fiquem sincronizadas com o `rsed.envvars` atual.

Os arquivos listados em Tabela 8 na página 26 devem ser copiados também se você planeja utilizar os recursos opcionais que fazem parte de:

Tabela 9. Arquivos de Configuração Opcionais

arquivo	definição
<code>projectcfg.properties</code>	Projetos Baseados em Host Consulte “(Opcional) Customizar <code>projectcfg.properties</code> , Configuração de Projetos do Host” na página 52
<code>FMIEXT.properties</code>	Integração do File Manager Consulte “(Opcional) Customizar <code>FMIEXT.properties</code> , Integração do File Manager” na página 53

Tabela 9. Arquivos de Configuração Opcionais (continuação)

arquivo	definição
CRASRV.properties	CARMA Consulte Capítulo 5, “(Opcional) Ativando o IBM CARMA”, na página 55

Os seguintes comandos de amostra,

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

crie um diretório chamado `/etc/wd4z` e copie `rsed.envvars` do diretório atual para `/etc/wd4z`. Repita o comando de cópia para os arquivos restantes.

O resultado da cópia pode ser verificado com o comando `ls /etc/wd4z`, que deve apresentar uma saída semelhante a esta (\$ é o prompt do z/OS UNIX):

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

Nota: Se desejar manter todos os arquivos do Developer for System z no mesmo HFS (privado), mas também desejar que os arquivos de configuração sejam colocados em `/etc/wd4z`, você pode utilizar links simbólicos para resolver esse problema. Os comandos de amostra a seguir criam um novo diretório (`/usr/lpp/wd4z/rse/lib/cust`) no HFS de instalação e definem um link simbólico (`/etc/wd4z`) para ele:

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

Customizar `rsed.envvars`, o Arquivo de Configuração para RSE

Todos os métodos de conexão do cliente do Developer for System z utilizam as variáveis configuradas no arquivo `rsed.envvars`, localizado por padrão no diretório de instalação `/usr/lpp/wd4z/rse/lib/`, mas poderia ser copiado para outro diretório na etapa anterior. O arquivo de amostra fornecido possui as instruções listadas em Figura 4 na página 36, nas quais as linhas de comentário iniciam com um sinal de libras (#).

```

#=====
# (1) customizações necessárias
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="$_RSE_JAVAOPTS -DTSO_SERVER=APPC"
#=====
# (2) customizações necessárias se SCLMDT for utilizado
_CMDSERV_BASE_HOME=/usr/lpp/SCLMDT
_CMDSERV_BASE_LOAD=BWB.SBWBLOAD
_CMDSERV_CONF_HOME=/etc/SCLMDT
_CMDSERV_WORK_HOME=/var/SCLMDT
STEPLIB=NONE
#STEPLIB=$_CMDSERV_BASE_LOAD
_RSE_CMDSERV_OPTS=""
#=====
# (3) customizações opcionais
#_RSE_PORTRANGE=8108-8118
#_FEKFLOCK_USERID=user_id
#_FEKFLOCK_JOBNAME=job_name
#_FEKFSCMD_TP_NAME=tp_name
#_FEKFSCMD_PARTNER_LU=lu_name
#=====
# (4) não altere, a menos que orientado pelo 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) variáveis de ambiente adicionais

```

Figura 4. rsed.envvars – arquivo de configuração RSE

As seguintes definições são requeridas:

JAVA_HOME

Diretório inicial Java. O padrão é /usr/lpp/java/J1.4. Altere para corresponder com a instalação do Java.

RSE_HOME

Diretório inicial do RSE. O padrão é /usr/lpp/wd4z. Altere para corresponder à instalação do Developer for System z.

TZ Seletor de fuso horário. O padrão é EST5EDT. O fuso horário padrão é UTC +5 horas (horário de verão do horário padrão na costa leste dos EUA). Altere para corresponder ao fuso horário. Informações adicionais podem ser localizadas no *UNIX System Services File System Interface Reference (SA22-7808)*.

LANG

Especifica o nome do código do idioma padrão. O padrão é C. C especifica o código do idioma POSIX e Ja_JP especifica o código do idioma em japonês. Altere para corresponder ao código do idioma.

PATH Caminho de comandos. O padrão é /bin:/usr/sbin:.. Pode ser alterado, se necessário.

_RSE_CLASS_OPTS

Opções Java adicionais para compartilhamento de classe. O padrão é "". Consulte "(Opcional) Definindo Parâmetros Adicionais de Inicialização Java com _RSE_*OPTS" na página 41 para obter informações adicionais sobre esta definição.

_RSE_JAVAOPTS

Opções Java específicas adicionais do RSE. O padrão é "". Consulte "(Opcional) Definindo Parâmetros Adicionais de Inicialização Java com _RSE_*OPTS" na página 41 para obter informações adicionais sobre esta definição.

Por padrão, o Developer for System z utiliza o SCLM Developer Toolkit para o serviço Comandos do TSO. O APPC é utilizado quando a seguinte opção _RSE_JAVAOPTS está sem comentários:

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

Nota: Ambos os métodos do serviço Comandos do TSO requerem mais customizações que apenas as customizações em rsed.envvars. As customizações necessárias para a configuração do APPC são descritas em "(Opcional) Definir uma Transação APPC para o Serviço de Comandos do TSO" na página 23, e as de SCLMDT são descritas em "Customizar ISPF.conf, o Arquivo de Configuração do ISPF" na página 45.

As seguintes definições são necessárias se o SCLM Developer Toolkit for utilizado, para o serviço Comandos do TSO ou quando o plug-in SCLMDT é instalado no cliente do Developer for System z.

_CMDSERV_BASE_HOME

Diretório inicial do SCLM Developer Toolkit. O padrão é /usr/lpp/SCLMDT. Altere para corresponder à instalação do SCLM Developer Toolkit. Essa diretiva só é necessária quando o SCLM Developer Toolkit é utilizado (serviço Comandos do TSO ou plug-in do cliente).

_CMDSERV_BASE_LOAD

Biblioteca de carregamento do SCLM Developer Toolkit. O padrão é BWB.SWBLOAD. Altere para corresponder à instalação do SCLM

Developer Toolkit. Essa diretiva só é necessária quando o SCLM Developer Toolkit é utilizado (serviço Comandos do TSO ou plug-in do cliente).

_CMDSERV_CONF_HOME

Diretório de configuração base do SCLM Developer Toolkit. O padrão é /etc/SCLMDT. Altere para corresponder à customização do SCLM Developer Toolkit. Essa diretiva só é necessária quando o SCLM Developer Toolkit é utilizado (serviço Comandos do TSO ou plug-in do cliente).

_CMDSERV_WORK_HOME

Diretório de trabalho base do SCLM Developer Toolkit. O padrão é /var/SCLMDT. Altere para corresponder à customização do SCLM Developer Toolkit. Essa diretiva só é necessária quando o SCLM Developer Toolkit é utilizado (serviço Comandos do TSO ou plug-in do cliente).

STEPLIB

STEPLIB para o servidor RSE. O padrão é NONE. Não altere essa linha, pois funciona como um valor padrão.

O Developer for System z utiliza o LINKLIST por padrão para acessar a biblioteca de carregamento do SCLM Developer Toolkit. O STEPLIB é utilizado quando a seguinte diretiva STEPLIB está sem comentários:

```
STEPLIB=$_CMDSERV_BASE_LOAD
```

Nota: A utilização de STEPLIB em z/OS UNIX tem um impacto negativo no desempenho, conforme descrito em “Evite o Uso de STEPLIB” na página 65.

_RSE_CMDSERV_OPTS

Opções Java específicas adicionais do serviço Comandos do TSO. O padrão é "". Consulte “(Opcional) Definindo Parâmetros Adicionais de Inicialização Java com _RSE_*OPTS” na página 41 para obter informações adicionais sobre esta definição. Essa diretiva só é necessária quando o SCLM Developer Toolkit é utilizado (serviço Comandos do TSO ou plug-in do cliente).

As definições a seguir são opcionais. Se omitido, os valores padrão serão utilizados.

_RSE_PORTRANGE

Especifica o intervalo de portas que o servidor RSE pode abrir para comunicação com um cliente. Qualquer porta pode ser utilizada por padrão. Consulte “(Opcional) Definindo o PORTRANGE Disponível para RSE” na página 40 para obter informações adicionais sobre esta definição.

_FEKFLOCK_USERID_

O ID de usuário deve ser utilizado pelo gerenciador de bloqueios. O padrão é o ID do usuário de logon.

_FEKFLOCK_JOBNAME_

Nome da tarefa a ser utilizada pelo gerenciador de bloqueios. O padrão é FEKFLK00.

_FEKFSCMD_TP_NAME_

Nome do programa de transações APPC. O valor padrão é FEKFRSRV. Remova o comentário e altere esta definição se você não utilizar o nome padrão do programa de transações ao definir a transação APPC. Consulte “(Opcional) Definir uma Transação APPC para o Serviço de Comandos do TSO” na página 23 para obter informações adicionais sobre a transação APPC.

_FEKFSCMD_PARTNER_LU_

Force o RSE a utilizar esta LU de base APPC. Consulte o Apêndice F, “Configurando o APPC”, na página 117 para obter informações adicionais sobre esta definição.

As seguintes definições são necessárias e não devem ser alteradas, a menos que orientado pelo IBM Support Center.

RSE_LIB

Caminho da biblioteca RSE. O padrão é `$RSE_HOME/rse/lib`. Não modifique.

ICU_LIB

Caminho da Biblioteca de ICU (International Components for Unicode). O padrão é `$RSE_HOME/icuc/lib`. Não modifique.

_CEE_RUNOPTS

Opções do tempo de execução do LE (Language Environment) utilizadas pelos processos iniciados. O padrão é `"ALL31(ON) HEAP(32M,32K,ANYWHERE,KEEP,,) TRAP(ON)"`. Não modifique.

_CEE_DMPTARG

Local de dump do LE (Language Environment) do z/OS UNIX utilizado pela JVM (Java Virtual Machine). O padrão é `$HOME/.eclipse/RSE/$RSE_USER_ID`. Não modifique.

_BPX_SHAREAS

Executar processos em primeiro plano no mesmo espaço de endereço que o shell. O padrão é YES. Não modifique.

_BPX_SPAWN_SCRIPT

Execute scripts de shell diretamente da função `spawn()`. O padrão é YES. Não modifique.

PATH Caminho de comandos. O padrão é `$JAVA_HOME/bin:$RSE_LIB:$_CMDSERV_BASE_HOME/lib:$PATH`. Não modifique.

LIBPATH

Caminho de bibliotecas. O padrão é `$JAVA_HOME/bin:$JAVA_HOME/bin/classic:$ICU_LIB:$RSE_LIB:..`. Não modifique.

CLASSPATH

Caminho de classes. O padrão é longo demais para repetir. Não modifique.

_RSE_CMDSERV_OPTS

Opções Java específicas adicionais do serviço Comandos do TSO. O padrão é `"&SESSION=SPAWN$_RSE_CMDSERV_OPTS"`. Não modifique.

_RSE_JAVAOPTS

Opções Java específicas adicionais do RSE. O padrão é longo demais para repetir. Não modifique.

_RSE_SERVER_CLASS

Classe Java do servidor RSE. O padrão é `com.ibm.etools.systems.dstore.core.server.Server`. Não modifique.

_RSE_SERVER_TIMEOUT

Valor de tempo limite do servidor RSE (aguardando o cliente) em milissegundos. O padrão é 120000 (2 minutos). Não modifique.

Nota: Você pode desconsiderar a necessidade de ter bibliotecas C/C++ e LE (Language Environment) em LINKLIST, incluindo a seguinte instrução

STEPLIB no FIM de `rsed.envvars` (os nomes dos conjuntos de dados podem variar em seu site). No entanto, lembre-se de que a utilização de STEPLIB em z/OS UNIX tem um impacto negativo no desempenho, conforme descrito em “Evite o Uso de STEPLIB” na página 65.

- Se a última diretiva STEPLIB definida anteriormente em `rsed.envvars` for igual a STEPLIB=NONE
STEPLIB=CEE.SCEERUN:CEE.SCEERUN2:CBC.SCLBDLL
- Se a última diretiva STEPLIB definida anteriormente em `rsed.envvars` não for igual a STEPLIB=NONE
STEPLIB=\$STEPLIB:CEE.SCEERUN:CEE.SCEERUN2:CBC.SCLBDLL

Nota: Links simbólicos são permitidos ao especificar diretórios em `rsed.envvars`.

(Opcional) Definindo o PORTRANGE Disponível para RSE

Essa é uma parte da customização de `rsed.envvars` que especifica as portas nas quais o servidor RSE pode se comunicar com o cliente. Este intervalo de portas não possui nenhuma conexão com o daemon RSE ou portas REXEC/SSH.

Para ajudar a compreender o uso da porta, segue uma breve descrição do processo de conexão do RSE:

1. O cliente se conecta à porta 4035 do host, serviço do daemon RSE INETD, à porta 512 do host, serviço REXEC INETD, ou à porta 22 do host, serviço SSH INETD.
2. O serviço do INETD escolhido cria um processo RSE.
3. O processo RSE abre uma porta do host para a conexão do cliente. A seleção dessa porta pode ser configurada pelo usuário, no cliente na guia de propriedades do subsistema (isso não é recomendado) ou por meio da definição `_RSE_PORTRANGE` em `rsed.envvars`.
4. O serviço do INETD retorna o número da porta para o cliente.
5. O cliente se conecta à porta do host.

Para especificar o intervalo de portas, para o cliente se comunicar com o z/OS, remova o comentário e altere a seguinte linha no `rsed.envvars`:

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

Nota: Antes de selecionar um intervalo de portas, verifique se o intervalo está disponível em seu sistema com os comandos **NETSTAT** e **NETSTAT PORTL**. Consulte “Portas TCP/IP Reservadas” na página 81 para obter informações adicionais.

O formato de PORTRANGE é: `_RSE_PORTRANGE=min-max` (max não está incluso; por ex., `_RSE_PORTRANGE=8108-8118` significa que os números de portas de 8108 até 8117 são utilizáveis). O número da porta utilizado pelo servidor RSE é determinado na seguinte ordem:

1. Se um número de porta diferente de zero for especificado nas propriedades do subsistema no cliente, o número da porta será utilizado. Se a porta não estiver disponível, a conexão falhará. Esta configuração não é recomendada.
2. Se um número de porta nas propriedades do subsistema for 0, e se `_RSE_PORTRANGE` for especificado no `rsed.envvars`, o intervalo de portas especificado pelo `_RSE_PORTRANGE` será utilizado. Se nenhuma porta no intervalo estiver disponível, a conexão falhará.

3. Se um número de porta nas propriedades do subsistema for 0, e `_RSE_PORTRANGE` não for especificado no `rsed.envvars`, nenhuma porta disponível será utilizada.

Nota: Quando um servidor abrir uma porta e estiver atendendo, o número da porta não poderá ser utilizado por outro servidor, porém, depois de conectado, o mesmo número de porta poderá ser utilizado novamente. Isso significa que o número de portas no intervalo não limita o número de usuários conectados simultaneamente.

(Opcional) Definindo Parâmetros Adicionais de Inicialização Java com `_RSE_*OPTS`

Com as diferentes diretivas `_RSE_*OPTS`, `rsed.envvars` oferece a possibilidade de fornecer parâmetros adicionais para Java quando ele inicia o servidor RSE.

As opções de amostra incluídas em `rsed.envvars` podem ser ativadas pela remoção dos comentários.

`_RSE_JAVAOPTS`

`_RSE_JAVAOPTS` define opções Java e específicas do RSE.

`_RSE_JAVAOPTS=""`

Inicialização variável. Não modifique.

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

Defina o tamanho de heap inicial (Xms) e máximo (Xmx). Os padrões do sistema são 1M e 64M, respectivamente. Remova o comentário e altere para impor os valores do tamanho de heap especificados.

`#_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Dfile.encoding=Cp424"`

Seleção da página de códigos do host. Remova o comentário e altere para impor o uso da página de códigos especificada.

`#_RSE_JAVAOPTS="$_RSE_JAVAOPTS`

`-DDSTORE_TRACING_ON=true"`

Inicie o rastreamento do dstore. Utilize apenas quando orientado pelo IBM Support Center.

`#_RSE_JAVAOPTS="$_RSE_JAVAOPTS`

`-DDSTORE_MEMLOGGING_ON=true"`

Inicie o rastreamento de memória do dstore. Utilize apenas quando orientado pelo IBM Support Center.

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

Utilize o APPC para o serviço Comandos do TSO. Consulte o "(Opcional) Definir uma Transação APPC para o Serviço de Comandos do TSO" na página 23 para obter informações adicionais.

`_RSE_CLASS_OPTS`

A diretiva `_RSE_CLASS_OPTS` define as opções Java 5.0 (e posterior) necessárias para compartilhar classes entre vários servidores RSE. Consulte o "Compartilhamento de Classe entre JVMs" na página 67 para obter informações adicionais.

`_RSE_CLASS_OPTS=""`

Inicialização variável. Não modifique.

#_RSE_CLASS_OPTS=-Xshareclasses:name=RSE,groupAccess,nonFatal
Java 5.0 e posterior apenas. Ative o compartilhamento de classe.
Remova o comentário para compartilhar classes entre vários servidores RSE.

#_RSE_CLASS_OPTS="\$_RSE_CLASS_OPTS -Xscmx6m"
Java 5.0 e posterior apenas. Configure o tamanho do cache da classe compartilhada. O padrão do sistema é 16M.

_RSE_CMDSERV_OPTS

As diretivas `_RSE_CMDSERV_OPTS` são opções Java específicas ao RSE e estão em efeito apenas quando o SCLM Developer Toolkit é utilizado como o Servidor de Comandos do TSO.

_RSE_CMDSERV_OPTS=""
Inicialização variável. Não modifique.

#_RSE_CMDSERV_OPTS="\$_RSE_CMDSERV_OPTS&ISPPROF=&SYSUID..ISPPROF"
Utilize um perfil ISPF existente para o Servidor de Comandos do TSO. Remova o comentário e altere o nome do conjunto de dados para utilizar o perfil ISPF especificado. `&SYSUID.` pode ser utilizado como uma substituição para o ID do usuário do desenvolvedor.

Configuração do Daemon INETD e REXEC/SSH do RSE

O Developer for System z depende do serviço do INETD para iniciar o Servidor do RSE (Remote Systems Explorer) quando um cliente solicita uma conexão. O INETD é um daemon z/OS UNIX padrão, que gerencia outros daemons que realizam o trabalho real (nesse caso, começando com o servidor RSE). A configuração do INETD não faz parte da customização do Developer for System z, mas informações importantes podem ser encontradas em Apêndice D, "Configurando o INETD", na página 97.

O Developer for System z suporta várias formas de iniciar o servidor RSE.

É necessário customizar pelo menos uma forma, dependendo de como os usuários planejam operar.

- O daemon RSE, iniciado pela conexão ao INETD na porta 4035 (padrão). Este é o método recomendado porque ele fornece ao usuário mais controle e segurança que o REXEC.
- O Servidor de Comandos do REXEC (Remote Execution), porta 512 padrão, executa um script de shell que chama o RSE. Este método de conexão é utilizável para prova de conceito, mas não para utilização a longo prazo. A vantagem da utilização deste método é que ele requer menos trabalho de instalação e configuração, se REXEC foi configurado antes (normalmente durante a configuração do TCP/IP).
- O Servidor de Comandos do SSH (Secure Shell), porta 22 padrão, executa um script de shell que chama o RSE. Esse método é comparável com o REXEC, mas o SSH utiliza comunicação segura (criptografada) com o cliente.

Nota: Ações remotas (baseadas em host) para subprojetos do z/OS UNIX requerem que o REXEC ou o SSH esteja ativo no host.

Você pode verificar que o INETD está ativo com o comando **ps -e** (fornecido por um usuário autorizado). A saída deve conter uma referência ao INETD, por exemplo (# é o prompt do z/OS UNIX):


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

Nota: Para que servidores z/OS UNIX (como o daemon RSE, REXEC e SSH) suportem conexões IPv6, o tcp6 deve ser especificado para o protocolo do nome do serviço no arquivo inetd.conf. Quando tcp6 é definido, os clientes IPv4 também são suportados.

Configuração do Daemon INETD RSE

1. Modifique /etc/services incluindo esta linha:

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

rse Nome do serviço do daemon, o padrão é rse (minúsculo). O nome deve corresponder com o nome que será utilizado no /etc/inetd.conf

4035/tcp

Porta e protocolo utilizados, a porta padrão é 4035, o protocolo deve ser tcp.

A porta utilizada deve corresponder com a porta definida no cliente, a qual é configurada durante a criação de uma nova conexão do z/OS.

Nota: Antes de selecionar uma porta, verifique se ela está disponível em seu sistema com os comandos **NETSTAT** e **NETSTAT PORTL**. Consulte “Portas TCP/IP Reservadas” na página 81 para obter informações adicionais.

#Developer for System z RSE

comentário

2. Modifique /etc/inetd.conf, incluindo estas duas linhas. Consulte Apêndice D, “Configurando o INETD”, na página 97 para obter as regras de continuação.

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

rse Nome do serviço do daemon. O padrão é rse (minúscula). O nome deve corresponder ao nome utilizado em /etc/services

stream tcp nowait

Instruções de configuração específicas ao INETD (tipo de soquete, protocolo, sinalizador de espera). Não modifique.

OMVSKERN

ID do usuário para o processo do daemon RSE. O padrão é OMVSKERN. Esse ID do usuário deve ser um ID de usuário com um segmento de segurança OMVS válido, permissão BPX.DAEMON e permissão de LEITURA e de EXECUÇÃO para os diretórios de instalação e configuração do Developer for System z. Consulte Apêndice D, “Configurando o INETD”, na página 97 para obter mais detalhes sobre os requisitos para os IDs de usuário utilizados para os serviços do sistema.

/usr/lpp/wd4z/rse/lib/fekfrsed

Programa do servidor (local absoluto de fekfrsed). O padrão é /usr/lpp/wd4z/rse/lib/fekfrsed

Tudo após este argumento INETD são argumentos do servidor, iniciando com o nome do servidor.

rsed Nome do servidor. Não modifique.

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

Diretório de trabalho (local dos arquivos de configuração do servidor RSE). O padrão é /usr/lpp/wd4z/rse/lib

Nota: É recomendável que você copie os arquivos de configuração RSE customizados para um novo diretório (tal como o /etc/wd4z/) para evitar sobrescrevê-los ao realizar manutenção. O diretório de trabalho definido aqui deve refletir esta alteração. Por exemplo:

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

As definições a seguir são opcionais. Se omitido, os valores padrão serão utilizados.

-t 60 Opção de tempo limite para especificar quantos segundos o daemon RSE aguarda pela resposta do servidor RSE. O padrão é 60 segundos. O tempo limite para o servidor RSE aguardando no cliente é definido em rsed.envvars e é 2 minutos por padrão.

3. O INETD deve ser reiniciado por um usuário autorizado para ativar as alterações feitas nos arquivos /etc, conforme descrito em Apêndice D, “Configurando o INETD”, na página 97. Consulte os seguintes comandos de amostra (# é o prompt do 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
```

Nota: Se o perfil BPX.DAEMON estiver definido na classe FACILITY do produto de segurança e o usuário que estiver (re)iniciando o INETD não tiver acesso a esse recurso, então o aviso de segurança a seguir poderá ser esperado para cada cliente que se conectar ao RSE, em que IBMUSER é o ID do usuário utilizado para iniciar o INETD.

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

Configuração do INETD REXEC (ou SSH)

Não existe nenhuma configuração específica do Developer for System z para utilização do servidor de Comandos INETD REXEC (ou SSH). Entretanto, o cliente precisa conhecer 2 valores para iniciar uma conexão RSE por meio do REXEC/SSH:

- O diretório no qual o script de inicialização server.zseries está localizado. Por padrão, esse é o diretório de instalação (/usr/lpp/wd4z/rse/lib/). Entretanto, server.zseries é um dos arquivos que também devem ser copiados se rsed.envvars for copiado para um diretório diferente, como /etc/wd4z/.
- A porta que está sendo utilizada. Uma porta comum utilizada pelo REXEC é 512. Uma forma rápida de verificar que esse é o comando **NETSTAT**, conforme mostrado na amostra a seguir (\$ é o prompt do z/OS UNIX):

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

Para verificar isso, você pode verificar os serviços /etc/inetd.conf e /etc/services para localizar o número de porta utilizado.

1. Localize o nome do serviço (1ª palavra, exec neste exemplo) do servidor rexecd (7ª palavra) em /etc/inetd.conf

```
exec      stream tcp nowait OMVSKERN /usr/sbin/orexecd rexecd -LV
```
2. Localize a porta (2ª palavra, 512 neste exemplo) conectada a esse nome de serviço (1ª palavra) em /etc/services

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

O mesmo princípio aplica-se ao SSH. Sua porta comum é 22, e o nome do servidor é sshd.

Nota: Ações remotas (baseadas em host) para subprojetos do z/OS UNIX requerem que o REXEC ou o SSH esteja ativo no host. Se o REXEC/SSH não estiver configurado para utilizar a porta padrão, o cliente Developer for System z deve definir a porta correta a ser utilizada por subprojetos do z/OS UNIX. Isso pode ser feito selecionando a página de preferências **Window > Preferências... > z/OS Soluções > Subprojetos do USS > Opções de Ação Remota**.

Customizar ISPF.conf, o Arquivo de Configuração do ISPF

Essa etapa é necessária apenas ao utilizar o SCLM Developer Toolkit para o serviço Comandos do TSO (esse é o padrão). Não é necessária ao utilizar o APPC para o serviço Comandos do TSO.

O SCLM Developer Toolkit requer as definições em ISPF.conf para criar um ambiente válido para execução de serviços do ISPF. O serviço Comandos do TSO deve ser incluído para esse ambiente do ISPF.

ISPF.conf é criado durante a customização do SCLM Developer Toolkit, descrita no *SCLM Developer Toolkit Installation and Customization Guide (SC23-8504)*. O local padrão é /etc/SCLMDT/CONFIG, mas isso pode não se aplicar ao seu site.

Inclua as seguintes linhas em ISPF.conf, em que hlq é igual ao qualificador de alto nível utilizado para instalar o Developer for System z (FEK padrão).

```
*****
* Servidor de Comandos do TSO do Developer for System z -
*****
sysexec=hlq.SFEKPROC
```

O resultado deve ser semelhante à amostra em Figura 5.

```
sysproc=ISP.SISPLIB
ispmlib=ISP.SISPMENU
isptlib=ISP.SISPTENU
ispplib=ISP.SISPPENU
ispslib=ISP.SISPSLIB
ispllib=BWB.SBWBLOAD
*****
* Servidor de Comandos do TSO do Developer for System z -
*****
sysexec=FEK.SFEKPROC
```

Figura 5. ISPF.conf - Arquivo de Configuração do ISPF

Nota: Se a instrução sysexec já estiver definida, inclua o conjunto de dados hlq.SFEKPROC no fim dela, separando os nomes dos conjuntos de dados com uma vírgula (,).

Verificar a Configuração do Servidor RSE

A instalação do Developer for System z oferece diversos IVPs (Installation Verification Programs) para o servidor RSE. Os scripts do IVP estão localizados no diretório de instalação padrão `/usr/lpp/wd4z/rse/lib/`.

- fekfivpa : “Conexão do Serviço Comandos do TSO (Utilizando APPC)” na página 50
- fekfivpc : “Conexão do Serviço Comandos do TSO (Utilizando SCLMDT)” na página 49
- fekfivpd : “Conexão do Daemon RSE” na página 48
- fekfivpj : “Conexão do JES Job Monitor” na página 49j
- fekfivpr : “Conexão REXEC” na página 47
- fekfivps : “Script de Shell REXEC/SSH” na página 48

Todos os comandos de amostra nesta seção esperam que as variáveis de ambiente do RSE estejam configuradas. Dessa forma, os scripts do IVP estão disponíveis por meio da instrução `PATH` e o local de `rsed.envvars` é conhecido. Utilize os comandos **pwd** e **cd** para verificar e alterar o diretório atual para o diretório com o `rsed.envvars` customizado. O script de shell `setup.env.zseries` pode, então, ser utilizado para definir as variáveis de ambiente do RSE, como na amostra a seguir (\$ é o prompt do z/OS UNIX):

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

O script de shell `./setup.env.zseries`, que reside no mesmo diretório que `rsed.envvars`, exporta as variáveis de ambiente para que outros processos possam utilizá-las. O primeiro “.” (ponto) em `./setup.env.zseries` é um comando z/OS UNIX para executar o shell no ambiente atual, para que as variáveis de ambiente configuradas no shell sejam efetivas mesmo após a saída do shell. O segundo ponto está relacionado ao diretório atual.

Nota: Se `./setup.env.zseries` não for executado antes que os scripts `fekfivp*`, o caminho para esses scripts deve ser especificado ao chamá-los, como na amostra a seguir:

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

Nota: Alguns testes IVP utilizam a API do soquete TCP/IP REXX, que requer que a biblioteca de carregamento TCP/IP padrão `TCPIP.SEZALOAD` esteja em `LINKLIST` ou em `STEPLIB`. Os seguintes comandos podem ser necessários para que seja possível executar esses testes IVP (\$ é o prompt do z/OS UNIX):

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

ou

$ echo $STEPLIB
SOME.STEPLIB.DATASET
1$ STEPLIB=$STEPLIB:TCPIP.SEZALOAD
```

Para obter informações sobre o diagnóstico de problemas de conexão do RSE, consulte Apêndice B, “Resolvendo Problemas de Configuração”, na página 75 ou as notas técnicas sobre a Página de Suporte do Developer for System z em <http://www-306.ibm.com/software/awdtools/devzseries/support/>.

Disponibilidade de Porta

A disponibilidade da porta daemon do JES Job Monitor, REXEC, SSH e do RSE pode ser verificada emitindo o comando **netstat**. O resultado deve mostrar as portas utilizadas por esses serviços, como na amostra a seguir (\$ é o prompt do z/OS UNIX):

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

Conexão REXEC

Para verificar a conexão REXEC, execute o seguinte comando. Substitua 512 pela porta utilizada pelo REXEC e USERID por um ID de usuário válido.

```
fekfivpr 512 USERID
```

Depois de solicitar uma senha, o comando deve retornar o rastreo REXEC, um aviso de tempo limite, a versão Java e a mensagem do servidor RSE, como na amostra a seguir, (\$ é o prompt do z/OS UNIX):

```
$ fekfivpr 512 USERID
Digite a Senha:
$ EZYRC01I Chamando a função rexec_af com o seguinte:
EZYRC02I Host: CDFMVS08, user USERID, cmd cd /etc/wd4z;export RSE_USER_ID=USERI
D;./server.zseries -ivp, port 512
EZYRC19I Data socket = 4, Control socket = 6.
```

espera ver mensagens de tempo limite após um teste de IVP bem-sucedido

```
java version "1.5.0"
Java(TM) 2 Runtime Environment, Standard Edition (build pmz31dev-20070201 (SR4))
IBM J9 VM (build 2.3, J2RE 1.5.0 IBM J9 2.3 z/OS s390-31 j9vmnz3123-20070201 (JI
T ativado)
J9VM - 20070131_11312_bHdSMr
JIT - 20070109_1805ifx1_r8
GC - 200701_09)
JCL - 20070126
```

```
Servidor Iniciado com Êxito
1272
Servidor executando em: CDFMVS08
```

Nota: Se você não obtiver nenhuma saída Java e do servidor RSE, o tamanho da região do INETD será provavelmente muito pequeno (deve ser 2096128 ou maior, se iniciado a partir de uma sessão de shell do TSO/OMVS ou o tamanho da região 0 para BPXBATCH).

Nota: Você pode testar o script de shell utilizado pelo REXEC separadamente, conforme descrito no próximo teste de IVP “Script de Shell REXEC/SSH” na página 48.

Nota: O servidor é iniciado sem a tentativa de conexão de um cliente, portanto atingirá o tempo limite (depois de 5 segundos). Isso resulta em um rastreo de pilha Java (25+ linhas) semelhante à seguinte amostra:

```
$ 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

Esse teste de IVP pode ser ignorado se o teste anterior, descrito em “Conexão REXEC” na página 47, for concluído com êxito.

Verifique se o script de shell utilizado pela conexão REXEC e SSH, executando o seguinte comando:

```
fekfivps
```

O comando deve retornar um aviso de tempo limite, a versão Java e a mensagem do servidor RSE, como na amostra a seguir (\$ é o prompt do z/OS UNIX):

```

$ fekfivps
$ java version "1.5.0"

```

espera ver mensagens de tempo limite após um teste de IVP bem-sucedido

```

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 (JI
T ativado)
J9VM - 20070131_11312_bHdSMr
JIT - 20070109_1805ifx1_r8
GC - 200701_09)
JCL - 20070126

```

```

Servidor Iniciado com Êxito
1751
Servidor executando em: CDFMVS08$

```

Nota: Se você não obtiver nenhuma saída, o tamanho da região (TSO) será provavelmente muito pequeno (deve ser 2096128 ou maior).

Nota: O servidor é iniciado sem a tentativa de conexão de um cliente, portanto atngirá o tempo limite (depois de 5 segundos). Isso resulta em um rastreo de pilha Java (25+ linhas) semelhante à seguinte amostra:

```

$ 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.
...

```

Conexão do Daemon RSE

Para verificar a conexão do daemon RSE, execute o seguinte comando. Substitua 4035 pela porta utilizada pelo daemon RSE e USERID por um ID de usuário válido.

```
fekfivpd 4035 USERID
```

Depois de solicitar uma senha, o comando deve retornar uma saída como nesta amostra (\$ é o prompt do z/OS UNIX):

```

$ fekfivpd 4035 USERID
Senha:
SSL está desativado

```

```
conectado
8108
570655399
Bem-sucedido
```

Nota: Ao testar uma conexão ativada para SSL, verifique se você especificou a porta correta, se aparecer esta mensagem de erro: gsk_secure_socket_init() falhou: Soquete fechado pelo parceiro remoto

Conexão do JES Job Monitor

Para verificar a conexão do JES Job Monitor, execute o seguinte comando. Substitua 6715 pelo número da porta do JES Job Monitor.

```
fekfivpj 6715
```

O comando deve retornar a mensagem de confirmação do JES Job Monitor, como na amostra a seguir (\$ é o prompt do z/OS UNIX):

```
$ fekfivpj 6715
Aguardando resposta do JES Job Monitor...
ACKNOWLEDGE01v03
```

```
Bem-sucedido
```

Conexão do Serviço Comandos do TSO (Utilizando SCLMDT)

Este teste de IVP só é necessário se você utilizar o SCLM Developer Toolkit, para o serviço Comandos do TSO ou para o plug-in do cliente.

Verifique a conexão com o servidor de Comandos do TSO, utilizando o SCLM Developer, executando o comando a seguir.

```
fekfivpc
```

O comando deve retornar o resultado de verificações relacionadas ao SCLM Developer Toolkit (variáveis, módulos HFS, tempo de execução REXX, iniciando e parando a sessão do TSO/ISPF), como na amostra a seguir (\$ é o prompt do z/OS UNIX):

```
$ fekfivpc
```

```
-----
Verificação de instalação do host para o RSE
Verifique as mensagens de log do IVP do HOST a seguir:
-----
```

Conexão RSE e verificação apenas de inicialização da sessão TSO/ISPF base

*** VERIFICAR: VARIÁVEIS DE AMBIENTE - variável de ambiente exibidas a seguir:

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

```
STEPLIB           = BWB.SWBLOAD
```

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

```
-----
*** CHECK : HFS MODULES
```

```
Verificando o diretório de instalação: /usr/lpp/SCLMDT
```

```
Verificando os módulos BWB no diretório /bin
```

```
RC=0
```

```
MSG: BEM-SUCEDIDO
```

```
-----
```



```
*** VERIFICAR: AMBIENTE DE TEMPO DE EXECUÇÃO DO REXX
RC=0
MSG: BEM-SUCEDIDO
```

```
-----
*** VERIFICAR: INICIALIZAÇÃO DO TSO/ISPF
(A sessão do TSO/ISPF será inicializada)
RC=0
MSG: BEM-SUCEDIDO
```

```
-----
*** VERIFICAR: Encerrando a sessão IVP do TSO/ISPF
RC=0
MSG: BEM-SUCEDIDO
```

```
-----
Verificação da instalação do host concluída com êxito
-----
```

Nota: Se alguma verificação SCLMDT falhar, informações mais detalhadas serão mostradas.

fekfivpc possui muitos parâmetros opcionais e não-posicionais:

-file fekfivpc pode produzir grandes quantidades de saída (centenas de linhas). O parâmetro `-file` envia essa saída para um arquivo, `home/.eclipse/RSE/USERID/fekfivpc.log`, em que `home` é o caminho inicial definido no segmento OMVS (ou o segmento OMVS padrão, se você não tiver um segmento OMVS) e `USERID` é o ID de usuário (maiúsculas).

-plugin Por padrão, fekfivpc só verifica as funções necessárias para o serviço Comandos do TSO. O parâmetro `-plugin` inclui testes adicionais para o plug-in do cliente SCLMDT.

-debug O parâmetro `-debug` cria saída de teste detalhada. Não utilize essa opção, a menos que orientado pelo IBM Support Center.

Conexão do Serviço Comandos do TSO (Utilizando APPC)

Não execute este procedimento se não tiver configurado o APPC para o serviço Comandos do TSO.

Verifique a conexão com o servidor de Comandos do TSO (utilizando o APPC), executando o comando a seguir. Substitua `USERID` por um ID de usuário válido.

```
./fekfivpa USERID
```

Depois de solicitar uma senha, o comando deve retornar a conversação do APPC, como na amostra a seguir (\$ é o prompt do z/OS UNIX):

```
$ fekfivpa USERID
Digite a Senha:
20070607 13:57:18.584060 /usr/lpp/wd4z/rse/lib/fekfscmd: version=Oct 2003
20070607 13:57:18.584326 Input parms: 1.2.3.4 * NOTRACE USERID *****
20070607 13:57:18.585132 TP_name set via envvar: FEKFRSRV
20070607 13:57:18.586800 APPC: Allocate succeeded
20070607 13:57:18.587022 Conversation id is 0DDBD3F800000000
20070607 13:57:18.587380 APPC: Set Send Type succeeded
20070607 13:57:26.736674 APPC: Confirm succeeded
20070607 13:57:26.737027 Req to send recd value is 0
20070607 13:57:26.737546 APPC: SEND_DATA return_code = 0
20070607 13:57:26.737726 request_to_send_received = 0
20070607 13:57:26.737893 Send Data succeeded
```



```

20070607 13:57:26.738169 APPC: Set Prepare to Receive type succeeded
20070607 13:57:26.738580 APPC: Prepare to Receive succeeded
20070607 13:57:26.808899 APPC: Receive data
20070607 13:57:26.809122 RCV return_code = 0
20070607 13:57:26.809270 RCV data_received= 2
20070607 13:57:26.809415 RCV received_length= 29
20070607 13:57:26.809556 RCV status_received= 4
20070607 13:57:26.809712 RCV req_to_send= 0
20070607 13:57:26.809868 Receive succeeded
:IP: 0 9.42.112.75 1674 50246
20070607 13:57:26.810533 APPC: CONFIRMED succeeded

```

Para obter informações sobre o diagnóstico de problemas de conexão do RSE, consulte Apêndice B, “Resolvendo Problemas de Configuração”, na página 75 ou as notas técnicas sobre a Página de Suporte do Developer for System z, em <http://www-306.ibm.com/software/awdtools/devzseries/support/>.

(Opcional) Customizar `ssl.properties`, Configuração do RSE SSL

Todos os métodos de conexão do cliente do Developer for System z utilizam as variáveis do SSL configuradas no arquivo `ssl.properties`, localizado por padrão no diretório de instalação `/usr/lpp/wd4z/rse/lib/`. No entanto, `ssl.properties` é um dos arquivos que também devem ser copiados se `rse.envvars` for copiado para um diretório diferente, como `/etc/wd4z/`. O arquivo de amostra fornecido possui as instruções listadas em Figura 6, nas quais as linhas de comentário iniciam com um sinal de libras (#).

```

# 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=

```

Figura 6. `ssl.properties` – Arquivo de configuração SSL

As propriedades do daemon e do servidor precisarão ser configuradas somente se você ativar o SSL. Consulte Apêndice E, “Configurando o SSL”, na página 105 para obter mais informações sobre a configuração do SSL.

(Opcional) Customizar `rsecomm.properties`, Configuração do Rastreio RSE

Todos os métodos de conexão do cliente do Developer for System z utilizam as variáveis configuradas no arquivo `rsecomm.properties` localizado por padrão no diretório de instalação `/usr/lpp/wd4z/rse/lib/`. No entanto, `rsecomm.properties` é um dos arquivos que também devem ser copiados se `rse.envvars` for copiado para um diretório diferente, como `/etc/wd4z/`. O arquivo de amostra fornecido

possui as instruções listadas em Figura 7, nas quais as linhas de comentário iniciam com um sinal de libras (#).

```
# 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
```

Figura 7. rsecomm.properties – Arquivo de configuração da criação de log

Ao selecionar `log_location=Log_To_File` (o padrão), a criação de log é gravada em `home/.eclipse/RSE/USERID/rsecomm.log`, em que `home` é o caminho inicial definido no segmento OMVS do usuário (ou o segmento OMVS padrão se o usuário não tiver um segmento OMVS) e `USERID` é o ID de usuário de logon (maiúsculas).

Nota: A definição `debug_level` também controla o nível de criação de log de outros arquivos de log que podem ser encontrados neste diretório.

Atenção: A alteração destas configurações pode causar diminuição no desempenho e deverá ser realizada somente sob a orientação do IBM Support Center.

(Opcional) Customizar projectcfg.properties, Configuração de Projetos do Host

Os projetos do z/OS podem ser definidos individualmente por meio da perspectiva Projetos do z/OS no cliente ou podem ser definidos centralmente no host e propagados para o cliente com base no usuário. Esses "projetos baseados no host" parecem e funcionam exatamente como os projetos definidos no cliente, exceto pelo fato de suas estruturas, membros e propriedades não poderem ser modificadas pelo cliente e serem acessíveis apenas quando conectados ao host.

O local das definições do projeto é definido em `projectcfg.properties`, localizado por padrão no diretório de instalação `/usr/lpp/wd4z/rse/lib/`. No entanto, `projectcfg.properties` é um dos arquivos que também devem ser copiados se `rsd.envvars` for copiado para um diretório diferente, como `/etc/wd4z/`.

O arquivo de amostra fornecido possui as instruções listadas em Figura 8 na página 53, nas quais as linhas de comentário iniciam com um sinal de libras (#).

```
#
# Projetos baseados em host – arquivo de configuração raiz
#
# WSED-VERSION – não modifique!
WSED-VERSION=7.0.0.0
# especifique o local dos arquivos de definição do projeto baseado em host
PROJECT-HOME=/var/wd4z/projects
```

Figura 8. *projectcfg.properties* – Arquivo de configuração de projetos baseados em host

A única variável a ser alterada é PROJECT-HOME. Seu valor padrão /var/wd4z/projects é o diretório de base para as definições de projetos.

Nota: Para ativar os projetos baseados em host, um arquivo `project.instance` deve existir em /var/wd4z/projects/USERID, em que /var/wd4z/projects é o local dos arquivos de definição de projetos e USERID é o ID de usuário com o qual o desenvolvedor efetua login.

Para obter mais informações sobre projetos baseados em host, consulte o white paper *Host Based Projects in WebSphere Developer for System z version 7.0* na biblioteca da Internet do Developer for System z <http://www-306.ibm.com/software/awdtools/devzseries/library/>

(Opcional) Customizar FMEXT.properties, Integração do File Manager

O Developer for System z suporta acesso direto do cliente a um conjunto limitado de funções do IBM File Manager para z/OS. O IBM File Manager para z/OS oferece ferramentas abrangentes para trabalhar com conjuntos de dados do MVS, arquivos z/OS UNIX, dados do DB2, IMS e CICS. Essas ferramentas incluem os utilitários familiares de procura, edição, cópia e impressão localizados no ISPF, aprimoradas para atender às necessidades de desenvolvedores de aplicativos. Na versão atual do Developer for System z, apenas a procura/edição de conjuntos de dados do MVS (incluindo VSAM KSDS e ESDS) é suportada.

Observe que o produto IBM File Manager para z/OS deve ser solicitado, instalado e configurado separadamente. Consulte *IBM Rational Developer para System z Host Planning Guide (GI11-8296-00)* para saber qual nível do File Manager é necessário para sua versão do Developer for System z. A instalação e customização deste produto não está descrita neste manual.

As definições do File Manager necessárias pelo Developer for System z são armazenadas em `FMEXT.properties`, localizado por padrão no diretório de instalação /usr/lpp/wd4z/rse/lib/. No entanto, `FMEXT.properties` é um dos arquivos que também devem ser copiados se `rse.envvars` for copiado para um diretório diferente, como /etc/wd4z/.

O arquivo de amostra fornecido possui as instruções listadas em Figura 9 na página 54, nas quais as linhas de comentário iniciam com um sinal de libras (#).

```
# Propriedades de Extensão do FMI (File Manager Integration)
#
startup.script=/usr/lpp/wd4z/rse/lib/fmiSub
startup.port=1957
startup.range=100
startup.fmload=FMN.SFMNMOD1
startup.jobcard1=//JOB CARD JOB <parâmetros da tarefa>
startup.jobcard2=//*
startup.jobcard3=//*
startup.sysout=*
```

Figura 9. FMIEXT.properties – Arquivo de configuração do File Manager

startup.script

Local absoluto de fmiSub, o script de inicialização do servidor FMI. O valor padrão é /usr/lpp/wd4z/rse/lib/fmiSub.

startup.port

A primeira porta utilizada para comunicação entre o servidor FMI e o servidor RSE, que retransmite as informações para o cliente. A porta padrão é 1957. A comunicação nesta porta está confinada para a máquina host.

Nota: Antes de selecionar uma porta, verifique se ela está disponível em seu sistema com os comandos **NETSTAT** e **NETSTAT PORTL**. Consulte “Portas TCP/IP Reservadas” na página 81 para obter informações adicionais.

startup.range

Intervalo de portas, iniciando em startup.port, que será utilizado para comunicação do servidor FMI. O padrão é 100. Por exemplo, ao utilizar os padrões, as portas de 1957 até 2056 (inclusive) podem ser utilizadas pelo servidor FMI.

startup.fmload

Local absoluto da biblioteca de carregamento do File Manager. O valor padrão é FMN.SFMNMOD1. Não utilize aspas (') para tornar o nome do conjunto de dados absoluto; um prefixo não é incluído.

startup.jobcard1

startup.jobcard2

startup.jobcard3

Informações da placa de tarefa do servidor FMI. Os valores padrão são //JOB CARD JOB <parâmetros da tarefa>, //* e //. O nome da tarefa será substituído por FEK<port> para garantir a exclusividade.

startup.sysout

Classe sysout do servidor FMI. O valor padrão é *.

Capítulo 5. (Opcional) Ativando o IBM CARMA

O CARMA (FMID: HCMA710) é um auxílio de produtividade para desenvolvedores criando APIs para SCMs (Software Configuration Managers). Por sua vez, essas APIs podem ser utilizadas por aplicativos (por exemplo, Developer for System z) para acessar os SCMs.

Antes de instalar a versão 7.1, se você for um usuário anterior do CARMA, é recomendável salvar a customização relacionada, conforme descrito em “Fazendo Backup de Arquivos Configurados Anteriormente” na página 11.

Após a instalação, você deve configurar o CARMA seguindo estas etapas:

1. Configure o servidor CARMA no host z/OS (requer ações no MVS e no z/OS UNIX)
2. (Opcional) Configure os RAMs de amostra
3. (Opcional) Restrinja o acesso aos arquivos de inicialização e clusters do VSAM. Na maioria das circunstâncias, somente os Administradores de Sistema e os desenvolvedores RAM do CARMA precisarão gravar nesses arquivos, enquanto outros usuários exigirão apenas acesso de leitura.

Nota: Os RAMs são APIs gravadas pelo usuário para fazer interface com os SCMs (Software Configuration Managers) do z/OS.

Consulte Tabela 1 na página 1 para obter uma lista de manuais que oferecem mais informações sobre o CARMA e como utilizá-lo.

O usuário pode controlar a quantidade de informações de rastreamento que o CARMA gera, configurando o Nível de Rastreamento na guia de propriedades da conexão CARMA no cliente. As opções para o Nível de Rastreamento são:

- Desativar Criação de Log
- Log de Erros
- Log de Avisos
- Log Informativo
- Log de Depuração

O valor padrão é

Log de Erros

. Consulte “Localização dos Arquivos de Log” na página 75 para obter informações adicionais sobre as localizações do arquivo de log.

Customizando os Componentes MVS do CARMA

Todas as referências a hlq nesta seção referem-se ao qualificador de alto nível utilizado durante a instalação do CARMA. O padrão da instalação é CRA, mas isto pode não se aplicar ao seu site.

Siga estas etapas para configurar o host MVS:

1. Copie os membros a serem customizados do diretório de instalação para uma biblioteca pessoal e customize estas cópias para evitar que sejam sobrescritas ao realizar manutenção.
 - hlq.SCRACLST(CRASUBMT)
 - hlq.SCRASAMP(CRA\$VDEF)
 - hlq.SCRASAMP(CRA\$VMSG)
 - hlq.SCRASAMP(CRA\$VSTR)
2. Customize a CLIST hlq.SCRACLST(CRASUBMT). Consulte a documentação em CRASUBMT para obter instruções de customização. A CLIST CRASUBMT envia um servidor CARMA.

Nota: Opcionalmente, você pode alterar o valor de tempo limite do CARMA modificando a linha PROC 1 PORT TIMEOUT(420) na CLIST hlq.SCRACLST(CRASUBMT). O valor de tempo limite é o número de segundos que o CARMA aguardará para o próximo comando do cliente. A configuração de um valor 0 resultará no valor de tempo limite padrão, atualmente, 420 segundos (7 minutos).

3. Customize e envie o JCL hlq.SCRASAMP(CRA\$VDEF). Consulte a documentação em CRA\$VDEF para obter instruções de customização.

Nota: Você pode utilizar o JCL CRA\$VDEF para atualizar o cluster VSAM CRADEF (configuração CARMA) posteriormente. Para atualizar o cluster, aponte a instrução INPUT DD para o conjunto de dados sequenciais escolhido em vez de CRAINIT. Consulte o *Common Access Repository Manager Developer's Guide (SC31-6914)* para obter mais informações sobre a definição deste conjunto de dados sequencial.

4. Customize e envie o JCL hlq.SCRASAMP(CRA\$VMSG). Consulte a documentação em CRA\$VMSG para obter instruções de customização. CRA\$VMSG cria e carrega o VSAM de mensagens CARMA CRAMSG.
5. Customize e envie o JCL hlq.SCRASAMP(CRA\$VSTR). Consulte a documentação em CRA\$VSTR para obter instruções de customização.

Nota: Você pode utilizar o JCL CRA\$VSTR para atualizar o cluster VSAM CRASTRS (informações customizadas CARMA) posteriormente. Para atualizar o cluster, aponte a instrução DD para seu conjunto de dados sequenciais escolhido em vez de CRASINIT. Consulte o *Common Access Repository Manager Developer's Guide (SC31-6914)* para obter mais informações sobre a definição deste conjunto de dados sequencial.

Customizando os Componentes do z/OS UNIX para o CARMA

Se não estiver familiarizado com o z/OS UNIX, é aconselhável solicitar assistência de um administrador z/OS UNIX ou outro administrador UNIX experiente para desempenhar as tarefas listadas nesta seção.

Os comandos z/OS UNIX necessários para desempenhar as tarefas listadas são brevemente descritos para sua conveniência. A menos que observado de outra forma, consulte *UNIX System Services Command Reference (SA22-7802)* para obter informações adicionais sobre estes comandos.

- As tarefas descritas abaixo esperam que você esteja ativo no z/OS UNIX. Isso pode ser feito emitindo o comando do TSO OMVS. Utilize o comando **exit** para retornar ao TSO.

- O MVS fornece a possibilidade de editar arquivos z/OS UNIX utilizando o ISPF por meio do comando **OEDIT**. Este comando pode ser utilizado no TSO e no OMVS.

Todas as instruções de caminho `/usr/lpp/wd4z/` neste capítulo se referem ao caminho utilizado durante a instalação do Developer for System z, não ao CARMA. O padrão é `/usr/lpp/wd4z/`, mas isto pode não se aplicar ao seu site.

Siga estas etapas para configurar os componentes do z/OS UNIX CARMA, que são instalados durante a instalação do IBM Rational Developer para System z (FMID: HHOP710):

1. O arquivo de configuração `CRASRV.properties` deve residir no mesmo diretório como o arquivo `rse.envvars` customizado. Ambos os arquivos residem, por padrão, no diretório de instalação (padrão path `/usr/lpp/wd4z/rse/lib/`). Conforme descrito em “Salvando o Arquivo de Configuração `rse.envvars` em um Outro Diretório” na página 34, é aconselhável copiar para outro diretório, para evitar sobrescrever ao aplicar a manutenção. Nas amostras utilizadas neste manual, esse diretório é `/etc/wd4z/`.

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

2. O arquivo de configuração `sample CRASRV.properties` consiste em um conjunto de definições de variáveis de ambiente. O arquivo de configuração de amostra deve ser alterado para corresponder aos padrões do site e contém as instruções listadas em Figura 10, nas quais as linhas de comentário iniciam com um sinal de libras (#).

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

Figura 10. CRASRV.properties – Arquivo de configuração CARMA

port.start

Primeira porta utilizada para comunicação entre os componentes MVS e z/OS UNIX. A porta padrão é 5227. A comunicação nesta porta está confinada para a máquina host.

Nota: Antes de selecionar uma porta, verifique se ela está disponível em seu sistema com os comandos **NETSTAT** e **NETSTAT PORTL**. Consulte “Portas TCP/IP Reservadas” na página 81 para obter informações adicionais.

port.range

Intervalo de portas, iniciando em `port.start`, que será utilizado para comunicação do Servidor CARMA. O padrão é 100. Por exemplo, ao utilizar os padrões, as portas de 5227 até 5326 (inclusive) podem ser utilizadas pelo CARMA.

startup.script.name

Define o caminho absoluto do script de envio REXX `rexxsub`. O padrão é `/usr/lpp/wd4z/rse/lib/rexxsub`. Este executável REXX acionará a execução da CLIST `CRASUBMT` no MVS.

clist.dsname

Define o local da CLIST `CRASUBMT`, utilizando convenções de referência

do MVS. Com apóstrofos (') é um local absoluto, sem o prefixo do usuário antecedendo o nome do conjunto de dados fornecido. O padrão é 'hlq.SCRACLST(CRASUBMT)'. A instalação CARMA SMP/E, que cria CRASUBMT, utiliza CRA como valor padrão para hlq. Esta CLIST iniciará um servidor CARMA ao abrir uma conexão.

Nota: No Developer for System z versão 7.0, o nome do conjunto de dados e do membro da CLIST foi movido de rexxsub (variável DSNNAME) em CRASRV.properties, eliminando a necessidade de customizar rexxsub. Deixe rexxsub no diretório de instalação se quiser que a possível manutenção do SMP/E seja ativada automaticamente.

(Opcional) Ativando os RAM (Repository Access Managers) de Amostra

Os RAMs (Repository Access Managers) são APIs escritas pelo usuário para fazer interface com os z/OS SCMs (Software Configuration Managers). Siga as instruções nas seções abaixo para as RAMs de amostra que você deseja ativar.

Nota: As RAMs de amostra são fornecidas com o propósito de testar a configuração do ambiente CARMA e como exemplos para que você desenvolva suas próprias RAMs. NÃO utilize os RAMs de amostra fornecidos em um ambiente de produção.

Nota: Todas as referências a hlq nesta seção referem-se ao qualificador de alto nível utilizado durante a instalação do CARMA. O padrão da instalação é CRA, mas isto pode não se aplicar ao seu site.

Consulte o *IBM Rational Developer para System z Common Access Repository Manager Developer's Guide (SC23-7660-00)* para obter mais informações sobre RAMs de amostra e sobre o código-fonte de amostra fornecido.

Ativando o RAM do SCLM

1. Copie os membros a serem customizados do diretório de instalação para uma biblioteca pessoal e customize estas cópias para evitar que sejam sobrescritas ao realizar manutenção.
 - hlq.SCRASAMP(CRA#VSLM)
 - hlq.SCRASAMP(CRA#ASLM)
2. Customize e envie o JCL hlq.SCRASAMP(CRA#VSLM). Consulte a documentação em CRA#VSLM para obter instruções de customização. CRA#VSLM cria e carrega o VSAM de mensagens SCLM RAM.
3. Customize o JCL hlq.SCRASAMP(CRA#ASLM). Consulte a documentação em CRA#ASLM para obter instruções de customização. CRA#ASLM aloca os conjuntos de dados necessários pelos clientes SCLM RAM.

Nota: Cada usuário deve enviar o CRA#ASLM uma vez antes de utilizar o CARMA com o SCLM RAM. Se isso não for feito, o resultado será um erro de alocação.

Ativando o PDS RAM

1. Copie os membros a serem customizados do diretório de instalação para uma biblioteca pessoal e customize estas cópias para evitar que sejam sobrescritas ao realizar manutenção.

- hlq.SCRASAMP(CRA#VPDS)
2. Customize e envie o JCL hlq.SCRASAMP(CRA#VPDS). Consulte a documentação em CRA#VPDS para obter instruções de customização. CRA#VPDS cria e carrega o VSAM de mensagens PDS RAM.

Capítulo 6. (Opcional) Ativando o IBM Software Configuration and Library Manager (SCLM) Developer Toolkit

O IBM SCLM (Software Configuration and Library Manager) (FMID: HSD3310) Developer Toolkit fornece as ferramentas necessárias para estender as capacidades do SCLM para o cliente. O SCLM por si só é um gerenciador de código-fonte com base no host enviado como parte do ISPF.

O SCLM Developer Toolkit, fornecido junto com o Developer for System z, é um plug-in com base no Eclipse que faz a interface com o SCLM e fornece acesso a todos os processos SCLM para o desenvolvimento de códigos antigos, assim como o suporte para implementação completa do Java e do J2EE na estação de trabalho com sincronização com o SCLM no mainframe, incluindo construção, montagem e implementação do código J2EE do mainframe.

Consulte Tabela 1 na página 1 para obter uma lista de manuais que oferecem mais informações sobre o SCLM Developers Toolkit e como utilizá-lo.

Capítulo 7. Considerações do Cliente do Developer for System z

Os usuários do cliente do Developer for System z devem conhecer o resultado de determinadas customizações do host, como os números de portas TCP/IP, para que o cliente funcione corretamente. Utilize a lista de verificação na Tabela 10 para coletar as informações necessárias.

Tabela 10. Lista de verificação do cliente do Developer for System z

Customização	Valor
Número da porta do servidor do JES Job Monitor (padrão 6715): Consulte SERV_PORT em “Customizar o FEJJCNFG, o Arquivo de Configuração do JES Job Monitor” na página 16.	
Local dos procedimentos de ELAXF* se não estiverem em uma biblioteca de procedimentos do sistema: Consulte a nota sobre JCLLIB em “Customizar ELAXF*, Procedimentos de Construção Remota” na página 21.	
Nomes de procedimentos e/ou de etapas de procedimentos de ELAXF*, se tiverem sido alterados: Consulte a nota sobre a alteração deles em “Customizar ELAXF*, Procedimentos de Construção Remota” na página 21.	
Nome do procedimento armazenado do DB2 (padrão ELAXMSAM): Consulte informações sobre procedimentos armazenados do DB2 no Apêndice A, “Executando Várias Instâncias do Developer for System z”, na página 71.	
Local do procedimento armazenado do DB2, se não estiver em uma biblioteca de procedimentos do sistema: Consulte o “(Opcional) Customizar ELAXM*, Membros do Procedimento Armazenado do DB2” na página 26.	
Utilize o método de conexão DAEMON, REXEC ou SSH para RSE: Consulte “Configuração do Daemon INETD e REXEC/SSH do RSE” na página 42.	
Número da porta TCP/IP do daemon RSE (padrão 4035): Consulte o “Configuração do Daemon INETD RSE” na página 43.	
Caminho para o script de shell server.zseries para REXEC/SSH (padrão /usr/lpp/wd4z/rse/lib, recomendável /etc/wd4z): Consulte o “Configuração do INETD REXEC (ou SSH)” na página 44.	
Número da porta REXEC ou SSH (padrão 512 ou 22, respectivamente): Consulte o “Configuração do INETD REXEC (ou SSH)” na página 44. Nota: Ações remotas (baseadas em host) para subprojetos do z/OS UNIX requerem que o REXEC ou o SSH esteja ativo no host.	

Tabela 10. Lista de verificação do cliente do Developer for System z (continuação)

Customização	Valor
<p>Local do JCL CRA#ASLM para alocações do conjunto de dados CARMA SCLM RAM:</p> <p>Consulte a nota sobre CRA#ASLM em “Ativando o RAM do SCLM” na página 58.</p>	

Capítulo 8. Considerações de Desempenho

O z/OS é um sistema operacional altamente customizável, e (algumas vezes pequenas) alterações no sistema podem ter um grande impacto sobre o desempenho geral. Este capítulo destaca algumas alterações que podem ser feitas para aprimorar o desempenho do Developer for System z.

Consulte o *MVS Initialization and Tuning Guide (SA22-7591)* e *UNIX System Services Planning (GA22-7800)* para obter mais informações sobre o ajuste do sistema.

Evite o Uso de STEPLIB

Cada processo do z/OS UNIX que possui um STEPLIB que é propagado do pai para o filho ou em um executável consumirá aproximadamente 200 bytes de ECSA (Extended Common Storage Area). Se nenhuma variável de ambiente STEPLIB estiver definida ou quando for definida como STEPLIB=CURRENT, o z/OS UNIX propaga todas as alocações TASKLIB, STEPLIB e JOBLIB ativas atualmente durante uma função fork(), spawn() ou exec(). O servidor RSE inicia diversos processos, e cada conexão do cliente possui um servidor RSE privado. Por isso, os números podem aumentar rapidamente.

O Developer for System z possui um padrão de STEPLIB=NONE codificado em rsed.envvars, conforme descrito em “Customizar rsed.envvars, o Arquivo de Configuração para RSE” na página 35. Pelos motivos mencionados acima, é aconselhável não alterar essa diretiva e colocar os conjuntos de dados de destino em LINKLIST ou LPA (Link Pack Area).

Se você utilizar a diretiva STEPLIB, deve verificar o conteúdo de rsed.envvars para saber se a instrução STEPLIB é a primeira ou não.

- Se a última diretiva STEPLIB definida anteriormente em rsed.envvars for igual a
STEPLIB=NONE
STEPLIB=first.steplib.dataset:second.steplib.dataset
- Se a última diretiva STEPLIB definida anteriormente em rsed.envvars não for igual a
STEPLIB=NONE
STEPLIB=\$STEPLIB:first.steplib.dataset:second.steplib.dataset

Aprimorar o Acesso às Bibliotecas do Sistema

Determinadas bibliotecas do sistema e módulos de carregamento são intensamente utilizadas pelo z/OS UNIX e pelas atividades de desenvolvimento do aplicativo. O aprimoramento do acesso, como a inclusão na LPA, pode aprimorar o desempenho do sistema. Consulte a *MVS Initialization and Tuning Reference (SC28-1752)* para obter mais informações sobre a alteração de membros SYS1.PARMLIB descritos a seguir.

Bibliotecas de Tempo de Execução do LE (Language Environment)

Quando programas C (incluindo o shell do z/OS UNIX) são executados, freqüentemente utilizam rotinas da biblioteca de tempo de execução do LE

(Language Environment). Em média, aproximadamente 4 MB da biblioteca de tempo de execução são carregados na memória para cada espaço de endereço executando um programa ativado para LE e copiados em cada fork.

O conjunto de dados CEE.SCEELPA contém um subconjunto das rotinas de tempo de execução do LE, intensamente utilizadas pelo z/OS UNIX. É aconselhável incluir esse desempenho no SYS1.PARMLIB(LPALSTxx) para ganho máximo de desempenho. Fazendo isso, os módulos são lidos do disco apenas uma vez e são armazenados em um local compartilhado.

Nota: Inclua a seguinte instrução em SYS1.PARMLIB(PROGxx), se você preferir incluir os módulos de carregamento na LPA dinâmica:

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

Também é aconselhável colocar as bibliotecas de tempo de execução do LE CEE.SCEERUN e CEE.SCEERUN2 em LINKLIST, incluindo os conjuntos de dados em SYS1.PARMLIB(LNKLSTxx) ou SYS1.PARMLIB(PROGxx). Isso elimina o código extra STEPLIB do z/OS UNIX e há uma redução de entrada/saída devido ao gerenciamento pelo LLA e VLF, ou produtos semelhantes.

Nota: Inclua a biblioteca de classes C/C++ DLL CBC.SCLBDLL também em LINKLIST pelos mesmos motivos.

Se você decidir não colocar essas bibliotecas em LINKLIST, então deve definir a instrução STEPLIB apropriada em rsed.envvars, conforme descrito em “Customizar rsed.envvars, o Arquivo de Configuração para RSE” na página 35. Embora esse método sempre utilize armazenamento virtual adicional, pode aprimorar o desempenho definindo as bibliotecas de tempo de execução do LE para o LLA ou um produto semelhante. Isso reduz a E/S necessária para carregar os módulos.

Desenvolvimento do Aplicativo

Em sistemas em que o desenvolvimento do aplicativo seja a atividade principal, o desempenho também pode ser aprimorado se você colocar o editor de vínculo na LPA dinâmica, incluindo essas linhas em SYS1.PARMLIB(PROGxx):

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

Para desenvolvimento do C/C++, você também pode incluir o conjunto de dados do compilador CBC.SCCNCMP em SYS1.PARMLIB(LPALSTxx).

As instruções acima são amostras de possíveis candidatas de LPA, mas as necessidades no seu site podem variar. Consulte a *Language Environment Customization (SA22-7564)* para obter informações sobre como colocar outros módulos de carregamento do LE na LPA dinâmica. Consulte *UNIX System Services Planning (GA22-7800)* para obter mais informações sobre a colocação de módulos de carregamento do compilador C/C++ na LPA dinâmica.

Aprimorando o Desempenho da Verificação de Segurança

Para aprimorar o desempenho da verificação de segurança realizada para o z/OS UNIX, defina o perfil BPX.SAFFASTPATH na classe FACILITY do software de segurança. Isso reduz o código extra ao realizar verificações de segurança do z/OS UNIX para uma ampla variedade de operações, incluindo verificação de acesso ao

arquivo, verificação de acesso ao IPC e verificação de propriedade do processo. Consulte *UNIX System Services Planning (GA22-7800)* para obter mais informações sobre esse perfil.

Nota: Os usuários não precisam ter permissão para o perfil BPX.SAFFASTPATH.

Compartilhamento de Classe entre JVMs

A JVM Java Virtual Machine IBM versão 5 e posterior permite compartilhar a auto-inicialização e classes de aplicativo entre JVMs, armazenando-as em um cache na memória compartilhada. O compartilhamento de classe reduz o consumo geral de memória virtual quando mais de uma JVM compartilha um cache. O compartilhamento de classe também reduz o tempo de inicialização para uma JVM após a criação do cache.

O cache da classe compartilhada é independente de qualquer JVM ativa e persiste além da vida útil da JVM que criou o cache. Como o cache da classe compartilhada persiste além da vida útil da JVM, o cache é atualizado dinamicamente para refletir as modificações que podem ter sido feitas nas JARs ou classes no sistema de arquivo.

O código extra para criar e preencher um novo cache é mínimo. O custo de inicialização da JVM em tempo para uma única JVM geralmente é entre 0 e 5% menor, em comparação com um sistema não utilizando o compartilhamento de classe, dependendo de quantas classes são carregadas. O aprimoramento do tempo de inicialização da JVM com um cache preenchido geralmente é entre 10% e 40% mais rápido, em comparação com um sistema não utilizando o compartilhamento de classe, dependendo do sistema operacional e do número de classes carregadas. Várias JVMs em execução simultânea mostrarão benefícios gerais de tempo de inicialização maiores.

Consulte o *SDK and Runtime Environment User Guide* para aprender mais sobre o compartilhamento de classe.

Ativar Compartilhamento de Classe

Para permitir o compartilhamento de classe para o servidor RSE, remova o comentário da diretiva a seguir em `rse.envvars`, conforme descrito em “(Opcional) Definindo Parâmetros Adicionais de Inicialização Java com `_RSE_*OPTS`” na página 41. A primeira instrução define um cache chamado RSE com acesso ao grupo e permite que o servidor RSE seja iniciado, mesmo se o compartilhamento de classe falhar. A segunda instrução é opcional e define o tamanho do cache como 6 megabytes (o padrão do sistema é 16 MB).

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

Nota: Conforme mencionado em “Segurança do Cache” na página 68, todos os usuários utilizando a classe compartilhada devem ter o mesmo GID (ID do grupo) primário. Isso significa que os usuários devem ter o mesmo grupo padrão definido no software de segurança ou que os grupos padrão diferentes tenham o mesmo GID no segmento OMVS.

Limites de Tamanho do Cache

O tamanho máximo do cache compartilhado teórico é 2 GB. O tamanho do cache que você pode especificar é limitado pela quantidade de memória física e espaço de troca disponível ao sistema. Como o espaço de endereço virtual de um processo

é compartilhado entre o cache da classe compartilhada e o heap Java, o aumento do tamanho máximo do heap Java reduzirá o tamanho do cache da classe compartilhada que você pode criar.

Segurança do Cache

O acesso ao cache da classe compartilhada é limitado pelas permissões do sistema operacional e pelas permissões de segurança Java.

Por padrão, os caches de classe são criados com segurança em nível de usuário, portanto só o usuário que criou o cache pode acessá-lo. No z/OS UNIX, há uma opção, `groupAccess`, que oferece acesso a todos os usuários no grupo principal do usuário que criou o cache. No entanto, independente do nível de acesso utilizado, um cache só pode ser destruído pelo usuário que o criou ou por um usuário root (UID 0).

Consulte o *SDK and Runtime Environment User Guide* para aprender mais sobre opções adicionais de segurança ao utilizar um Java SecurityManager.

SYS1.PARMLIB(BPXPRMxx)

Algumas configurações de SYS1.PARMLIB(BPXPRMxx) afetam o desempenho de classes compartilhadas. A utilização das configurações incorretas pode impedir o funcionamento das classes compartilhadas. Essas configurações também podem ter implicações de desempenho. Para obter mais informações sobre implicações do desempenho e o uso desses parâmetros, consulte *MVS Initialization and Tuning Reference (SC28-1752)* e *UNIX System Services Planning (GA22-7800)*. Os parâmetros BPXPRMxx mais significantes que afetam a operação de classes compartilhadas são:

- MAXSHAREPAGES, IPCSHMPAGES, IPCSHMMPAGES e IPCSHMNSEGS

Essas configurações afetam a quantidade de páginas da memória compartilhada disponíveis para a JVM. O tamanho da página compartilhada para um serviço do sistema z/OS UNIX é fixado em 4 KB. As classes compartilhadas tentam criar um cache de 16 MB, por padrão. Portanto, defina IPCSHMMPAGES como mais de 4096.

Se você definir um tamanho de cache utilizando `-Xscmx`, a JVM arredondará o valor para o megabyte mais próximo. Você deve levar isso em consideração ao definir IPCSHMMPAGES no sistema.

- IPCSEMIDS e IPCSEMSEMS

Essas configurações afetam a quantidade de semáforos disponíveis aos processos UNIX. As classes compartilhadas utilizam semáforos IPC para comunicação entre as JVMs.

Espaço em Disco

O cache da classe compartilhada requer espaço em disco para armazenar informações de identificação sobre os caches existentes no sistema. Essas informações são armazenadas em `/tmp/javasharedresources`. Se o diretório de informações de identificação for excluído, a JVM não poderá identificar as classes compartilhadas no sistema e deverá recriar o cache.

Utilitários de Gerenciamento do Cache

O comando da linha Java `-Xshareclasses` pode utilizar diversas opções, sendo alguns utilitários de gerenciamento do cache. Três deles são mostrados na amostra a seguir (\$ é o prompt do z/OS UNIX). Consulte o *SDK and Runtime Environment User Guide* para obter uma visão geral completa das opções de linha de comandos suportadas.

```
$ java -Xshareclasses:listAllCaches
Cache Compartilhado    OS shmid    em uso    Hora da última desconexão
RSE                    401412      0         Seg 18 Jun 2007 17h23min16s
```

Não foi possível criar a Java virtual machine.

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

Estatísticas atuais para o cache "RSE":

```
endereço base          = 0x0F300058
endereço final         = 0x0F8FFFF8
ponteiro de alocação   = 0x0F4D2E28
```

```
tamanho do cache      = 6291368
bytes livres          = 4355696
bytes ROMClass        = 1912272
bytes de metadados    = 23400
% de metadados utilizados = 1%
```

```
# ROMClasses          = 475
# Caminhos de classe  = 4
# URLs                = 0
# Tokens              = 0
# Classes antigas     = 0
% Classes antigas     = 0%
```

O cache está 30% cheio

Não foi possível criar a Java virtual machine.

```
$ java -Xshareclasses:name=RSE,destroy
JVMSHRC010I O Cache Compartilhado "RSE" foi destruído
Não foi possível criar a Java virtual machine.
```

Nota: Os utilitários de cache desempenham a operação necessária no cache especificado sem iniciar a JVM, portanto a mensagem “Não foi possível criar a Java virtual machine.” é normal.

Nota: Um cache só pode ser destruído se todas as JVMs que o utilizam estiverem encerradas e se o usuário tiver permissões suficientes.

Tamanho de Heap Java Fixo

Com um heap de tamanho fixo, nenhuma expansão ou contração de heap ocorre e isso pode ocasionar significantes ganhos de desempenho em algumas situações. No entanto, a utilização de um heap de tamanho fixo geralmente não é uma boa idéia, porque atrasa o início de uma coleta de lixo até que o heap esteja cheio, momento em que será uma tarefa principal. Também aumenta o risco de fragmentação, o que requer uma compactação de heap. Portanto, utilize heaps de tamanho fixo só depois de desempenhar teste adequado ou quando orientado pelo IBM Support Center. Consulte o *Diagnostics Guide* (SC34-6358) para obter mais informações sobre tamanhos de heap e coleta de lixo.

Por padrão, o tamanho de heap inicial de uma JVM (Java Virtual Machine) do z/OS é 1 megabyte. O tamanho máximo é de 64 megabytes. Os limites podem ser definidos com as opções da linha de comandos `-Xms` (inicial) e `-Xmx` (máximo) Java.

No Developer for System z, as opções da linha de comandos Java são definidos na diretiva `_RSE_JAVAOPTS` de `rsed.envvars`, conforme descrito em “(Opcional) Definindo Parâmetros Adicionais de Inicialização Java com `_RSE_*OPTS`” na página 41.

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

Gerenciamento de Carga de Trabalho

Cada site possui necessidades específicas e é possível customizar o sistema operacional z/OS para aproveitar ao máximo os recursos disponíveis de acordo com essas necessidades. Com gerenciamento de carga de trabalho, você define suas metas de desempenho e designa uma importância de negócios a cada meta. Você define as metas para o trabalho em termos de negócios e o sistema decide quantos recursos, como CPU e armazenamento, devem ser fornecidos para o trabalho, de acordo com a meta.

O desempenho do Developer for System z pode ser equilibrado pela configuração das metas corretas para os processos. Algumas orientações gerais estão listadas a seguir.

- Designar a transação APPC a um grupo de desempenho do TSO
- Designe o servidor RSE a um grupo de desempenho do TSO, ou um imediatamente abaixo dele, se tiver uso do TSO significativo

Consulte *MVS Planning: Workload Management (SA22-7602)* para obter mais informações sobre isso.

Apêndice A. Executando Várias Instâncias do Developer for System z

Há situações em que você deseja várias instâncias do Developer for System z ativas no mesmo sistema, por exemplo, durante o teste de um upgrade. No entanto, alguns recursos como portas TCP/IP não podem ser compartilhadas, portanto os padrões nem sempre são aplicáveis. Utilize as informações neste apêndice para planejar a coexistência de instâncias diferentes do Developer for System z, após as quais você pode utilizar este guia de configuração para customizá-las.

Embora seja possível compartilhar determinadas partes do Developer for System z entre 2 (ou mais) instâncias, é aconselhável NÃO fazê-lo, a menos que seus níveis de software sejam idênticos e as únicas alterações sejam nos membros da configuração. O Developer for System z deixa espaço de customização suficiente para criar várias instâncias que não se sobrepõem, e recomendamos a utilização destes recursos.

Arquivos de Configuração Diferentes de Níveis de Software Idênticos

Em algum conjunto limitado de circunstâncias, é possível compartilhar tudo, menos (algumas das) as partes customizáveis. Um exemplo é fornecer acesso não-SSL para uso no site e comunicação codificada por SSL para uso externo.

Atenção: O grupo compartilhado NÃO pode ser utilizado com segurança para testar a manutenção, uma visualização técnica ou um novo release.

Para configurar outra instância de uma instalação ativa do Developer for System z, refaça as etapas de customização para as partes que são diferentes, utilizando conjuntos de dados/diretórios/portas diferentes para evitar sobrepor a configuração atual.

Na amostra SSL mencionada acima, as customizações atuais do servidor RSE podem ser clonadas, e depois as `ssl.properties` clonadas podem ser atualizadas. As customizações MVS (JES Job Monitor, etc.) podem ser compartilhadas entre as instâncias SSL e não-SSL. Isso resultaria nas seguintes ações:

1. Criar um novo diretório para reter os membros de configuração customizada SSL
2. Copiar os membros de configuração ativa para este diretório
3. Atualizar as `ssl.properties` copiadas para fornecer as informações relacionadas ao SSL
4. Definir um novo número de porta do daemon RSE em `/etc/services`
5. Definir um novo processo do daemon RSE em `/etc/inetd.conf`, utilizando o novo diretório para o parâmetro `-d`
6. Reiniciar o INETD para ativar o novo daemon RSE

Todas as Outras Situações

Quando alterações de código estiverem envolvidas (manutenção, visualizações técnicas, novo release) ou quando as alterações forem razoavelmente complexas, é aconselhável fazer outra instalação do Developer for System z. Esta seção descreve os possíveis pontos de conflito entre as diferentes instalações.

A lista a seguir é uma breve visão geral dos itens que devem ser ou são altamente aconselhados a serem diferentes entre as instâncias do Developer for System z:

- Bibliotecas de instalação
- Porta TCP/IP do JES Job Monitor e seu arquivo de configuração FEJJC�FG
- JCL de inicialização do JES Job Monitor
- Nome da transação APPC
- Arquivo de configuração RSE, rsed.envvars
- Porta TCP/IP do daemon RSE

Uma visão geral mais detalhada é listada abaixo.

- Bibliotecas de instalação
 1. Instale cada instância do Developer for System z em conjuntos de dados e diretórios separados. Tenha em mente que você só pode alterar o caminho z/OS UNIX prefixando o padrão IBM fornecido de /usr/lpp/wd4z. Uma amostra válida seria /service/usr/lpp/wd4z.
- Partes obrigatórias
 1. O arquivo de configuração do JES Job Monitor hlq.SFEKSAMP(FEJJC�FG) contém o número da porta TCP/IP do JES Job Monitor e não pode ser compartilhado. O membro pode ser renomeado (se o JCL também for atualizado), portanto você pode colocar todas as versões customizadas deste membro no mesmo conjunto de dados se não estiver realizando as atualizações no conjunto de dados da instalação.
 2. O JCL de inicialização do JES Job Monitor hlq.SFEKSAMP(FEJJJCL) se refere ao FEJJC�FG e, portanto, também não pode ser compartilhado. Após renomear o membro (e a placa JOB) você pode colocar todas as JCLs no mesmo conjunto de dados.
 3. A transação APPC possui uma referência para hlq.SFEKPROC(FEKFRSRV), o servidor de comandos do TSO. Isto é específico ao nível do software, portanto você deve criar uma transação APPC por instância.
 4. O arquivo de configuração RSE /usr/lpp/wd4z/rse/lib/rsed.envvars contém referências para o caminho de instalação e o nome da transação APPC, o qual necessita ser exclusivo. O nome do arquivo é obrigatório, portanto você não pode manter as cópias diferentes no mesmo diretório. Tenha em mente que, uma vez que o nome da transação APPC for alterado, a variável _FEKFSCMD_TP_NAME_ deve ser definida em rsed.envvars.
 5. Todos os arquivos de configuração e scripts de shell obrigatórios que devem residir no mesmo diretório que rsed.envvars não podem ser compartilhados, pois rsed.envvars deve estar em um local não-compartilhado.
 6. As portas TCP/IP do REXEC e do SSH podem ser compartilhadas sem qualquer restrição.
 7. O daemon RSE não pode ser compartilhado uma vez que cada daemon RSE possui sua própria porta TCP/IP. Um novo daemon, com um nome de serviço e porta diferentes, deve ser criado.
- Partes opcionais

1. Alguns procedimentos ELAXF* possuem uma referência a hlq.SFEKLOAD, a biblioteca de carregamento do Developer for System z. Consulte a nota na JCLLIB em “Customizar ELAXF*, Procedimentos de Construção Remota” na página 21 para uma possível solução ao disponibilizar conjuntos diferentes aos usuários.
2. Para ativar duas instâncias do procedimento armazenado do DB2, as tarefas a seguir devem ser realizadas. Observe, entretanto, que esta não é uma descrição suportada da forma como se encontra.
 - a. Copie hlq.SFEKPROC(ELAXMREX) para um membro com nome diferente, por exemplo, ELAXMRXX
 - b. Copie o membro de amostra hlq.SFEKSAMP(ELAXMSAM) para um membro com nome diferente, por exemplo, ELAXMWDZ
 - c. Altere o membro de amostra hlq.SFEKSAMP(ELAXMJCL) para refletir estas alterações de nomes, por exemplo:


```
//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. Proceda com a customização conforme descrito em “(Opcional) Customizar ELAXM*, Membros do Procedimento Armazenado do DB2” na página 26, mas com os novos membros.
 - e. O novo nome do ambiente WLM (por exemplo, ELAXMWDZ) deve ser utilizado no assistente do procedimento armazenado DB2 no cliente.
3. O bidi depende de um membro da biblioteca de carregamento e, portanto, não pode ser compartilhado entre releases. No entanto, se o nome do módulo de carregamento for idêntico para todas as instâncias, você poderá compartilhar a versão mais recente entre as instâncias, entre os releases. A compatibilidade com versões anteriores não estará disponível se o módulo de carregamento mudar de nome.
4. Os módulos de carregamento ADM são compatíveis com versões anteriores, e, assim, a versão mais recente pode ser compartilhada entre os releases.
5. O ADM CRD VSAM é compatível com versões anteriores, e, assim, a versão mais recente pode ser compartilhada entre os releases.
6. As definições de recurso ADM CICS são compatíveis com versões anteriores, e, assim, a versão mais recente pode ser compartilhada entre os releases.
7. Todos os arquivos de configuração e scripts de shell opcionais que devem residir no mesmo diretório que rsed.envvars não podem ser compartilhados, pois rsed.envvars deve estar em um local não-compartilhado.
8. Em rsed.envvars, é possível ativar o compartilhamento de classes Java entre vários servidores RSE, atualizando a diretiva _RSE_JAVA0PTS. O grupo de compartilhamento deve ter um nome exclusivo (-Xshareclasses:name=RSE).

9. Os CARMA VSAMs poderiam ser alterados entre os níveis de software, portanto, não é aconselhável compartilhá-los.

Apêndice B. Resolvendo Problemas de Configuração

Este apêndice é fornecido para auxiliá-lo com alguns problemas comuns que você pode encontrar durante a configuração do Developer for System z.

Mais informações estão disponíveis por meio da seção Support do Web site do Developer for System z (<http://www-306.ibm.com/software/awdtools/devzseries/support/>) no qual você poderá encontrar notas técnicas que fornecem as informações mais recentes de nossa equipe de suporte.

Na seção Library do Web site, você também pode encontrar a versão mais recente da documentação do Developer for System z, inclusive whitepapers.

As valiosas informações também podem ser encontradas na biblioteca do z/OS na Internet, disponível em <http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/>

Localização dos Arquivos de Log

O Developer for System z cria arquivos de log que podem auxiliar você e o centro de suporte da IBM na identificação e solução de problemas. A lista a seguir é uma visão geral dos arquivos de log que podem ser criados. Próximo a estes logs específicos do produto, certifique-se de verificar o SYSLOG em busca de alguma mensagem relacionada.

Os logs baseados no MVS podem ser localizados na instrução DD apropriada. Arquivos de log baseados no z/OS UNIX estão localizados nos seguintes diretórios:

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

A maioria dos arquivos de log está localizada em `home/.eclipse/RSE/USERID/`, em que `home` é o caminho inicial definido no segmento OMVS do usuário (ou o segmento OMVS padrão se o usuário não possui um segmento OMVS) e `USERID` é o ID de usuário do logon (maiúsculas).

- `daemon.log` - O log do daemon
- `fa.log` - O log da integração do Fault Analyzer
- `fekfivpc.log` - O log do teste IVP `fekfivpc`
- `ffs.log` - O log do servidor FFS, que executa funções nativas
- `ffsget.log` - O log do leitor de arquivos, que lê um conjunto de dados seqüencial ou um membro PDS
- `ffsput.log` - O log do gravador de arquivos, que grava um conjunto de dados seqüencial ou um membro PDS
- `lock.log` - O log do gerenciador de bloqueios, que bloqueia/desbloqueia um conjunto de dados seqüencial ou um membro PDS
- `rmt_class_loader.cache.jar` - O cache das classes carregadas pelo carregador da classe remota do RSE
- `rsecomm.log` - O log do servidor RSE, que administra comandos do cliente (pode conter rastreo de pilha de exceção do Java)
- `stderr.log` - Os dados direcionados da saída de erro padrão `stderr`
- `stdout.log` - Os dados direcionados da saída padrão `stdout`.

- **/tmp/**
/tmp/ mantém os arquivos de log criados antes que o ID do usuário seja verificado.
– rsedaemon.log - O log do daemon antes do login

Criação de Logs do JES Job Monitor

- **SYSOUT DD**
Criação de logs de operações normais. O valor padrão no JCL de amostra hlq.SFEKSAMP(FEJJJCL) é SYSOUT=*.
hlq.SFEKSAMP(FEJJJCL) é SYSOUT=*. O rastreamento é ativado com o parâmetro -TV; consulte “Customizar o JCL de Inicialização do JES Job Monitor” na página 18 para obter mais detalhes.
- **SYSPRINT DD**
Criação de logs de rastreamento. O valor padrão no JCL de amostra hlq.SFEKSAMP(FEJJJCL) é SYSOUT=*. O rastreamento é ativado com o parâmetro -TV; consulte “Customizar o JCL de Inicialização do JES Job Monitor” na página 18 para obter mais detalhes.

Criação de Logs de Transações APPC (Serviço de Comandos do TSO)

- **SYSPRINT DD**
Quando o utilitário de administração APPC inclui e modifica um perfil de programa de transação (TP), ele verifica o perfil TP e seu JCL em busca de erros de sintaxe. A saída desta fase consiste de mensagens de erro de sintaxe do perfil TP, mensagens do processamento do utilitário e instruções de conversão do JCL. A criação de logs para mensagens desta fase é controlada pela instrução SYSPRINT DD para o utilitário ATBDFMU. O valor padrão no JCL de amostra hlq.SFEKSAMP(FEKAPPC) é SYSOUT=*. Consulte *MVS Planning: APPC/MVS Management (SA22-7599)* para obter mais detalhes.
- **&SYSUID.FEKFRSRV.&TPDATE.&TPTIME.LOG**
Quando um TP é executado, as mensagens de tempo de execução do TP, tais como mensagens de alocação e finalização, vão para um log nomeado pela palavra-chave MESSAGE_DATA_SET em seu perfil TP. O valor padrão no JCL de amostra hlq.SFEKSAMP(FEKAPPC) é &SYSUID.FEKFRSRV.&TPDATE.&TPTIME.LOG. Consulte *MVS Planning: APPC/MVS Management (SA22-7599)* para obter mais detalhes.

Nota: Dependendo das definições de transações e os padrões de site do APPC, este arquivo de log talvez não apareça, a menos que a palavra-chave KEEP_MESSAGE_LOG(ALWAYS) seja incluída nas definições de transações. Consulte *MVS Planning: APPC/MVS Management (SA22-7599)* para obter informações adicionais.

Criação de Logs do RSE

- **home/.eclipse/RSE/USERID/**
Existem diversos arquivos de log criados pelos componentes relacionados ao RSE, e a maioria deles está localizada em home/.eclipse/RSE/USERID/, em que home é o caminho definido no segmento OMVS do usuário (ou o segmento OMVS padrão se o usuário não possui um segmento OMVS) e USERID é o ID de usuário do logon (maiúsculo).
– daemon.log - O log do daemon
– ffs.log - O log do servidor FFS, que executa funções MVS nativas
– ffsget.log - O log do leitor de arquivos, que lê um conjunto de dados sequencial ou um membro PDS

- `ffsput.log` - O log do gravador de arquivos, que grava um conjunto de dados seqüencial ou um membro PDS
- `lock.log` - O log do gerenciador de bloqueios, que bloqueia/desbloqueia um conjunto de dados seqüencial ou um membro PDS
- `rmt_class_loader.cache.jar` - O cache das classes carregadas pelo carregador da classe remota do RSE
- `rsecomm.log` - O log do servidor RSE, que administra comandos do cliente (pode conter rastreo de pilha de exceção do Java)
- `stderr.log` - Os dados direcionados da saída de erro padrão `stderr`.
- `stdout.log` - Os dados direcionados da saída `stdout` padrão.

A quantidade de dados gravados em `ffs*.log`, `lock.log` e em `rsecomm.log` é controlada pela configuração `debug_level` em `rsecomm.properties`. Consulte “(Opcional) Customizar `rsecomm.properties`, Configuração do Rastreo RSE” na página 51 para obter mais detalhes.

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

Este arquivo contém um log do daemon antes de efetuar login.

Criação de Log IVP `fekfivpc`

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

A saída do comando `fekfivpc -file` (teste IVP relacionado ao SCLM Developer Toolkit), em que `home` é o caminho inicial definido no segmento OMVS do usuário (ou o segmento OMVS padrão se o usuário não possui um segmento OMVS) e `USERID` é o ID de usuário do logon (maiúsculas).

Consulte “Conexão do Serviço Comandos do TSO (Utilizando SCLMDT)” na página 49 para obter mais detalhes.

Criação de Log de Integração do Fault Analyzer

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

A criação de log do Fault Analyzer, em que `home` é o caminho inicial definido no segmento OMVS do usuário (ou o segmento OMVS padrão se o usuário não possuir um segmento OMVS) e `USERID` é o ID do usuário de logon (maiúsculas).

Criação de Log de Integração do File Manager

- **Tarefa do Servidor de Integração do File Manager**

Ao estabelecer uma conexão com o File Manager, uma tarefa do servidor será iniciada, com o ID do usuário como proprietário. Seu nome é `FEKport`, em que `port` é a porta TCP/IP utilizada.

- **SYSPRINT DD**

O SYSPRINT da tarefa do servidor mantém a criação de log do servidor FMI.

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

A criação de log de comunicação do FMI, em que `home` é o caminho inicial definido no segmento OMVS do usuário (ou o segmento OMVS padrão se o usuário não possuir um segmento OMVS) e `USERID` é o ID do usuário de logon (maiúsculas).

A quantidade de dados gravados neste arquivo é controlada pela configuração `debug_level` em `rsecomm.properties`. Consulte “(Opcional) Customizar `rsecomm.properties`, Configuração do Rastreo RSE” na página 51 para obter mais detalhes.

Criação de Logs do CARMA

- Tarefa do Servidor CARMA

Ao estabelecer uma conexão com o CARMA, `hlq.SCRACLST(SCRASUBMT)` iniciará uma tarefa do servidor (com o ID do usuário como proprietário) chamada `CRAport`, em que `port` é a porta TCP/IP utilizada.

- CARMALOG DD

Se o CARMALOG da instrução DD for especificado em `hlq.SCRACLST(SCRASUBMT)`, a criação de log do CARMA será redirecionada para esta instrução DD na tarefa do servidor; caso contrário, vai para `SYSPRINT`.

A quantidade de criação de log é controlada pela configuração do Nível de Rastreio no cliente. Consulte Capítulo 5, “(Opcional) Ativando o IBM CARMA”, na página 55 para obter mais detalhes sobre a configuração do Nível de Rastreio.

- **SYSPRINT DD**

O `SYSPRINT` da tarefa do servidor mantém a criação de log do CARMA, se o CARMALOG da instrução DD não estiver definido.

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

A criação de log de comunicação do CARMA, em que `home` é o caminho inicial definido no segmento OMVS do usuário (ou o segmento OMVS padrão se o usuário não possuir um segmento OMVS) e `USERID` é o ID do usuário de logon (maiúsculas).

A quantidade de dados gravados neste arquivo é controlada pela configuração `debug_level` em `rsecomm.properties`. Consulte “(Opcional) Customizar `rsecomm.properties`, Configuração do Rastreio RSE” na página 51 para obter mais detalhes.

Arquivos de Dump

Quando um produto é finalizado de forma anormal, um dump de armazenamento é criado para auxiliar na determinação do problema. A disponibilidade e localização destes dumps depende quase que totalmente das configurações específicas do site. Portanto, pode ser que eles não sejam criados, ou criados em localizações diferentes da mencionada abaixo.

Dumps do MVS

Quando um programa está em execução no MVS, verifique os arquivos de dump do sistema e verifique o JCL em busca das seguintes instruções DD (dependendo do produto):

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

Consulte a *MVS JCL Reference (SA22-7597)* e o *Language Environment Debugging Guide (GA22-7560)* para obter mais informações sobre estas instruções DD.

Dumps Java

No z/OS UNIX, os dumps do Developer for System z são controlados pela JVM (Java Virtual Machine).

A JVM cria um conjunto de agentes de dump por padrão, durante a inicialização (SYSTDUMP e JAVADUMP). Você pode sobrepor este conjunto de agentes de dump utilizando a variável de ambiente `JAVA_DUMP_OPTS` e sobrepor o conjunto pelo uso de `-Xdump` na linha de comandos. As opções da linha de comandos da JVM são definidas na diretiva `_RSE_JAVA0PTS` de `rsed.envvars`. Não altere nenhuma configuração de dump, a menos que tenha sido solicitado pelo IBM Support Center.

Nota: A opção `-Xdump:what` na linha de comandos pode ser utilizada para determinar quais agentes de dump existem na conclusão da inicialização.

Os tipos de dump que podem ser produzidos são:

SYSTDUMP

Dump de transação Java. Um dump de armazenamento não-formatado gerado pelo z/OS.

O dump é gravado em um conjunto de dados MVS seqüencial, utilizando o nome padrão do formulário `&userid.JVM.TDUMP.&jobname.D&date.T&time` ou como determinado pela configuração da variável de ambiente `JAVA_DUMP_TDUMP_PATTERN`. Se você não deseja que dumps de transação sejam criados, inclua a variável de ambiente `IBM_JAVA_ZOS_TDUMP=NO` em `rsed.envvars`.

CEEDUMP

Dump do LE (Language Environment). Um dump do sistema de resumo formatado que mostra rastreios de pilha para cada encadeamento que está no processo JVM, junto com informações de registro e um dump curto de armazenamento para cada registro.

O dump é gravado em um arquivo z/OS UNIX chamado `CEEDUMP.aaaammdd.hhmmss.pid`, em que `aaaammdd` é igual à data atual, `hhmmss` é a hora atual e `pid` é o ID do processo atual. Os locais possíveis deste arquivo são descritos em “Locais de Dump do z/OS UNIX” na página 80.

HEAPDUMP

Um dump formatado (uma lista) dos objetos que estão no heap Java.

O dump é gravado em um arquivo z/OS UNIX chamado `HEAPDUMP.aaaammdd.hhmmss.pid.TXT`, em que `aaaammdd` é igual à data atual, `hhmmss` é a hora atual e `pid` é o ID do processo atual. Os locais possíveis deste arquivo são descritos em “Locais de Dump do z/OS UNIX” na página 80.

JAVADUMP

Uma análise formatada da JVM. Contém informações de diagnóstico relacionadas à JVM e ao aplicativo Java, como o ambiente de aplicativos, encadeamentos, pilha nativa, bloqueios e memória.

O dump é gravado em um arquivo z/OS UNIX chamado `JAVADUMP.aaaammdd.hhmmss.pid.TXT`, em que `aaaammdd` é igual à data atual, `hhmmss` é a hora atual e `pid` é o ID do processo atual. Os locais possíveis deste arquivo são descritos em “Locais de Dump do z/OS UNIX” na página 80.

Consulte o *Java Diagnostic Guide* (SC34-6358) para obter mais informações sobre dumps JVM e o *Language Environment Debugging Guide* (GA22-7560) para informações específicas do LE

Locais de Dump do z/OS UNIX

A JVM verifica cada um dos seguintes locais quanto à existência e permissão de gravação, e armazena os arquivos CEEDUMP, HEAPDUMP e JAVADUMP no primeiro disponível. Observe que você deve possuir espaço em disco livre suficiente para que o arquivo de dump seja gravado corretamente.

1. O diretório na variável de ambiente `_CEE_DMPTARG`, se localizado. Essa variável é definida em `rzed.envvars` como `home/.eclipse/RSE/USERID`, em que `home` é o caminho inicial definido no segmento OMVS do usuário (ou o segmento OMVS padrão se o usuário não possui um segmento OMVS) e `USERID` é o ID de usuário do logon (maiúsculas).
2. O diretório de trabalho atual, se o diretório não for o diretório raiz (`/`) e se o diretório for gravável.
3. O diretório na variável de ambiente `TMPDIR` (uma variável de ambiente que indica o local de um diretório temporário se ele não for `/tmp`), se localizado.
4. O diretório `/tmp`.
5. Se o dump não puder ser armazenado em nenhum dos acima, ele é colocado no `stderr`.

Autorização do Controle de Programas para Programas RSE

O RSE (Remote Systems Explorer) é o componente do Developer for System z que fornece os principais serviços, como de conexão do cliente ao host. Ele deve executar o programa controlado para desempenhar tarefas como a comutação para o ID de usuário do cliente.

O bit de controle de programa é configurado durante a instalação do SMP/E, onde necessário.

Esse resultado pode ser verificado com o comando `ls -lE fekf*`, que fornece uma saída como a seguinte amostra (\$ é o prompt z/OS UNIX):

```
$ ls -lE /usr/lpp/wd4z/rse/lib/fekf*
-rwxr-xr-x -ps- 2 user group 94208 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfdir.dll
-rwxr-xr-x -ps- 2 user group 143360 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfdivp
-rwxr-xr-x --s- 2 user group 480 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfivpa
-rwxr-xr-x --s- 2 user group 342 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfivpc
-rwxr-xr-x --s- 2 user group 445 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfivpd
-rwxr-xr-x --s- 2 user group 1491 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfivpj
-rwxr-xr-x --s- 2 user group 883 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfivpr
-rwxr-xr-x --s- 2 user group 307 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfivps
-rwxr-xr-x -ps- 2 user group 139264 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekflock
-rwxr-xr-x -ps- 2 user group 196608 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfrsed
-rwxr-xr-x --s- 2 user group 42443 Jun 12 16:21 /usr/lpp/wd4z/rse/lib/fekfscmd
```

Nota: `fekfivp*` e `fekfscmd` não requerem o atributo `+p`.

Emita os seguintes comandos se o bit de controle de programa precisar ser configurado manualmente, considerando que o diretório de instalação padrão (`/usr/lpp/wd4z/rse/lib/`) é utilizado:

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

Nota: Para poder utilizar o comando **extattr +p**, você deve ter pelo menos acesso de LEITURA ao perfil BPX.FILEATTR.PROGCTL na classe FACILITY do software de segurança. Para obter mais informações, consulte *UNIX System Services Planning (GA22-7800)*.

Portas TCP/IP Reservadas

Com o comando **NETSTAT** (TSO ou z/OS UNIX), você pode obter uma visão geral das portas atualmente em uso. A saída deste comando será semelhante ao exemplo abaixo. As portas utilizadas são os últimos números (atrás de '..') na coluna "Soquete Local". Como estas portas já estão em uso, elas não podem ser utilizadas para a configuração do Developer for System z.

```
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
```

Outra limitação que pode existir são as portas TCP/IP reservadas. Existem 2 locais comuns para reservar portas TCP/IP:

- **PROFILE.TCPIP**

Este é o conjunto de dados consultado pela instrução PROFILE DD da tarefa iniciada do TCP/IP, chamada geralmente de SYS1.TCPPARMS(TCPPROF)

- PORT: Reserva uma porta para nomes de tarefas especificados.
- PORTRANGE: Reserva um intervalo de portas para nomes de tarefas especificados.

Consulte o *Communications Server: IP Configuration Guide (SC31-8775)* para obter mais informações sobre estas instruções.

- **SYS1.PARMLIB(BPXPRMxx)**

- INADDRANYPORT: Especifica o número da porta inicial para o intervalo de números de portas que o sistema reserva para utilização com PORT 0, ligações INADDR_ANY. Este valor é necessário somente para CINET.
- INADDRANYCOUNT: Especifica o número de portas que o sistema reserva, iniciando com o número de porta especificado no parâmetro INADDRANYPORT. Este valor é necessário somente para CINET.

Consulte *UNIX System Services Planning (GA22-7800)* e *MVS Initialization and Tuning Reference (SA22-7592)* para obter mais informações sobre estas instruções.

Estas portas reservadas podem ser listadas com o comando **NETSTAT PORTL** (TSO ou z/OS UNIX), que cria uma saída como no exemplo abaixo:

```
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
```

Consulte *Communications Server: IP System Administrator's Commands* (SC31-8781) para obter mais informações sobre o comando **NETSTAT**.

Nota: O comando **NETSTAT** mostra somente as informações definidas em **PROFILE.TCPIP**, que devem sobrepor as definições de **BPXPRMxx**. Em caso de dúvidas ou problemas, verifique o membro **parmlib** de **BPXPRMxx** para verificar as portas sendo reservadas aqui.

Nota: Leia “Definições de Portas **PROFILE.TCPIP**” na página 100 se você tiver problemas com a ligação **INETD** para portas reservadas.

Tamanho do Espaço de Endereço

O servidor RSE, que é um processo Java UNIX, requer um tamanho de região grande para desempenhar essas funções. Portanto, é importante definir limites de armazenamento grandes para espaços de endereço do OMVS.

Requisitos do INETD

Um servidor RSE é iniciado pelo **INETD** quando um cliente se conecta à porta do daemon RSE ou quando o cliente emite o script de inicialização utilizando **REXEC/SSH**. Isso é feito utilizando os limites de armazenamento do **INETD**, portanto devem ser definidos suficientemente altos.

- Se o **INETD** do processo do servidor UNIX for iniciado pelo **JCL** utilizando **BPXBATCH**, o tamanho da região deve ser 0.
- Se o **INETD** for iniciado a partir de uma sessão **TSO/OMVS**, o tamanho da região **TSO** deve ser 2096128 ou maior.

Limitações Definidas em SYS1.PARMLIB(BPXPRMxx)

Configure **MAXASSIZE** em **SYS1.PARMLIB(BPXPRMxx)**, que define o tamanho da região do espaço de endereço do OMVS (processo) como 2147483647 ou maior.

Esse valor pode ser verificado e configurado dinamicamente (até o próximo IPL) com os seguintes comandos do console, conforme descrito em *MVS System Commands* (GC28-1781):

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

Limitações Armazenadas no Perfil de Segurança

Verifique **ASSIZEMAX** no segmento OMVS do ID do usuário do cliente e defina-o como 2147483647 ou, preferencialmente, como **NONE** para utilizar o valor de **SYS1.PARMLIB(BPXPRMxx)**.

Utilizando o **RACF**, esse valor pode ser verificado e configurado com os seguintes comandos do **TSO**, conforme descrito em *Security Server RACF Command Language Reference* (SA22-7687):

1. **LISTUSER userid NORACF OMVS**
2. **ALTUSER userid OMVS(NOASSIZEMAX)**

Limitações Impostas por Saídas do Sistema

Certifique-se de não estar permitindo que saídas do sistema IEFUSI ou IEALIMIT controlem tamanhos de região do espaço de endereço do OMVS. Uma forma possível de fazer isso é pela codificação de SUBSYS(OMVS,NOEXITS) em SYS1.PARMLIB(SMFPRMxx).

Os valores de SYS1.PARMLIB(SMFPRMxx) podem ser verificados e ativados com os seguintes comandos do console, conforme descrito em *MVS System Commands (GC28-1781)*:

1. DISPLAY SMF,0
2. SET SMF=xx

Rastreo de Feedback de Erro

O procedimento a seguir permite reunir informações necessárias para diagnosticar problemas de feedback de erro com procedimentos de construção remota. Esse rastreo causará diminuição no desempenho e deverá ser realizado somente sob a orientação do IBM Support Center. Todas as referências a hlq nesta seção referem-se ao qualificador de alto nível utilizado durante a instalação do Developer for System z. O padrão da instalação é FEK, mas isto talvez não se aplique ao seu site.

1. Faça uma cópia de backup do procedimento de compilação ELAXFCOC ativo. Esse procedimento é o padrão fornecido no conjunto de dados hlq.SFEKSAMP, mas pode ter sido copiado para um local diferente, como SYS1.PROCLIB, conforme descrito em “Customizar ELAXF*, Procedimentos de Construção Remota” na página 21.
2. Altere o procedimento ELAXFCOC ativo para incluir a cadeia 'MAXTRACE' na opção de compilação EXIT(ADEXIT(ELAXMGUX)).

```
//COBOL EXEC PGM=IGYCRCTL,REGION=2048K,  
//*      PARM=('EXIT(ADEXIT(ELAXMGUX))',  
//      PARM=('EXIT(ADEXIT('MAXTRACE',ELAXMGUX))',  
//      'ADATA',  
//      'LIB',  
//      'TEST(NONE,SYM,SEP)',  
//      'LIST',  
//      'FLAG(I,I)'&CICS &DB2 &COMP)
```

Nota: É necessário duplicar os apóstrofes em MAXTRACE. A opção é agora: EXIT(ADEXIT('MAXTRACE',ELAXMGUX))

3. Desempenhe uma Verificação de Sintaxe Remota em um programa COBOL. O Developer for System z oferece um programa COBOL de amostra em hlq.SFEKSAMP(ADNTMSGH).
4. A parte SYSOUT da saída JES começará listando os nomes dos conjuntos de dados para SIDEFILE1, SIDEFILE2, SIDEFILE3 e SIDEFILE4.

```
ABOUT TOO OPEN SIDEFILE1 - NAME = 'uid.DT021207.TT110823.M0000045.C0000000'  
SUCCESSFUL OPEN SIDEFILE1 - NAME = 'uid.DT021207.TT110823.M0000045.C0000000'  
ABOUT TOO OPEN SIDEFILE2 - NAME = 'uid.DT021207.TT110823.M0000111.C0000001'  
SUCCESSFUL OPEN SIDEFILE2 - NAME = 'uid.DT021207.TT110823.M0000111.C0000001'  
ABOUT TOO OPEN SIDEFILE3 - NAME = 'uid.DT021207.TT110823.M0000174.C0000002'  
SUCCESSFUL OPEN SIDEFILE3 - NAME = 'uid.DT021207.TT110823.M0000174.C0000002'  
ABOUT TOO OPEN SIDEFILE4 - NAME = 'uid.DT021207.TT110823.M0000236.C0000003'  
SUCCESSFUL OPEN SIDEFILE4 - NAME = 'uid.DT021207.TT110823.M0000236.C0000003'
```

Nota: Dependendo das suas configurações, SIDEFILE1 e SIDEFILE2 podem estar apontando para uma instrução DD (SUCCESSFUL OPEN SIDEFILE1 - NAME

= DD:WSEDSF1). Consulte a parte JESJCL da saída (localizada antes da parte SYSOUT) para obter o nome real do conjunto de dados.

```
22 //COBOL.WSEDSF1 DD DISP=MOD,  
    // DSN=uid.ERRCOB.member.SF1.Z682746.XML  
23 //COBOL.WSEDSF2 DD DISP=MOD,  
    // DSN=uid.ERRCOB.member.SF1.Z682747.XML
```

5. Copie esses quatro conjuntos de dados para seu PC, por exemplo, criando um projeto COBOL local no Developer for System z e incluindo os conjuntos de dados SIDEFILE1->4.
6. Copie o log da tarefa do JES completo para seu PC, por exemplo, abrindo a saída da tarefa no Developer for System z e salvando-a no projeto local, selecionando **Arquivo > Salvar como ...**.
7. Restaure o procedimento ELAXFC0C para o estado original, desfazendo a alteração (remova a cadeia "MAXTRACE" nas opções de compilação) ou restaurando o backup.
8. Envie os arquivos coletados (SIDEFILE1->4 e log da tarefa) para o IBM Support Center.

Transação APPC e Serviço de Comandos do TSO

Se não puder utilizar a versão APPC do serviço Comandos do TSO, haverá duas áreas nas quais os problemas poderão surgir: iniciando a transação do servidor APPC e conectando ao RSE.

- Se as mensagens sobre a configuração do APPC não forem exibidas, procure no log do sistema mensagens RACF (ID de mensagem ICHxxxxx) ou outras mensagens relacionadas ao comando que foi emitido ou o ID de usuário que o emitiu. As causas comuns dos problemas incluem:
 - Você não possui autoridade de leitura ao conjunto de dados hlq.SFEKPROC.
 - O TCP/IP não está ativo, possui um nome DNS incorreto anexado ou o sistema está inatingível (sem resposta ao efetuar ping) devido a problemas de rede, um endereço de IP incorreto ou outras causas.
- Se forem exibidas as mensagens sobre a configuração do APPC, mas não for exibida a mensagem confirmando que a configuração foi bem-sucedida, a transação do servidor APPC provavelmente não poderá ser iniciada. Consulte o log de erros de transação (userid.FEKFRSRV.&TPDATE.&TPTIME.LOG). Algumas das prováveis causas dos problemas são:
 - A pilha do TCP/IP não está utilizando o nome padrão do TCPIP e o cartão SYSTCPD DD não foi configurado ou está apontando para o conjunto de dados incorreto.
 - O servidor não pôde alocar SYSPROC ou SYSTSPRT.
 - O JCL aponta para o SYSPROC incorreto (o SYSPROC deve incluir hlq.SFEKPROC).
 - O servidor não pôde abrir ou acessar o conjunto de dados de mensagens (log) consultado pelo MESSAGE_DATA_SET.
 - Não há inicializadores do planejador do APPC suficientes disponíveis.
 - Os espaços de endereços de APPC ou ASCH não estão ativos.
 - A classe utilizada (chamada, por padrão, de "A") não está definida para o ASCH do planejador do APPC.
 - Não há segmento OMVS padrão para o sistema, e o usuário não possui um segmento OMVS pessoal ou há um erro de definição em um dos dois.
 - O grupo padrão do segmento OMVS padrão ou o grupo padrão do usuário não tem um número de GID.

O REXX fornecido em “(Opcional) Definir uma Transação APPC para o Serviço de Comandos do TSO” na página 23 pode ajudar com a solução de problemas APPC desde que ele forneça a possibilidade de gerenciar o APPC de forma interativa por meio dos painéis do ISPF. Esteja ciente, entretanto, de que é possível desativar a transação com esta ferramenta; a transação ainda está lá, mas não aceitará nenhuma conexão.

A lista a seguir é uma seleção de notas técnicas atualmente disponíveis no Web site de suporte (<http://www-306.ibm.com/software/awdtools/devzseries/support/>). Consulte o Web site de suporte para obter informações adicionais:

- A verificação do APPC falha com o código de retorno 2016 - EHOSTNOTFOUND
- A verificação do APPC falha com o código de retorno 1004 - EIBMIUCVERR
- A verificação do APPC falha com o código de retorno 9 - Nome do TP não reconhecido
- A verificação do APPC falha com o código de retorno 10 - TP não disponível; sem nova tentativa
- A verificação do APPC falha com o código de retorno 19 - Erro do Parâmetro
- A verificação do APPC falha com o código de retorno 20 - Erro específico do produto
- A verificação do APPC falha com o código de retorno 26 - Falha do recurso
- CEE3501S: O módulo IOSTREAM não foi localizado
- Servidor Falhou ao Iniciar: EDC5129I Nenhum arquivo ou diretório
- exec/tcp: bind: EDC5111I Permissão negada, rsn=744C7246
- Nenhuma resposta do servidor, com uma destas mensagens
 - IEA995I SAÍDA DO DUMP DO SIMTOMA 473 CÓDIGO DE CONCLUSÃO DO USUÁRIO=4093 CÓDIGO DE RAZÃO=0000001C (no LOG SDSF)
 - CEE3512S Um carregamento HFS do módulo libicudata32.0.dll falhou. O código de retorno do sistema era 0000000157; o código de razão era 0BDF019B. (em CEEDUMP)
 - Falha ao obter espaço (no cliente .log)
- Comando C_CONNECT não está disponível
- Mensagem de erro “Falha na inicialização do servidor FFS” ao se conectar ao host
- “EDC5139I Operação não permitida” ao se conectar ao host
- “RSEG1056U Falha na inicialização do servidor FFS” ao abrir um arquivo do MVS

Nota: Esta lista não é definitiva; consulte o Web site de suporte para obter notas técnicas adicionais.

Informações Diversas

Limites do Sistema

SYS1.PARMLIB(BPXPRMxx) define muitas limitações relacionadas ao z/OS UNIX, que pode ser acessado quando muitos clientes do Developer for System z estão ativos. A maioria dos valores de BPXPRMxx pode ser alterada dinamicamente com os comandos do console **SETOMVS** e **SET OMVS**.

Conexão Recusada

Cada conexão RSE inicia diversos processos que são permanentemente ativos. Novas conexões podem ser recusadas devido ao limite definido em SYS1.PARMLIB(BPXPRMxx) na quantidade de processos, especialmente quando os usuários compartilham o mesmo UID (como na utilização do segmento OMVS padrão).

- O limite por UID é definido pela palavra-chave MAXPROCUSER e possui um valor padrão de 25.
- O limite do sistema é definido pela palavra-chave MAXPROCSYS e possui um valor padrão de 200.

Outra origem de conexões recusadas é o limite da quantidade de espaços de endereço do z/OS e de usuários do z/OS UNIX ativos.

- A quantidade máxima de ASIDs (Address Space IDs) é definida em SYS1.PARMLIB(IEASYSxx) com a palavra-chave MAXUSER e possui o valor padrão de 255.
- A quantidade máxima de UIDs (z/OS UNIX User IDs) é definida em SYS1.PARMLIB(BPXPRMxx) com a palavra-chave MAXUIDS e possui o valor padrão de 200.

Problemas Conhecidos

Falha ao Abrir Conjuntos de Dados MVS

A leitura e gravação de um conjunto de dados MVS requer o uso de um domínio do sistema de arquivo físico do soquete. Se o sistema de arquivo não estiver definido adequadamente ou não tiver soquetes suficientes, o gerenciador de bloqueio (FFS) poderá falhar nos pedidos de leitura/gravação. Os arquivos ffs*.log mostrarão mensagens como a seguinte:

- Erro 127 ao obter o par de soquetes - configurando a porta como 0.
- Não foi possível criar o soquete no domínio do UNIX. O erro é: "A família de endereços não é suportada"

Verifique se o membro SYS1.PARMLIB(BPXPRMxx) contém as seguintes instruções:

```
FILESYSTYPE TYPE(UDS) ENTRYPPOINT(BPXTUINT)
NETWORK DOMAINNAME(AF_UNIX)
          DOMAINNUMBER(1)
          MAXSOCKETS(200)
          TYPE(UDS)
```

Falha nas Ligações DVIPA

Ao utilizar VIPA (Virtual IP Address) dinâmico em várias pilhas TCPIP, é necessária coordenação no sysplex da designação de portas transitórias para que o 4-tuple (combinação de endereços e portas IP de origem e de destino) para cada conexão permaneça exclusivo. Isso pode ser feito incluindo o parâmetro SYSPLEXPORTS opcional na primeira instrução VIPADISTRIBUTE, conforme descrito no *IP Configuration Guide (SC31-8775)*.

Ao utilizar essa técnica, certifique-se de que a estrutura do recurso de acoplamento EZBEPOR (contendo informações de designação de porta sysplex) tenha sido definida. Consulte o *SNA Network Implementation Guide (SC31-8777)* para obter informações sobre como fazer isso.

Emulador de Conexão do Host

- O Emulador de Conexão do Host utiliza o telnet TN3270, e não o servidor RSE, para se conectar ao host.
- Quando você utiliza o telnet seguro (SSL) e está trabalhando com certificados que não são assinados por uma CA bem conhecida, cada cliente deve incluir o certificado CA na lista de Emuladores de Conexão do Host de CAs confiáveis.
- A opção NOSNAEXT de TELNETPARMS do TCP/IP pode ser necessária para desativar as extensões funcionais do SNA. Se NOSNAEXT for especificado, o servidor telnet TN3270 não negociará as funções de resolução de contenção e de detecção do SNA.

Entrando em Contato com o Suporte IBM

Se você ainda estiver com dificuldades depois de ler este manual e consultar o Web site de suporte (<http://www-306.ibm.com/software/awdtools/devzseries/support/>) e for necessária assistência do suporte da IBM, reúna as seguintes informações e abra um registro de problema com a IBM:

- Seu nível de software atual, que está documentado em:
 - hlq.SFEKJCL(FEK#LVL#) para Developer for System z. O valor padrão para hlq é FEK.
 - hlq.SCRAJCL(CRA#LVL#) para CARMA. O valor padrão para hlq é CRA.

As seguintes informações são úteis para resolver problemas de conexão

- A saída do(s) seguinte(s) comando(s) de console:
 - D OMVS,0
- A saída do(s) seguinte(s) comando(s) do TSO:
 - HOMETEST
 - LISTUSER userid NORACF OMVS
 - LISTGROUP group NORACF OMVS
- Arquivo rsed.envvars, localizado por padrão em /usr/lpp/wd4z/rse/lib/, mas é recomendável que seja customizado em /etc/wd4z/
- Arquivo /etc/inetd.conf
- Todos os arquivos *.log localizados em home/.eclipse/RSE/USERID/, em que home é o caminho inicial definido no segmento OMVS do usuário (ou o segmento OMVS padrão se o usuário não possui um segmento OMVS) e USERID é o ID de usuário do logon (maiúsculas).
- Mensagens SYSLOG relacionadas
- Saída completa do(s) teste(s) IVP relacionado(s), conforme descrito em “Verificar a Configuração do Servidor RSE” na página 46.
 - netstat : “Disponibilidade de Porta” na página 47
 - fekfivps : “Script de Shell REXEC/SSH” na página 48
 - fekfivpr : “Conexão REXEC” na página 47
 - fekfivpd : “Conexão do Daemon RSE” na página 48.
 - fekfivpj : “Conexão do JES Job Monitor” na página 49.
 - fekfivpc : “Conexão do Serviço Comandos do TSO (Utilizando SCLMDT)” na página 49
 - fekfivpa : “Conexão do Serviço Comandos do TSO (Utilizando APPC)” na página 50

Se você utilizar o SCLM Developer Toolkit para o serviço Comandos do TSO (método padrão)

- Saída completa do teste IVP SCLMDT (fekfivpc), conforme descrito em “Conexão do Serviço Comandos do TSO (Utilizando SCLMDT)” na página 49
- Arquivo /etc/SCLMDT/CONFIG/ISPF.conf

Se você utilizar o APPC para o serviço Comandos do TSO

- Saída completa do teste IVP APPC (fekfivpa), conforme descrito em “Conexão do Serviço Comandos do TSO (Utilizando APPC)” na página 50
- O JCL que define a transação do servidor APPC, que é hlq.SFEKSAMP(FEKAPPCC), se você tiver utilizado a tarefa de amostra
- Membro SYS1.PARMLIB(APPCPMxx)
- Membro SYS1.PARMLIB(ASCHPMxx)
- Conjunto de dados userid.FEKFSERV.&TPDATE.&TPTIME.LOG
- A saída dos seguintes comandos de console:
 - D APPC,TP,ALL
 - D APPC,LU,ALL
 - D ASCH,ALL

Forneça todas as informações que pareçam relevantes para problemas funcionais, como

- Criação de log da tarefa
- Mensagens SYSLOG relacionadas
- Arquivos de log localizados em home/.eclipse/RSE/USERID/, em que home é o caminho inicial definido no segmento OMVS do usuário (ou o segmento OMVS padrão se o usuário não possui um segmento OMVS) e USERID é o ID de usuário do logon (maiúsculas).

Apêndice C. Configurando o TCP/IP

Este apêndice é fornecido para auxiliá-lo com alguns problemas comuns que você pode encontrar ao configurar o TCP/IP, ou durante a verificação e/ou modificação de uma configuração existente.

Consulte *Communications Server: IP Configuration Guide (SC31-8775)* e *Communications Server: IP Configuration Reference (SC31-8776)* para obter informações adicionais sobre a configuração do TCP/IP.

Dependência do Nome do Host

O Developer for System z depende de o TCP/IP ter o nome do host correto quando for inicializado. Isto significa que o TCP/IP diferente e os arquivos de configuração do Resolver devem ser configurados corretamente.

Você pode testar sua configuração TCP/IP com o comando **HOMETEST** do TSO. Consulte *Communications Server: IP System Administrator's Commands (SC31-8781)* para obter informações adicionais sobre o comando **HOMETEST**.

Saída de exemplo do comando **HOMETEST**:

Executando o Testador de Configuração TCP/IP IBM MVS TCP/IP CS V1R7

O arquivo de parâmetros de configuração de FTP utilizado
será "SYS1.TCPPARMS(FTPDATA)"

O Nome do Host TCP é: CDFMVS08

Utilizando o Servidor de Nomes para Resolver CDFMVS08
Os endereços IP a seguir correspondem ao Nome do Host TCP: CDFMVS08
9.42.112.75

Os endereços IP a seguir são os endereços IP HOME definidos no PROFILE.TCPIP:
9.42.112.75
127.0.0.1

Todos os endereços IP do CDFMVS08 estão na lista HOME!

O Hometest foi bem-sucedido – todos os Testes Passaram!

Compreendendo os Resolvedores

O resolvedor atua em nome de programas como um cliente que acessa servidores de nomes para resolução de nome para endereço e de endereço para nome. Para resolver a consulta do programa solicitante, o resolvedor pode acessar os servidores de nomes disponíveis, utilize definições locais (por exemplo, /etc/resolv.conf, /etc/hosts, /etc/ipnodes, HOSTS.SITEINFO, HOSTS.ADDRINFO ou ETC.IPNODES) ou utilize uma combinação delas.

Quando o espaço de endereços do resolvedor é iniciado, ele lê um conjunto de dados de configuração do resolvedor opcional apontado pelo cartão SETUP DD no procedimento JCL do resolvedor. Se as informações de configuração não forem fornecidas, o resolvedor utiliza a ordem de procura nativa aplicável do MVS ou do z/OS UNIX sem qualquer informação de GLOBALTCPIPDATA, DEFAULTTCPIPDATA, GLOBALIPNODES, DEFAULTIPNODES ou COMMONSEARCH.

Compreendendo as Ordens de Procura das Informações de Configuração

É importante compreender o ordem de procura dos arquivos de configuração utilizados pelas funções TCP/IP, e quando você pode sobrescrever a ordem de procura padrão com variáveis de ambiente, JCL ou outras variáveis fornecidas. Este conhecimento permite acomodar o conjunto de dados local e padrões de nomenclatura de arquivos HFS, e é útil conhecer o conjunto de dados da configuração ou o arquivo HFS em uso ao diagnosticar problemas.

Um outro ponto importante a ser observado é que quando uma ordem de procura é aplicada para qualquer arquivo de configuração, a procura finaliza com o primeiro arquivo localizado. Portanto, podem ocorrer resultados inesperados se você colocar informações de configuração em um arquivo que nunca é encontrado, pois outros arquivos aparecem antes na ordem de procura, ou porque o arquivo não está incluído na ordem de procura escolhida pelo aplicativo.

Ao procurar por arquivos de configuração, você pode informar explicitamente ao TCP/IP onde estão a maioria dos arquivos de configuração utilizando instruções DD nos procedimentos JCL ou configurando variáveis de ambiente. Caso contrário, você pode permitir que o TCP/IP determine dinamicamente a localização dos arquivos de configuração, com base nas ordens de procura documentadas no *Communications Server: IP Configuration Guide (SC31-8775)*.

O componente de configuração da pilha do TCP/IP utiliza o TCPIP.DATA durante a inicialização da pilha do TCP/IP para determinar o HOSTNAME da pilha. Para obter seu valor, a ordem de procura do ambiente do z/OS UNIX é utilizada.

Nota: Utilize o recurso do resolvidor de rastreo para determinar que os valores de TCPIP.DATA estão sendo utilizados pelo resolvidor e de onde eles foram lidos. Para obter informações sobre o início dinâmico do rastreo, consulte o *Communications Server: IP Diagnosis Guide (GC31-8782)*. Assim que o rastreo for ativado, emita um comando **NETSTAT HOME** do TSO e um comando **netstat -h** do shell do z/OS UNIX para exibir os valores. A emissão de um PING de um nome de host do TSO e a partir do shell do z/OS UNIX também mostra a atividade de qualquer servidor DNS que possa estar configurado.

Ordens de Procura Utilizadas no Ambiente z/OS UNIX

O arquivo ou tabela específico que é procurado pode ser um conjunto de dados MVS ou um arquivo HFS, dependendo das definições de configuração do resolvidor e da presença de determinados arquivos no sistema.

Arquivos de Base da Configuração do Resolvidor

O arquivo de base da configuração do resolvidor contém instruções TCPIP.DATA. Além das diretivas do resolvidor, ele é referido para determinar, entre outras coisas, o prefixo do conjunto de dados (valor da instrução DATASETPREFIX) a ser utilizado ao tentar acessar alguns arquivos de configuração especificados nesta seção.

A ordem de procura utilizada para acessar o arquivo de base da configuração do resolvidor é conforme a seguir:

1. GLOBALTCPIPDATA

Se definido, o valor da instrução de configuração GLOBALTCPIPDATA do resolvidor é utilizado (consulte também “Compreendendo os Resolvedores” na página 89). A procura continua por um arquivo de configuração adicional. A procura finaliza com o próximo arquivo localizado.

2. O valor da variável de ambiente **RESOLVER_CONFIG**

O valor da variável de ambiente é utilizado. Esta procura falhará se o arquivo não existir ou estiver alocado exclusivamente em outro lugar.

3. **/etc/resolv.conf**

4. Cartão **//SYSTCPD DD**

O conjunto de dados alocado para o SYSTCPD de nome DD é utilizado. No ambiente z/OS UNIX, um processo-filho não possui acesso ao DD SYSTCPD. Isto porque a alocação de SYSTCPD não é herdada do processo-pai por meio das chamadas de função fork() ou exec.

5. **userid.TCPIP.DATA**

userid é o ID do usuário associado ao ambiente de segurança atual (espaço de endereço ou tarefa/encadeamento).

6. **jobname.TCPIP.DATA**

jobname é o nome especificado na instrução JOB do JCL para tarefas em lote ou o nome do procedimento para um procedimento iniciado.

7. **SYS1.TCPPARMS(TCPDATA)**

8. **DEFAULTTCPIPDATA**

Se definido, o valor da instrução de configuração DEFAULTTCPIPDATA do resolvidor é utilizado (consulte também “Compreendendo os Resolvedores” na página 89).

9. **TCPIP.TCPIP.DATA**

Tabelas de Conversão

As tabelas de conversão (EBCDIC-to-ASCII e ASCII-to-EBCDIC) são consultadas para determinar os conjuntos de dados de conversão a serem utilizados. A ordem de procura utilizada para acessar este arquivo de configuração é conforme a seguir. A ordem de procura finaliza no primeiro arquivo localizado:

1. O valor da variável de ambiente **X_XLATE**. O valor da variável de ambiente é o nome da tabela de conversão produzida pelo comando CONVXLAT do TSO.

2. **userid.STANDARD.TCPXLBIN**

userid é o ID do usuário associado ao ambiente de segurança atual (espaço de endereço ou tarefa/encadeamento).

3. **jobname.STANDARD.TCPXLBIN**

jobname é o nome especificado na instrução JOB do JCL para tarefas em lote ou o nome do procedimento para um procedimento iniciado.

4. **hlq.STANDARD.TCPXLBIN**

hlq representa o valor da instrução DATASETPREFIX especificada no arquivo de base da configuração do resolvidor (se localizado); caso contrário, hlq é TCPIP por padrão.

5. Se nenhuma tabela for localizada, o resolvidor utiliza uma tabela padrão de código permanente, idêntica à tabela listada no membro do conjunto de dados SEZATCPX (STANDARD).

Tabelas do Host Local

Por padrão, o resolvidor primeiro tenta utilizar qualquer servidor de nomes de domínios configurado para pedidos de resolução. Se o pedido de resolução não

puder ser satisfeito, as tabelas do host local são utilizadas. O comportamento do resolvedor é controlado pelas instruções TCPIP.DATA.

As instruções do resolvedor TCPIP.DATA definem se e como os servidores de nomes de domínios devem ser utilizados. A instrução LOOKUP TCPIP.DATA também pode ser utilizada para controlar como os servidores de nomes de domínios e as tabelas do host local são utilizadas. Para obter informações adicionais sobre as instruções TCPIP.DATA, consulte *Communications Server: IP Configuration Reference (SC31-8776)*.

O resolvedor utiliza a ordem de procura exclusiva Ipv4 para obter informações de nomes de sites incondicionalmente para chamadas de API getnetbyname. A ordem de procura exclusiva Ipv4 para obter informações de nome do site é conforme a seguir. A procura finaliza no primeiro arquivo localizado:

1. O valor da variável de ambiente **X_SITE**
O valor da variável de ambiente é o nome do arquivo de informações HOSTS.SITEINFO criado pelo comando **MAKESITE** do TSO.
2. O valor da variável de ambiente **X_ADDR**
O valor da variável de ambiente é o nome do arquivo de informações HOSTS.ADDRINFO criado pelo comando **MAKESITE** do TSO.
3. **/etc/hosts**
4. **userid.HOSTS.SITEINFO**
userid é o ID do usuário associado ao ambiente de segurança atual (espaço de endereço ou tarefa/encadeamento).
5. **jobname.HOSTS.SITEINFO**
jobname é o nome especificado na instrução JOB do JCL para tarefas em lote ou o nome do procedimento para um procedimento iniciado.
6. **hlq.HOSTS.SITEINFO**
hlq representa o valor da instrução DATASETPREFIX especificada no arquivo de base da configuração do resolvedor (se localizado); caso contrário, hlq é TCPIP por padrão.

Aplicando Isto no Developer for System z

Conforme informado anteriormente, o Developer for System z depende de o TCP/IP ter o nome do host correto quando for inicializado. Isto significa que o TCP/IP diferente e os arquivos de configuração do Resolver devem ser configurados corretamente.

No exemplo abaixo teremos como foco algumas tarefas de configuração para TCP/IP e Resolver. Observe que isto não abrange uma configuração completa de TCP/IP ou Resolver, ela destaca alguns aspectos principais que podem ser aplicáveis no seu site.

1. No JCL abaixo vemos que o TCP/IP utilizará SYS1.TCPPARMS(TCPDATA) para determinar o nome do host da pilha.

```
//TCPIP    PROC  PARMS='CTRACE(CTIEZB00)',PROF=TCPPROF,DATA=TCPDATA
//*
/** TCP/IP NETWORK
/**
//TCPIP    EXEC  PGM=EZBTCPIP,REGION=0M,TIME=1440,PARM=&PARMS
//PROFILE  DD   DISP=SHR,DSN=SYS1.TCPPARMS(&PROF)
//SYSTCPD  DD   DISP=SHR,DSN=SYS1.TCPPARMS(&DATA)
//SYSPRINT DD   SYSOUT=*,DCB=(RECFM=VB,LRECL=132,BLKSIZE=136)
//ALGPRINT DD   SYSOUT=*,DCB=(RECFM=VB,LRECL=132,BLKSIZE=136)
```

```
//CFGPRINT DD SYSOUT=*,DCB=(RECFM=VB,LRECL=132,BLKSIZE=136)
//SYSOUT DD SYSOUT=*,DCB=(RECFM=VB,LRECL=132,BLKSIZE=136)
//CEEDUMP DD SYSOUT=*,DCB=(RECFM=VB,LRECL=132,BLKSIZE=136)
//SYSERROR DD SYSOUT=*
```

2. SYS1.TCPPARMS(TCPDATA) nos diz que desejamos que o nome do sistema seja o nome do host e que não utilizamos um DNS (servidor de nomes de domínios); todos os nomes serão resolvidos por meio de consulta na tabela de sites.

```
; HOSTNAME especifica o nome do host TCP deste sistema. Se não
; especificado, o HOSTNAME padrão será o nome do nó especificado
; no membro IEFSSNxx PARMLIB.
;
; HOSTNAME
;
; DOMAINORIGIN especifica a origem do domínio que será anexado
; nos nomes dos hosts transmitidos para o resolvidor. Se um nome
; de host contiver algum ponto, então DOMAINORIGIN não será anexado
; ao nome do host.
;
; DOMAINORIGIN RALEIGH.IBM.COM
;
; NSINTERADDR especifica o endereço IP do servidor de nomes.
; LOOPBACK (14.0.0.0) especifica o servidor de nomes local. Se um servidor
; de nomes não será utilizado, então não codifique uma instrução NSINTERADDR.
; (Comente a linha NSINTERADDR abaixo). Isto fará com que todos os nomes
; sejam resolvidos por meio de consulta na tabela de sites.
;
; NSINTERADDR 14.0.0.0
;
; TRACE RESOLVER realizará um rastreamento completo de todas as consultas e
; fará com que as respostas do servidor de nomes ou tabelas de sites sejam
; gravadas no console do usuário. Este comando é destinado somente para
; finalidades de depuração.
;
; TRACE RESOLVER
```

3. No JCL do Resolver vemos que a instrução SETUP DD não é utilizada. Conforme mencionado em “Compreendendo os Resolvedores” na página 89, isto significa que GLOBALTCPIPDATA e outras variáveis não serão utilizadas.

```
//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. Se assumirmos que a variável de ambiente RESOLVER_CONFIG não está configurada, podemos ver na Tabela 9 na página 34 que o Resolvedor tentará utilizar /etc/resolv.conf como arquivo de configuração base.

```
TCPIPJOBNAME TCPIP
DomainOrigin RALEIGH.IBM.COM
HostName CDFMVS08
```

Conforme mencionado em “Ordens de Procura Utilizadas no Ambiente z/OS UNIX” na página 90, o arquivo de base da configuração contém instruções TCPIP.DATA. Se o nome do sistema for CDFMVS08 (TCPDATA informou que o nome do sistema é utilizado como nome do host) podemos ver que /etc/resolv.conf está em sincronismo com SYS1.TCPPARMS(TCPDATA). Não há definições DNS, portanto a consulta à tabela de sites será utilizada.

5. A Tabela 11 na página 94 também informa que, se não temos de fazer nada para utilizar a tabela de conversão ASCII-EBCDIC padrão.
6. Assumindo que o comando **MAKESITE** do TSO não seja utilizado (pode criar as variáveis X_SITE e X_ADDR), /etc/hosts será a tabela de sites utilizada para consulta de nomes.

```
# 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
```

O conteúdo mínimo deste arquivo é a informação sobre o sistema atual. Na amostra acima, definimos cdfmvs08 e cdfmvs08.raleigh.ibm.com como um nome válido para o endereço IP do nosso sistema z/OS.

Se estivéssemos utilizando um DNS (servidor de nomes de domínios), o DNS conteria as informações de /etc/hosts e /etc/resolv.conf e SYS1.TCPPARMS(TCPDATA) teriam instruções que identificam o DNS em nosso sistema.

Para evitar confusão, é aconselhável manter os arquivos de configuração do TCP/IP e do Resolver em sincronismo um com o outro.

Tabela 11. Definições locais disponíveis para o resolvedor

Descrição do Tipo do Arquivo	APIs afetadas	Arquivos do Candidato
Arquivos de Base da Configuração do Resolvedor	Todas as APIs	<ol style="list-style-type: none"> 1. GLOBALTCPIPDATA 2. variável de ambiente RESOLVER_CONFIG 3. /etc/resolv.conf 4. SYSTCPD DD-name 5. userid.TCPIP.DATA 6. jobname.TCPIP.DATA 7. SYS1.TCPPARMS(TCPDATA) 8. DEFAULTTCPIPDATA 9. TCPIP.TCPIP.DATA
Tabelas de Conversão	Todas as APIs	<ol style="list-style-type: none"> 1. variável de ambiente X_XLATE 2. userid.STANDARD.TCPXLBIN 3. jobname.STANDARD.TCPXLBIN 4. hlq.STANDARD.TCPXLBIN 5. Tabela de conversão fornecida pelo resolvedor, membro STANDARD em SEZATCPX
Tabelas do Host 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"> 1. variável de ambiente X_SITE 2. variável de ambiente X_ADDR 3. /etc/hosts 4. userid.HOSTS.xxxxINFO 5. jobname.HOSTS.xxxxINFO 6. hlq.HOSTS.xxxxINFO IPv6 <ol style="list-style-type: none"> 1. GLOBALIPNODES 2. variável de ambiente RESOLVER_IPNODES 3. userid.ETC.IPNODES 4. jobname.ETC.IPNODES 5. hlq.ETC.IPNODES 6. DEFAULTIPNODES 7. /etc/ipnodes

Nota: A tabela acima é uma cópia parcial de uma tabela em *Communications Server: IP Configuration Guide (SC31-8775)*. Consulte esse manual para obter a tabela completa.

Apêndice D. Configurando o INETD

Este apêndice é fornecido para auxiliá-lo com alguns problemas comuns que você pode encontrar ao configurar o INETD ou durante a verificação e/ou modificação de uma configuração existente.

O daemon do INETD fornece gerenciamento de serviço para uma rede IP. Reduz a carga do sistema, invocando outros daemons apenas quando forem necessários e fornecendo vários serviços de Internet simples (como echo) internamente. O INETD lê o arquivo de configuração `inetd.conf` para determinar quais serviços adicionais fornecer. `ETC.SERVICES` é utilizado para vincular os serviços a portas.

inetd.conf

Os serviços que dependem do INETD são definidos em `inetd.conf`, lido pelo INETD no momento da inicialização. O local padrão e o nome de `inetd.conf` é `/etc/inetd.conf`. Um arquivo `inetd.conf` de amostra pode ser encontrado em `/samples/inetd.conf`.

As seguintes regras de sintaxe se aplicam às entradas de `inetd.conf`:

- Os comentários começam com um sinal de sustenido (#) ou ponto-e-vírgula (;) e continuarão até o fim da linha
- As entradas fazem distinção entre maiúsculas e minúsculas
- As entradas fazem distinção de campo, mas não de coluna
- Os campos são separados com um espaço ou caractere tabular
- As entradas podem abranger várias linhas, seguindo estas regras adicionais de sintaxe
 - A divisão deve estar entre duas palavras separadas (separadas por um espaço ou caractere tabular)
 - A linha de continuação deve começar com um espaço ou caractere tabular
 - Não é possível incluir comentários na continuação

Cada entrada consiste em 7 campos posicionais, correspondendo ao formato:

`service_name socket_type protocol wait_flag userid server_program server_program_arguments`

[ip_address:]service_name

`ip_address` é um IP local, seguido por dois pontos (:). Se especificado, o endereço será utilizado, em vez de `INADDR_ANY`, ou o padrão atual. Para solicitar especificamente `INADDR_ANY`, utilize `"*:"`. Se `ip_address` (ou dois-pontos) for especificado sem nenhuma outra entrada na linha, será o padrão para linhas subsequentes, até que um novo padrão seja especificado. `service_name` é um nome de serviço bem conhecido ou definido pelo usuário. O nome especificado deve corresponder a um dos nomes de servidor definidos em `ETC.SERVICES`.

socket_type

`stream` ou `dgram`, para indicar que um soquete de fluxo ou de datagrama é utilizado para o serviço.

protocol[,sndbuf=n[,rcvbuf=n]]

protocol pode ser tcp[4|6] ou udp[4|6], e é utilizado para qualificar o nome do serviço. O nome do serviço e o protocolo devem corresponder a uma entrada em ETC.SERVICES, exceto que "4" ou "6" não deve ser incluído na entrada ETC.SERVICES.

sndbuf e rcvbuf especificam o tamanho dos buffers de envio e recebimento. O tamanho, representado por n, pode estar em bytes, ou um "k" ou "m" pode ser incluído para indicar kilobytes ou megabytes respectivamente. sndbuf e rcvbuf pode ser utilizado na ordem.

wait_flag[.max]

wait ou nowait.wait indica que o daemon é de encadeamento único e outro pedido não será submetido a manutenção até que o primeiro seja concluído. Se nowait for especificado, o INETD emitirá uma aceitação quando um pedido de conexão for recebido no soquete de fluxo. Se a espera for especificada, será responsabilidade do servidor emitir a aceitação, se este for um soquete de fluxo.

max é o número máximo de usuários permitidos para solicitar manutenção em um intervalo de 60 segundos. O padrão é 40. Se for excedido, a porta do serviço será encerrada.

userid[.group]

userid é o ID do usuário em que o daemon bifurcado deve ser executado. Esse ID do usuário pode ser diferente do ID do usuário do INETD. As permissões designadas a este ID do usuário dependem das necessidades do serviço. O ID do usuário do INETD precisa da permissão BPX.DAEMON para alternar o processo bifurcado para este ID do usuário.

O valor group opcional, separado do ID do usuário por um ponto (.), permite que o servidor seja executado com um ID de grupo diferente do padrão para este ID do usuário.

server_program

server_program é o nome do caminho completo do serviço. Por exemplo: /usr/sbin/rlogind é o nome do caminho completo para o comando rlogind.

server_program_arguments

Máximo de 20 argumentos. O primeiro argumento é o nome do servidor.

ETC.SERVICES

O INETD utiliza ETC.SERVICES para mapear números de porta e protocolos para os serviços que deve suportar. Pode ser um conjunto de dados do MVS ou um arquivo z/OS UNIX. Uma amostra é fornecida em SEZAINST(SERVICES), que também está disponível como /usr/lpp/tcpip/samples/services. A ordem de procura para ETC.SERVICES depende do método de inicialização do INETD; z/OS UNIX ou MVS nativo.

As seguintes regras de sintaxe se aplicam à especificação de informações de serviços:

- Um conjunto de dados ETC.SERVICES MVS deve ser fixo ou um bloco fixo em um LRECL entre 56 e 256
- Um arquivo HFS ETC.SERVICES pode ter um comprimento máximo de linha de 256
- Itens de uma linha são separados por espaços ou caracteres tabulares

- Cada serviço é listado em uma única linha
- Um nome de serviço deve começar na primeira posição em uma linha
- O comprimento máximo para o nome do serviço e nome do alias é de 32 caracteres
- No máximo 35 aliases serão reconhecidos
- Os nomes de serviço e de alias fazem distinção entre maiúsculas e minúsculas
- Os comentários começam com um sinal de sustenido (#) ou ponto-e-vírgula (;) e continuarão até o fim da linha

Cada entrada consiste em 4 campos posicionais, correspondendo ao formato:

`service_name port_number/protocol aliases`

service_name

Especifica um nome de serviço bem conhecido ou definido pelo usuário

port_number

Especifica o número da porta do soquete utilizado para o serviço

protocol

Especifica o protocolo de transporte utilizado para o serviço. Os valores válidos são tcp e udp

aliases

Especifica uma lista de nomes de serviços não-oficiais

Ordem de Procura Utilizada no Ambiente z/OS UNIX

A ordem de procura utilizada para acessar ETC.SERVICES em z/OS UNIX é conforme a seguir. A procura finaliza no primeiro arquivo localizado:

1. **/etc/services**
2. **userid.ETC.SERVICES**
userid é o ID do usuário utilizado para iniciar INETD.
3. **hlq.ETC.SERVICES**
hlq representa o valor da instrução DATASETPREFIX especificada no arquivo de base da configuração do resolvidor (se localizado); caso contrário, hlq é TCPIP por padrão.

Ordem de Procura Utilizada no Ambiente MVS Ambiente

A ordem de procura utilizada para acessar ETC.SERVICES no MVS nativo é conforme a seguir. A procura termina no primeiro conjunto de dados encontrado:

1. **//Placa DD SERVICES**
O conjunto de dados alocado para a instrução DD SERVICES é utilizado.
2. **userid.ETC.SERVICES**
userid é o ID do usuário utilizado para iniciar INETD.
3. **jobname.ETC.SERVICES**
jobname é o nome especificado na instrução JOB do JCL para tarefas em lote ou o nome do procedimento para um procedimento iniciado.
4. **hlq.ETC.SERVICES**
hlq representa o valor da instrução DATASETPREFIX especificada no arquivo de base da configuração do resolvidor (se localizado); caso contrário, hlq é TCPIP por padrão.

Nota: O início do INETD por meio de BPXPATCH não resulta na utilização da ordem de procura nativa do MVS, pois BPXBATCH executa o comando inicial no ambiente z/OS UNIX. A ordem de procura nativa do MVS só é utilizada ao iniciar um módulo de carregamento do MVS, como SEZALOAD(FTP).

Definições de Portas PROFILE.TCPIP

Não confunda as definições PORT (ou PORTRANGE) em PROFILE.TCPIP com as portas definidas em ETC.SERVICES, pois essas definições têm finalidades diferentes. As portas definidas em PROFILE.TCPIP são utilizadas pelo TCPIP para ver se a porta está reservada para um determinado serviço. ETC.SERVICES é utilizado pelo INETD para mapear uma porta para um serviço.

Quando o INETD recebe um pedido em uma porta monitorada, bifurca um processo-filho (com o serviço solicitado) chamado `inetdx`, em que `inetd` é o nome da tarefa do INETD (depende do método de inicialização) e `x` é um número de dígito único.

Isso complica a reserva de porta, portanto, se uma porta monitorada do INETD for reservada em PROFILE.TCPIP, é aconselhável utilizar o nome do procedimento JCL iniciado para o Espaço de Endereço de Kernel do z/OS UNIX, a fim de permitir que quase todos os processos sejam ligados à porta. Esse nome é geralmente OMVS, a menos que um nome diferente seja especificado explicitamente no parâmetro `STARTUP_PROC` do membro `parmlib BPXPRMxx`.

A lista a seguir explica como determinar o nome da tarefa, de acordo com o ambiente em que o aplicativo é executado:

- Os aplicativos executados do lote utilizam o nome da tarefa em lote.
- Os aplicativos iniciados do console do operador do MVS utilizam o nome do procedimento iniciado (STC) como o nome da tarefa.
- Os aplicativos executados de um ID de usuário do TSO utilizam o ID de usuário do TSO como o nome da tarefa.
- Os aplicativos executados a partir do shell do z/OS possuem um nome de tarefa registrado no ID do usuário mais um sufixo de um caractere.
- Os usuários autorizados podem executar aplicativos a partir do shell do z/OS e utilizam a variável de ambiente `_BPX_JOBNAME` para definir o nome da tarefa. Nesse caso, o valor especificado para a variável de ambiente é o nome da tarefa.
- O nome do procedimento JCL iniciado para o Espaço de Endereço de Kernel dos Serviços do Sistema UNIX pode ser utilizado para permitir que quase todos os responsáveis pela chamada da API do soquete `bind()` (exceto para usuários da API Pascal) se liguem à porta. Esse nome é geralmente OMVS, a menos que um nome diferente seja especificado explicitamente no parâmetro `STARTUP_PROC` do membro `parmlib BPXPRMxx`.
- Aplicativos z/OS UNIX iniciados pelo INETD utilizam o nome da tarefa do servidor INETD.

Nota: Embora não seja aconselhável fazer isso, as portas definidas em ETC.SERVICES podem ser diferentes do número da porta reservada para o serviço em PROFILE.TCPIP.

/etc/inetd.pid

O processo do INETD cria um arquivo temporário, `/etc/inetd.pid`, que contém o PID (ID do processo) do daemon INETD em execução no momento. Esse valor de PID é utilizado para identificar registros syslog originados do processo INETD e para fornecer o valor PID para comandos que requerem um, como `kill`. Também é utilizado como um mecanismo de bloqueio para impedir que mais de um 1 processo INETD esteja ativo.

Inicialização

A implementação do z/OS UNIX do INETD está localizada, por padrão, em `/usr/sbin/inetd` e suporta 2 parâmetros de inicialização opcionais e não-posicionais:

`/usr/sbin/inetd [-d] [inetd.conf]`

-d Opção de depuração. A opção de depuração é gravada em `stderr`, que pode ser encaminhada para um arquivo pelo daemon `syslogd`. Consulte o *Communications Server IP Configuration Guide (SC31-8775)* para obter mais informações sobre `syslogd`. Observe que, quando iniciado dessa forma, o INETD não bifurcará um processo-filho para iniciar um serviço.

inetd.conf

Arquivo de configuração. O valor padrão é `/etc/inetd.conf`

É aconselhável iniciar o INETD no momento do IPL. A forma mais comum de fazer isso é iniciá-lo no `/etc/rc` ou `/etc/inittab` (z/OS 1.8 e posterior apenas). Pode ser iniciado de uma tarefa ou uma tarefa iniciada utilizando `BPXBATCH` ou de uma sessão de shell de um usuário com autoridade apropriada.

/etc/rc

Quando iniciado a partir do script do shell de inicialização do z/OS UNIX, `/etc/rc`, o INETD utiliza a ordem de procura do z/OS UNIX para localizar `ETC.SERVICES`. Um arquivo `/etc/rc` de amostra é fornecido como `/samples/rc`. Os seguintes comandos de amostra podem ser utilizados para iniciar o INETD:

```
# Start INETD
_BPX_JOBNAME='INETD' /usr/sbin/inetd /etc/inetd.conf &
sleep 5
```

/etc/inittab

O z/OS 1.8 e superior oferecem um método alternativo, `/etc/inittab`, para emissão de comandos durante a inicialização do z/OS UNIX. `/etc/inittab` permite a definição do parâmetro `respawn`, que reinicia o processo automaticamente quando é encerrado (um `WTOR` é enviado para o operador, para um segundo reinício, em 15 minutos). Quando iniciado a partir de `/etc/inittab`, o INETD utiliza a ordem de procura do z/OS UNIX para localizar `ETC.SERVICES`. Um `/etc/inittab` de amostra é fornecido como `/samples/inittab`. O seguinte comando de amostra pode ser utilizado para iniciar o INETD:

```
# Start INETD
inetd::respfrk:/usr/sbin/inetd /etc/inetd.conf
```

Nota: Lembre-se de que o parâmetro `respfrk` utilizado na amostra transferirá o atributo `respawn` para todos os processos bifurcados, incluindo o RSE. Quando o cliente encerrar a conexão, `respawn` a iniciará novamente. O

servidor RSE será encerrado novamente mais tarde, devido ao tempo limite. Consulte *UNIX System Services Planning (GA22-7800)* para aprender mais sobre inittab.

BPXBATCH

O método de inicialização BPXBATCH trabalha para STCs e tarefas do usuário. Observe que o INETD é um processo em segundo plano, portanto a etapa BPXBATCH iniciando o INETD será encerrada em segundos após a inicialização. Quando iniciado pelo BPXBATCH, o INETD utiliza a ordem de procura do z/OS UNIX para localizar ETC.SERVICES. O JCL listado em Figura 11 é um procedimento de amostra para iniciar o INETD (a etapa KILL remove um processo INETD ativo, se houver):

```
//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 e STDENV são padronizados como /dev/null
//* A saída do daemon INETD pode ser controlada pelas configurações de syslogd
//*
```

Figura 11. JCL de Inicialização do INETD

Nota: STDIN, STDOUT e STDERR devem ser arquivos z/OS UNIX, quando alocados. STDENV pode ser um conjunto de dados do MVS ou um arquivo z/OS UNIX. Consulte a *UNIX System Services Command Reference (SA22-7802)* para aprender mais sobre BPXBATCH.

Nota: inetd.conf pode ser um conjunto de dados ou um membro do MVS quando o INETD for iniciado pelo BPXBATCH. Para fazer isso, codifique a instrução PARM como a seguinte amostra (utilize apenas aspas simples (')):

```
// PARM='PGM /usr/sbin/inetd //'SYS1.TCPPARMS(INETCONF)'' &PRM'
```

Sessão de Shell

Quando iniciado a partir de uma sessão de shell, o INETD utiliza a ordem de procura do z/OS UNIX para localizar ETC.SERVICES. Os seguintes comandos de amostra podem ser utilizados (por uma pessoa com autoridade suficiente) para parar e iniciar o INETD (# é o prompt do z/OS UNIX):

```
1. # ps -e | grep inetd
   7 ?          0:00 /usr/sbin/inetd
2. # kill 7
3. # _BPX_JOBNAME='INETD' /usr/sbin/inetd
```

Nota: Esse método não é aconselhável para a inicialização inicial; /etc/rc ou /etc/inittab são mais apropriados, pois são executados quando o z/OS UNIX é inicializado.

Segurança

O INETD é um processo do z/OS UNIX e, portanto, requer definições OMVS válidas no software de segurança para o ID do usuário associado ao INETD. UID, HOME e PROGRAM devem ser definidos para o ID do usuário, juntamente com o GID para o grupo padrão do usuário. Se o INETD for iniciado pelo /etc/rc ou pelo /etc/inittab, o ID do usuário será herdado do kernel z/OS UNIX, padrão OMVSKERN.

```
ADDGROUP OMVSGRP OMVS(GID(1))
ADDUSER OMVSKERN DFLTGRP(OMVSGRP) NOPASSWORD +
      OMVS(UID(0) HOME('/') PROGRAM('/bin/sh'))
```

O INETD é um daemon que requer acesso a funções como setuid(). Portanto, o ID do usuário utilizado para iniciar o INETD requer acesso de LEITURA ao perfil BPX.DAEMON na classe FACILITY. Se esse perfil não estiver definido, UID 0 será obrigatório.

```
PERMIT BPX.DAEMON CLASS(FACILITY) ACCESS(READ) ID(OMVSKERN)
```

O ID do usuário do INETD requer permissão de EXECUÇÃO para o programa inetd (/usr/sbin/inetd), acesso de LEITURA para os arquivos inetd.conf e ETC.SERVICES e acesso de GRAVAÇÃO para /etc/inetd.pid. Se você deseja executar o INETD sem o UID 0, pode conceder acesso de CONTROLE ao perfil SUPERUSER.FILESYS na classe UNIXPRIV para fornecer as permissões necessárias para os arquivos z/OS UNIX.

Os programas que requerem autoridade de daemon devem ser controlados por programa, se BPX.DAEMON estiver definido na classe FACILITY. Isso já é feito para o programa INETD padrão (/usr/sbin/inetd), mas deve ser definido manualmente se você utilizar uma cópia ou uma versão customizada. Utilize o comando **extattr +p** para tornar um arquivo z/OS UNIX controlado por programa. Utilize a classe RACF PROGRAM para tornar um módulo de carregamento MVS controlado por programa.

Os programadores de sistema que precisam reiniciar o INETD em sua sessão de shell iniciarão o INETD utilizando suas permissões. Portanto, devem ter a mesma lista de permissões que o ID do usuário do INETD regular. Além disso, também precisam de permissões para listar e parar o processo do INETD. Isso pode ser feito de várias maneiras.

- UID 0
Isso não é recomendado para IDs de usuário “humanos”, pois não há restrições relacionadas ao z/OS UNIX.
- Acesso de LEITURA ao perfil BPX.SUPERUSER na classe FACILITY
Permite que o usuário se torne o UID 0 por meio do comando **su**. Essa é a configuração recomendada.
- acesso a perfis individuais que abrangem as permissões necessárias
 - Acesso de LEITURA a SUPERUSER.PROCESS.GETPSENT na classe UNIXPRIV (para o comando **ps**)
 - Acesso de LEITURA a SUPERUSER.PROCESS.KILL na classe UNIXPRIV (para o comando **kill**)
 - Acesso de LEITURA a BPX.JOBNAME na classe FACILITY (para a variável de ambiente **_BPX_JOBNAME**)

Consulte a *UNIX System Services Command Reference (SA22-7802)* para aprender mais sobre os comandos **extattr** e **su**. Consulte *UNIX System Services Planning*

(GA22-7800) para aprender mais sobre a classe UNIXPRIV e os perfis BPX.* na classe FACILITY. Consulte o *Security Server RACF Security Administrator's Guide* (SA22-7683) para obter mais informações sobre as definições do segmento OMVS e a classe PROGRAM.

Requisitos do Developer for System z

O Developer for System z depende de INETD para configuração da conexão de host do cliente. Também impõe requisitos adicionais além da configuração do INETD descrita acima.

INETD

As configurações ambientais do INETD, informadas ao iniciar um processo, e as permissões para o ID do usuário do INETD, devem ser definidas adequadamente para que o INETD inicie o servidor RSE.

- Se INETD for iniciado pelo JCL utilizando BPXBATCH, o tamanho da região deve ser 0.
- Se o INETD for iniciado a partir de uma sessão de shell do TSO/OMVS, o tamanho da região TSO deve ser 2096128 ou maior.
- Se o INETD for iniciado pelo `/etc/rc` ou `/etc/inittab`, o tamanho da região `SYS1.PROCLIB(BPX0INIT)` será utilizado, sendo 0 como padrão.
- Acesso de LEITURA e EXECUÇÃO aos diretórios de instalação do Developer for System z, padrão `/usr/lpp/wd4z/*`

Daemon do RSE

O daemon RSE que é iniciado pelo INETD quando um cliente se conecta à porta 4035 é utilizado para desempenhar autenticação, iniciar o servidor RSE e retornar o número da porta para comunicação com o cliente. Para fazer isso, o ID do usuário designado ao daemon RSE (em `inetd.conf`) requer as seguintes permissões:

- Definições OMVS válidas no software de segurança; UID, HOME e PROGRAM devem ser definidos, juntamente com o GID para o grupo padrão do usuário
- Acesso de LEITURA ao perfil BPX.DAEMON na classe FACILITY
- Acesso de LEITURA e EXECUÇÃO aos diretórios de instalação do Developer for System z, padrão `/usr/lpp/wd4z/*`
- Acesso de LEITURA e EXECUÇÃO ao diretório de configuração do Developer for System z, padrão `/usr/lpp/wd4z/rse/lib/`, mas é aconselhável utilizar um diretório diferente, como o `/etc/wd4z/`

Apêndice E. Configurando o SSL

Este apêndice é fornecido para auxiliá-lo com alguns problemas comuns que você pode encontrar ao configurar o SSL (Secure Socket Layer) ou durante a verificação e/ou modificação de uma configuração existente.

Comunicação segura significa garantir que seu parceiro de comunicação seja o que alega ser e transmitir informações de forma que dificulte a interceptação e leitura dos dados por terceiros. SSL (Secure Sockets Layer) oferece essa capacidade em uma rede TCP/IP. Funciona utilizando certificados digitais para se identificar e um protocolo de chave pública para criptografar a comunicação. Consulte o *Security Server RACF Security Administrator's Guide (SA22-7683)* para obter mais informações sobre certificados digitais e sobre o protocolo de chave pública utilizado pelo SSL.

As ações necessárias para configurar as comunicações SSL para o Developer for System z variam de site para site, dependendo das necessidades exatas, do método de comunicação RSE utilizado e do que já está disponível no site.

Neste apêndice, clonaremos as definições RSE atuais, para que tenhamos uma 2ª conexão que utilizará SSL. O método de comunicação de REXEC e de daemon será configurado para o SSL. Também criaremos nossos próprios certificados de segurança a serem utilizados pelas diferentes partes da conexão RSE. Tudo isso será feito nas seguintes etapas:

1. Clonar a configuração RSE existente
2. Determinar o(s) arquivo(s) de chaves a ser(em) utilizado(s)
3. Criar um armazenamento de chaves com keytool
4. Criar um banco de dados de chaves (daemon apenas), com o RACF ou gskkyman
5. Ativar o SSL, atualizando ssl.properties
6. Testa a conexão

Nota: Consulte o white paper *Setting up SSL for RSE Communication (SC23-5816)* na biblioteca na Internet do Developer for System z, <http://www-306.ibm.com/software/awdtools/devzseries/library/>, para obter mais informações sobre a utilização de um certificado assinado por uma CA (Autoridade de Certificação) confiável ou configurando sua própria CA.

Todo este apêndice apresenta uma convenção de nomenclatura uniforme:

- Certificado: wd4zrse
- Armazenamento de chaves e de certificados: wd4zssl.*
- Senha: rsessl

A maioria das tarefas descritas abaixo espera que você esteja ativo no z/OS UNIX. Isso pode ser feito emitindo o comando do TSO **OMVS**. Utilize o comando **exit** para retornar ao TSO.

Clonar a Configuração RSE Existente

Nesta etapa, uma nova instância do servidor RSE e daemon RSE é criada para execução paralela com a(s) existente(s). Dessa forma, o teste do SSL não impedirá operações normais. Conforme aconselhável em “Salvando o Arquivo de Configuração rsed.envvars em um Outro Diretório” na página 34, os comandos de amostra a seguir esperam que os arquivos de configuração estejam em /etc/wd4z/

```
$ cd /etc/wd4z
$ mkdir ssl
$ cp * ssl
cp: FSUM6254 "ssl" é um diretório (não copiado)
$ 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 utilizando SSL

incluir o serviço rsessl, utilizando a porta 4047
# oedit /etc/inetd.conf

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

incluir o daemon rsessl, utilizando o diretório de configuração /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
```

Os comandos listados acima criam um subdiretório chamado ssl e o preenchem com os arquivos de configuração obrigatórios. Nenhuma alteração na configuração precisa ser feita (ainda). Podemos compartilhar o diretório de instalação e os componentes do MVS, pois não são específicos ao SSL. No entanto, um novo daemon (rsessl) deve ser definido utilizando os novos arquivos de configuração. A porta 4047 é designada ao novo daemon.

Consulte Capítulo 4, “Ativando Componentes do z/OS UNIX para o Developer for System z”, na página 33 para obter mais informações sobre as ações mostradas acima.

Determinar o(s) Arquivo(s) de Chaves a Ser(em) Utilizado(s)

Os certificados de identidade e as chaves de criptografia/descriptografia utilizadas pelo SSL são armazenados em um arquivo de chaves. Existem diferentes implementações deste arquivo de chaves, dependendo do tipo de aplicativo.

O daemon RSE é um aplicativo SSL do Sistema e utiliza um arquivo de banco de dados de chaves. Esse arquivo de banco de dados de chaves pode ser um arquivo físico criado por gskkyman ou um anel de chave gerenciado pelo software de segurança (por exemplo, RACF). O servidor RSE (iniciado pelo daemon ou

REXEC) é um aplicativo SSL Java e utiliza um arquivo de armazenamento de chaves criado pelo keytool. Atualmente, o RACF não tem suporte out-of-the-box para o SSL Java.

Assim, para se conectar via REXEC, tudo o que é necessário é o arquivo de armazenamento de chaves:

- armazenamento de chaves (keytool)

Para se conectar pelo daemon, precisamos do armazenamento de chaves e do arquivo do banco de dados de chaves:

- armazenamento de chaves (keytool)
- banco de dados de chaves (RACF ou gskkyman)

Consulte *Security Server RACF Security Administrator's Guide (SA22-7683)* para obter informações sobre o RACF e certificados digitais. A documentação do gskkyman pode ser encontrada em *System SSL Programming (SC24-5901)*. A documentação do keytool está disponível no endereço <http://java.sun.com/j2se/1.4.2/docs/tooldocs/solaris/keytool.html>.

Criar um Armazenamento de Chaves com keytool

"keytool -genkey" Gera um par de chaves (uma chave pública e chave privada associada). Então, agrupa a chave pública em um certificado auto-assinado X.509 v1, armazenado como uma cadeia de certificados de elemento único. Essa cadeia de certificados e a chave privada são armazenadas como uma entrada (identificada por um alias) em um arquivo de armazenamento de chaves (novo).

Nota: Java deve ser incluído nos diretórios de procura de comando. A seguinte instrução pode ser necessária para executar o keytool (/usr/lpp/java/J1.4 é o diretório em que o Java está instalado): `PATH=$PATH:/usr/lpp/java/J1.4/bin`

Todas as informações podem ser transmitidas como um parâmetro, mas, devido ao comprimento da linha de comandos, é necessária alguma interatividade.

```
$ keytool -genkey -alias wd4zrse -validity 3650 -keystore wd4zssl.jks -storepass rsessl -keypass rsessl
```

```
Qual é o seu nome e sobrenome?
```

```
[Desconhecido]: wd4z rse ssl
```

```
Qual é o nome de sua unidade organizacional?
```

```
[Desconhecido]: wd4z
```

```
Qual é o nome de sua organização?
```

```
[Desconhecido]: IBM
```

```
Qual é o nome de sua cidade ou localidade?
```

```
[Desconhecido]: Raleigh
```

```
Qual é o nome de seu estado ou província?
```

```
[Desconhecido]: NC
```

```
Qual é o código do país de duas letras para esta unidade?
```

```
[Desconhecido]: US
```

```
É CN=wd4z rse ssl, OU=wd4z, O=IBM, L=Raleigh, ST=NC, C=US correto? (digite "sim" ou "não")
```

```
[não]: sim
```

```
$ ls
```

```
rsecomm.properties  server.zseries      ssl.properties  
rsed.envvars        setup.env.zseries   wd4zssl.jks
```

O certificado auto-assinado criado acima é válido durante aproximadamente 10 anos (não contando o dia 29 de fevereiro). É armazenado em /etc/wd4z/ssl/

wd4zssl.jks, utilizando wd4zrse. Sua senha (rsessl) é idêntica à senha do armazenamento de chaves, que é um requisito para o RSE.

O resultado pode ser verificado com a opção -list:

```
$ keytool -list -alias wd4zrse -keystore wd4zssl.jks -storepass rsessl -v
Nome do alias: wd4zrse
Data de criação: 24 de maio de 2007
Tipo de entrada: keyEntry
Comprimento da cadeia de certificados: 1
Certificado 1:
Proprietário: CN=wd4z rse ssl, OU=wd4z, O=IBM, L=Raleigh, ST=NC, C=US
Emissor: CN=wd4z rse ssl, OU=wd4z, O=IBM, L=Raleigh, ST=NC, C=US
Número serial: 46562b2b
Válido de: 24/5/07 14h17 até: 21/5/17 14h17
Impressões digitais do certificado:
    MD5: 9D:6D:F1:97:1E:AD:5D:B1:F7:14:16:4D:9B:1D:28:80
    SHA1: B5:E2:31:F5:B0:E8:9D:01:AD:2D:E6:82:4A:E0:B1:5E:12:CB:10:1C
```

Criar um Banco de Dados de Chaves (daemon apenas)

Conforme mencionado anteriormente, o daemon é um aplicativo SSL do Sistema e utiliza um banco de dados de chaves. Pode ser um arquivo físico criado por gskkyman ou um anel de chave do RACF. Os anéis de chave do RACF são o método preferencial para o gerenciamento de chaves privadas e certificados para o SSL do Sistema.

Nota: O SSL do Sistema utiliza o ICSF (Integrated Cryptographic Service Facility), se estiver disponível. O ICSF oferece suporte criptográfico a hardware que será utilizado, em vez de algoritmos de software do SSL do Sistema. Consulte *System SSL Programming (SC24-5901)* para obter mais informações sobre isso.

Criar um Anel de Chave com o RACF

Não execute esta etapa se utilizar o gskkyman para SSL do Sistema.

O comando **RACDCERT** instala e mantém chaves privadas e certificados no RACF. O RACF suporta várias chaves privadas e certificados para serem gerenciados como um grupo. Esses grupos são chamados anéis de chave.

Consulte *Security Server RACF Command Language Reference (SA22-7687)* para obter detalhes sobre o comando **RACDCERT**.

```
RDEFINE FACILITY IRR.DIGTCERT.LIST UACC(NONE)
RDEFINE FACILITY IRR.DIGTCERT.LISTRING UACC(NONE)
PERMIT IRR.DIGTCERT.LIST CLASS(FACILITY) ACCESS(READ) ID(omvskern)
PERMIT IRR.DIGTCERT.LISTRING CLASS(FACILITY) ACCESS(READ) ID(omvskern)
SETROPTS RACLIST(FACILITY) REFRESH
```

```
RACDCERT ID(omvskern) GENCERT SUBJECTSDN(CN('wd4z rse ssl') +
OU('wd4z') O('IBM') L('Raleigh') SP('NC') C('US')) +
NOTAFTER(DATE(2017-05-21)) WITHLABEL('wd4zrse') KEYUSAGE(HANDSHAKE)
```

```
RACDCERT ID(omvskern) ADDRING(wd4zssl.racf)
RACDCERT ID(omvskern) CONNECT(LABEL('wd4zrse') RING(wd4zssl.racf) +
DEFAULT USAGE(PERSONAL))
```

A amostra acima começa criando os perfis necessários e permitindo que o ID do usuário OMVSKERN acesse os anéis de chave. O ID do usuário utilizado deve corresponder ao ID do usuário codificado em /etc/inetd.conf para o daemon SSL

RSE. A próxima etapa é criar um novo certificado auto-assinado com a etiqueta wd4zrse. Não é necessária senha. Esse certificado é, então, incluído em um anel de chave recém-criado (wd4zssl.racf). Assim como com o certificado, não é necessária senha para o anel de chave. A vida útil do certificado corresponde a um criado com keytool.

O resultado pode ser verificado com a opção list:

RACDCERT ID(omvskern) LIST
Informações do certificado digital para o usuário OMVSKERN:

Etiqueta: wd4zrse
ID do Certificado: 2QjW10Xi0sXZ1aaEqZmihUBA
Status: TRUST
Data de Início: 24/05/2007 0h
Data de Encerramento: 21/05/2017 23h59min59s
Número de Série:
>00<
Nome do Emissor:
>CN=wd4z rse ssl.OU=wd4z.O=IBM.L=Raleigh.SP=NC.C=US<
Nome do Assunto:
>CN=wd4z rse ssl.OU=wd4z.O=IBM.L=Raleigh.SP=NC.C=US<
Tipo de Chave Privada: Não-ICSF
Tamanho da Chave Privada: 1024
Associações de Anel:
Proprietário do Anel: OMVSKERN
Anel:
>wd4zssl.racf<

Criar um Banco de Dados de Chaves com gskkyman

Não execute esta etapa se utilizar o RACF para SSL do Sistema.

gskkyman é um programa z/OS UNIX baseado em shell e orientado por menus, que cria, preenche e gerencia um arquivo z/OS UNIX que contém chaves privadas, pedidos de certificado e certificados. Esse arquivo z/OS UNIX é chamado de banco de dados de chaves.

Nota: As seguintes instruções podem ser necessárias para configurar o ambiente para gskkyman. Consulte System SSL Programming (SC24-5901) para obter mais informações sobre isso.

```
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

Menu do Banco de Dados

- 1 - Criar novo banco de dados
- 2 - Abrir banco de dados
- 3 - Alterar senha do banco de dados
- 4 - Alterar comprimento do registro do banco de dados
- 5 - Excluir banco de dados
- 6 - Criar arquivo de parâmetro-chave
- 0 - Sair do programa

Digite o número da opção: 1

Digite o nome do banco de dados-chave (pressione ENTER para retornar ao menu): wd4zssl.kdb

Digite a senha do banco de dados (pressione ENTER para retornar ao menu): rsessl

Digite novamente a senha do banco de dados: rsessl

Digite a expiração da senha em dias (pressione ENTER para nenhuma expiração):

Digite o comprimento do registro do banco de dados (pressione ENTER para utilizar 2500):

Banco de dados-chave /etc/wd4z/ssl/wd4zssl.kdb criado.

Pressione ENTER para continuar.

Menu do Key Management

Banco de dados: /etc/wd4z/ssl/wd4zssl.kdb

- 1 - Gerenciar chaves e certificados
- 2 - Gerenciar certificados
- 3 - Gerenciar pedidos de certificado
- 4 - Criar novo pedido de certificado
- 5 - Receber certificado solicitado ou um certificado de renovação
- 6 - Criar um certificado auto-assinado
- 7 - Importar um certificado
- 8 - Importar um certificado e uma chave privada
- 9 - Mostrar a chave padrão
- 10 - Armazenar senha do banco de dados
- 11 - Mostrar comprimento do registro do banco de dados

- 0 - Sair do programa

Digite o número da opção (pressione ENTER para retornar ao menu anterior): 6

Tipo de Certificado

- 1 - Certificado CA com chave RSA de 1024 bits
- 2 - Certificado CA com chave RSA de 2048 bits
- 3 - Certificado CA com chave RSA de 4096 bits
- 4 - Certificado CA com chave DSA de 1024 bits
- 5 - Certificado do usuário ou do servidor com chave RSA de 1024 bits
- 6 - Certificado do usuário ou do servidor com chave RSA de 2048 bits
- 7 - Certificado do usuário ou do servidor com chave RSA de 4096 bits
- 8 - Certificado do usuário ou do servidor com chave DSA de 1024 bits

Selecione o tipo de certificado (pressione ENTER para retornar ao menu): 5

Digite a etiqueta (pressione ENTER para retornar ao menu): wd4zrse

Digite o nome do assunto para o certificado

Nome comum (necessário): wd4z rse ssl

Unidade organizacional (opcional): wd4z

Organização (necessário): IBM

Cidade/Localidade (opcional): Raleigh

Estado/Província (opcional): NC

País/Região (2 caracteres - necessário): EUA

Digite o número de dias que o certificado permanecerá válido (padrão 365): 3650

Digite 1 para especificar nomes alternativos de assunto ou 0 para continuar: 0

Aguarde

Certificado criado.

Pressione ENTER para continuar.

Menu do Key Management

Banco de dados: /etc/wd4z/ssl/wd4zssl.kdb

- 1 - Gerenciar chaves e certificados
- 2 - Gerenciar certificados
- 3 - Gerenciar pedidos de certificado
- 4 - Criar novo pedido de certificado
- 5 - Receber certificado solicitado ou um certificado de renovação
- 6 - Criar um certificado auto-assinado
- 7 - Importar um certificado
- 8 - Importar um certificado e uma chave privada
- 9 - Mostrar a chave padrão
- 10 - Armazenar senha do banco de dados

11 - Mostrar comprimento do registro do banco de dados

0 - Sair do programa

Digite o número da opção (pressione ENTER para retornar ao menu anterior): 0

```
$ ls -l
total 152
-rwxr-xr-x 1 IBMUSER SYS1      333 May 24 13:52 rsecomm.properties
-rwxr-xr-x 1 IBMUSER SYS1    6067 May 24 13:52 rsed.envvars
-rwxr-xr-x 1 IBMUSER SYS1     332 May 24 13:52 server.zseries
-rwxr-xr-x 1 IBMUSER SYS1     645 May 24 13:52 setup.env.zseries
-rwxr-xr-x 1 IBMUSER SYS1     638 May 24 13:52 ssl.properties
-rw-r--r-- 1 IBMUSER SYS1    1224 May 24 14:17 wd4zssl.jks
-rw----- 1 IBMUSER SYS1   35080 May 24 14:24 wd4zssl.kdb
-rw----- 1 IBMUSER SYS1      80 May 24 14:24 wd4zssl.rdb
$ chmod 644 wd4zssl.kdb
$ chmod 644 wd4zssl.rdb
$ ls -l
total 152
-rwxr-xr-x 1 IBMUSER SYS1      333 May 24 13:52 rsecomm.properties
-rwxr-xr-x 1 IBMUSER SYS1    6067 May 24 13:52 rsed.envvars
-rwxr-xr-x 1 IBMUSER SYS1     332 May 24 13:52 server.zseries
-rwxr-xr-x 1 IBMUSER SYS1     645 May 24 13:52 setup.env.zseries
-rwxr-xr-x 1 IBMUSER SYS1     638 May 24 13:52 ssl.properties
-rw-r--r-- 1 IBMUSER SYS1    1224 May 24 14:17 wd4zssl.jks
-rw-r--r-- 1 IBMUSER SYS1   35080 May 24 14:24 wd4zssl.kdb
-rw-r--r-- 1 IBMUSER SYS1      80 May 24 14:24 wd4zssl.rdb
```

A amostra acima inicia criando um banco de dados de chaves chamado `wd4zssl.kdb` com a senha `rsessl`. Quando o banco de dados existe, é preenchido criando um novo certificado auto-assinado armazenado com a etiqueta `wd4zrse` e com a mesma senha (`rsessl`) que a utilizada para o banco de dados de chaves (esse é um requisito do RSE).

`gskkyman` aloca o banco de dados de chaves com uma máscara de bits de permissão 600 (muito segura) (só o proprietário tem acesso). A menos que o daemon utilize o mesmo ID do usuário que o criador do banco de dados de chaves, as permissões devem ser configuradas menos restritivas. 640 (o proprietário tem leitura/gravação, o grupo do proprietário tem leitura) ou 644 (o proprietário tem leitura/gravação, todos têm leitura) são máscaras úteis para o comando **`chmod`**.

O resultado pode ser verificado selecionando a opção **Mostrar informações do certificado** no submenu **Gerenciar chaves e certificados**:

```
$ gskkyman
```

Menu do Banco de Dados

- 1 - Criar novo banco de dados
- 2 - Abrir banco de dados
- 3 - Alterar senha do banco de dados
- 4 - Alterar comprimento do registro do banco de dados
- 5 - Excluir banco de dados
- 6 - Criar arquivo de parâmetro-chave

0 - Sair do programa

Digite o número da opção: 2

Digite o nome do banco de dados-chave (pressione ENTER para retornar ao menu): `wd4zssl.kdb`

Digite a senha do banco de dados (pressione ENTER para retornar ao menu): `rsessl`

Menu do Key Management

Banco de dados: `/etc/wd4z/ssl/wd4zssl.kdb`

- 1 - Gerenciar chaves e certificados
- 2 - Gerenciar certificados
- 3 - Gerenciar pedidos de certificado
- 4 - Criar novo pedido de certificado
- 5 - Receber certificado solicitado ou um certificado de renovação
- 6 - Criar um certificado auto-assinado
- 7 - Importar um certificado
- 8 - Importar um certificado e uma chave privada
- 9 - Mostrar a chave padrão
- 10 - Armazenar senha do banco de dados
- 11 - Mostrar comprimento do registro do banco de dados
- 0 - Sair do programa

Digite o número da opção (pressione ENTER para retornar ao menu anterior): 1

Lista de Chaves e Certificados

Banco de dados: /etc/wd4z/ssl/wd4zssl.kdb

- 1 - wd4zrse
- 0 - Retornar ao menu de seleção

Digite o número da etiqueta (ENTER para retornar ao menu de seleção, p para lista anterior: 1

Menu Chave e Certificado

Etiqueta: wd4zrse

- 1 - Mostrar informações do certificado
- 2 - Mostrar informações chave
- 3 - Definir chave como padrão
- 4 - Definir status de confiança do certificado
- 5 - Copiar certificado e chave para outro banco de dados
- 6 - Exportar certificado para um arquivo
- 7 - Exportar certificado e chave para um arquivo
- 8 - Excluir certificado e chave
- 9 - Alterar etiqueta
- 10 - Criar um certificado assinado e uma chave
- 11 - Criar um pedido de renovação do certificado
- 0 - Sair do programa

Digite o número da opção (pressione ENTER para retornar ao menu anterior): 1

Informações do Certificado

Etiqueta: wd4zrse

ID do Registro: 14

ID do Registro do Emissor: 14

Confiável: Sim

Versão: 3

Número serial: 45356379000ac997

Nome do emissor: wd4z rse ssl
wd4z

O IBM
Rodovia SP 101 km 09
CEP 13185-900
Hortolândia, SP
NCUS

Nome do assunto: wd4z rse ssl
wd4z

O IBM
Rodovia SP 101 km 09
CEP 13185-900
Hortolândia, SP

```

NCUS
    Data de efetivação: 24/05/2007
    Data de expiração: 21/05/2017
    Algoritmo de chave pública: rsaEncryption
    Tamanho da chave pública: 1024
    Algoritmo de assinatura: sha1WithRsaEncryption
    ID exclusivo do emissor: Nenhum
    ID exclusivo do assunto: Nenhum
    Número de extensões: 3

Digite 1 para exibir extensões, 0 para retornar ao menu: 0

    Menu Chave e Certificado

    Etiqueta: wd4zrse

    1 - Mostrar informações do certificado
    2 - Mostrar informações chave
    3 - Definir chave como padrão
    4 - Definir status de confiança do certificado
    5 - Copiar certificado e chave para outro banco de dados
    6 - Exportar certificado para um arquivo
    7 - Exportar certificado e chave para um arquivo
    8 - Excluir certificado e chave
    9 - Alterar etiqueta
    10 - Criar um certificado assinado e uma chave
    11 - Criar um pedido de renovação do certificado

    0 - Sair do programa

Digite o número da opção (pressione ENTER para retornar ao menu anterior): 0

```

Ativar o SSL, Atualizando ssl.properties

Observe que os certificados existem, então o RSE pode começar a utilizar o SSL. Dependendo das definições escolhidas nas etapas acima, diferentes valores devem ser fornecidos em `ssl.properties`.

```
$ oedit ssl.properties
```

```
enable_ssl=true
```

Os valores válidos são `true` e `false` (padrão).

```
daemon_keydb_file=wd4zssl.racf
```

Nome do banco de dados de chaves gskkyman ou nome do anel de chave do RACF. Só necessário para uso do daemon.

```
daemon_keydb_password=
```

Senha do banco de dados de chaves gskkyman ou branco para anel de chaves do RACF. Só necessário para uso do daemon.

```
daemon_key_label=wd4zrse
```

Etiqueta gskkyman/RACF utilizada, se não estiver definida como padrão. O comentário deve ser removido, se o padrão for utilizado. Só necessário para uso do daemon.

```
server_keystore_file=wd4zssl.jks
```

Nome do armazenamento de chaves keytool.

```
server_keystore_password=rsessl
```

Senha do armazenamento de chaves keytool.

Testar a Conexão

A configuração do host SSL agora está concluída e pode ser testada conectando-se ao cliente do Developer for System z. Como criamos uma nova configuração para uso pelo SSL (clonando a existente), uma nova conexão deve ser definida, utilizando as seguintes características:

- REXEC: utilizando o caminho `/etc/wd4z/ssl`
- daemon: utilizando a porta 4047

Nota: Para executar um aplicativo SSL do Sistema (conexão daemon), o `SYS1.SIEALNKE` deve estar em `LINKLIST` ou em `STEPLIB`. Se você preferir o método `STEPLIB`, inclua a seguinte instrução no fim de `rsed.envvars`. No entanto, lembre-se de que a utilização de `STEPLIB` em `z/OS UNIX` tem um impacto negativo no desempenho, conforme descrito em “Evite o Uso de `STEPLIB`” na página 65.

- Se a última diretiva `STEPLIB` definida anteriormente em `rsed.envvars` for igual a `STEPLIB=NONE`
`STEPLIB=SYS1.SIEALNKE`
- Se a última diretiva `STEPLIB` definida anteriormente em `rsed.envvars` não for igual a `STEPLIB=NONE`
`STEPLIB=$STEPLIB:SYS1.SIEALNKE`

Na conexão, o host e o cliente começarão com alguma troca para configurar um caminho seguro. Parte dessa troca é o intercâmbio de certificados. Se o cliente do Developer for System z não reconhecer o certificado do host, perguntará ao usuário se esse certificado pode ser confiável.

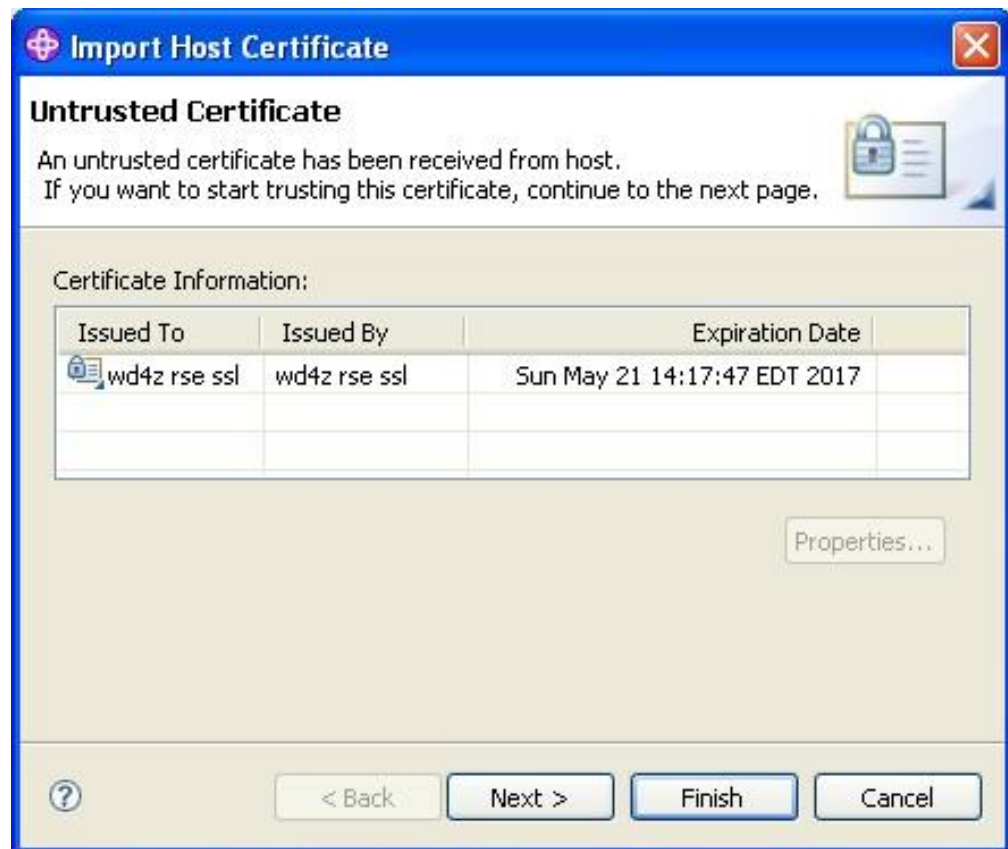


Figura 12. Importar Certificado do Host

Clicando no botão Concluir, o usuário pode aceitar esse certificado como confiável; depois disso, a inicialização da conexão continuará.

Nota: A conexão do daemon utiliza 2 locais de certificado (SSL do Sistema e SSL Java), resultando em 2 certificados diferentes e, assim, 2 confirmações.

Quando um certificado é conhecido do cliente, esse diálogo não é mostrado novamente. A lista de certificados confiáveis pode ser gerenciada selecionando **Janela > Preferências... > Sistemas Remotos > SSL**, que mostra o seguinte diálogo:

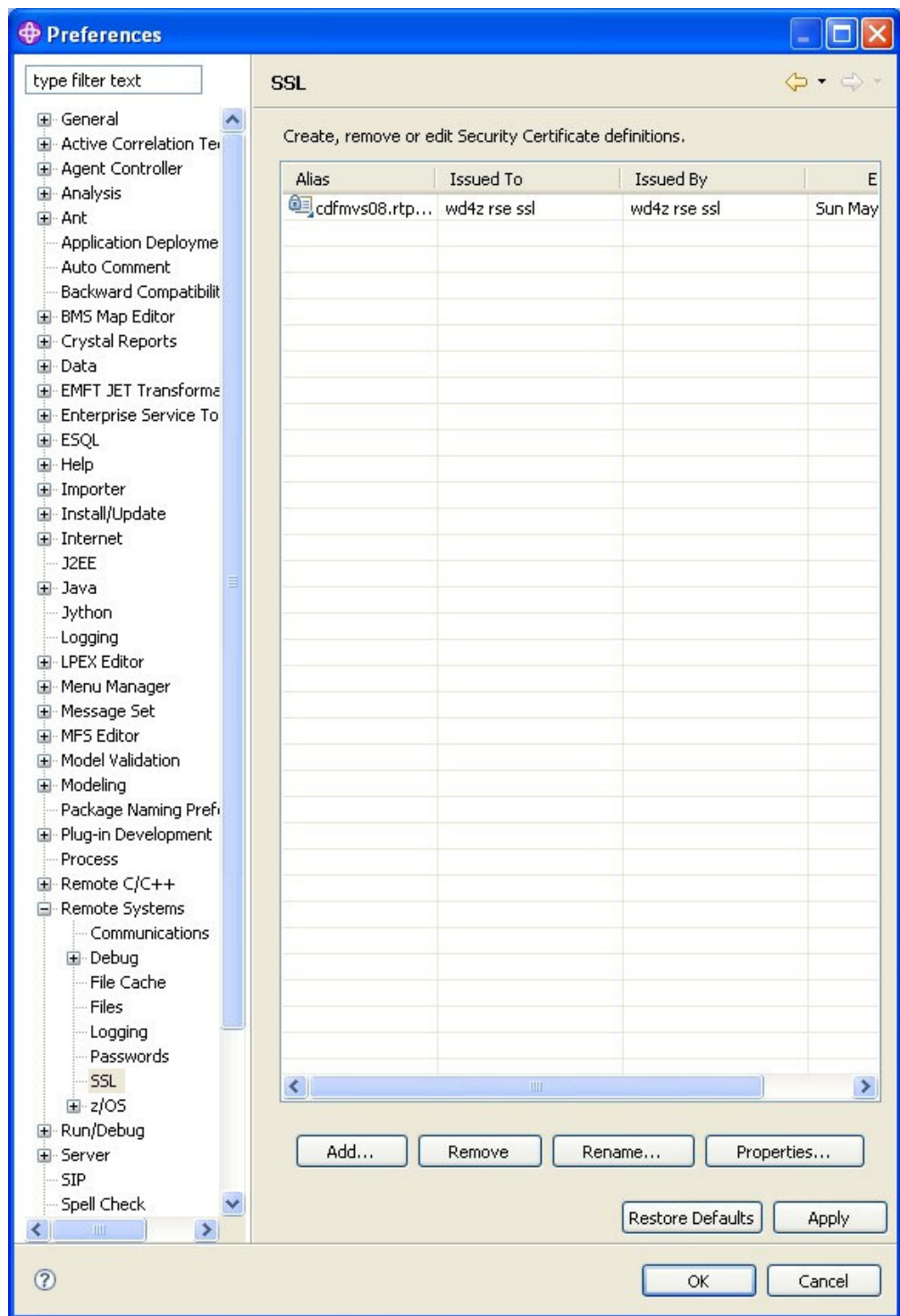


Figura 13. Preferências

Se a comunicação SSL falhar, o cliente retornará uma mensagem de erro. Mais informações estão disponíveis em arquivos de log diferentes (home/.eclipse/RSE/USERID/* e /tmp/rsdaemon.log), conforme descrito em “Criação de Logs do RSE” na página 76.

Apêndice F. Configurando o APPC

Este apêndice é fornecido para auxiliá-lo com alguns problemas comuns que você pode encontrar ao configurar o APPC (Advanced Program-to-Program Communication) ou durante a verificação e/ou modificação de uma configuração existente.

Consulte *MVS Planning: APPC/MVS Management (SA22-7599)* e *MVS Initialization and Tuning Reference (SA22-7592)* para obter informações adicionais sobre o gerenciamento do APPC e os membros parmlib discutidos abaixo.

Observe que isso não abrange uma configuração completa do APPC, apenas destaca alguns aspectos principais que podem ser aplicáveis a seu site.

O membro SYS1.SAMPLIB(ATBALL) contém uma lista e descrições de todos os membros relacionados ao APPC (amostra) em SYS1.SAMPLIB.

VSAM

O APPC/MVS armazena os dados de configuração nos membros SYS1.PARMLIB e em dois conjuntos de dados VSAM:

- O conjunto de dados VSAM de TP (Transaction Program) (nome padrão SYS1.APPCTP) contém informações de planejamento e de segurança para programas z/OS.
- O conjunto de dados VSAM de SI (Side Information) (nome padrão SYS1.APPCSI) contém a conversão de nomes de destino simbólicos utilizados pelos servidores TP e APPC/MVS do z/OS locais.

Um TP é um programa aplicativo que utiliza o APPC para se comunicar com um TP no mesmo sistema, ou em outro, para acessar recursos. A configuração do APPC para o Developer for System z ativa um novo TP chamado FEKFRSRV, mencionado como o serviço Comandos do TSO.

A tarefa de amostra listada na figura 14 é uma concatenação dos membros de amostra SYS1.SAMPLIB(ATBTPVSM) e SYS1.SAMPLIB(ATBSIVSM) e pode ser utilizada para definir os VSAMs do APPC.

```

//APPCVSAM JOB <parâmetros da tarefa>
//*
/* CUIDADO: Isso não é um procedimento JCL nem uma tarefa completa.
/* Antes de utilizar esta amostra, você deverá fazer as seguintes
/* modificações:
/* 1. Altere os parâmetros da tarefa de acordo com os requisitos do sistema.
/* 2. Altere ***** para o volume que conterá os VSAMs do APPC.
//*
//TP      EXEC PGM=IDCAMS
//SYSPRINT DD  SYSOUT=*
//SYSIN    DD  *
        DEFINE CLUSTER (NAME(SYS1.APPCTP) -
                        VOLUME(*****)) -
                        INDEXED REUSE -
                        SHAREOPTIONS(3 3) -
                        RECORDSIZE(3824 7024) -
                        KEYS(112 0) -
                        RECORDS(300 150)) -
        DATA      (NAME(SYS1.APPCTP.DATA)) -
        INDEX      (NAME(SYS1.APPCTP.INDEX))
/*
//SI      EXEC PGM=IDCAMS
//SYSPRINT DD  SYSOUT=*
//SYSIN    DD  *
        DEFINE CLUSTER (NAME(SYS1.APPCSI) -
                        VOLUME(*****)) -
                        INDEXED REUSE -
                        SHAREOPTIONS(3 3) -
                        RECORDSIZE(248 248) -
                        KEYS(112 0) -
                        RECORDS(50 25)) -
        DATA      (NAME(SYS1.APPCSI.DATA)) -
        INDEX      (NAME(SYS1.APPCSI.INDEX))
/*

```

Figura 14. JCL para Criar VSAMs do APPC

VTAM

O APPC é uma implementação do protocolo SNA (Systems Network Architecture) LU 6.2. O SNA oferece formatos e protocolos que definem uma variedade de componentes SNA físicos e lógicos, como a LU (Logical Unit). A LU 6.2 é um tipo de unidade lógica desenvolvida especificamente para controlar a comunicação entre os programas aplicativos.

Para utilizar SNA no MVS, você precisa instalar e configurar o VTAM (Virtual Telecommunications Access Method). O VTAM deve estar ativo antes que as tarefas do sistema APPC possam ser utilizadas.

A parte específica APPC da configuração VTAM consiste em três etapas:

1. Definir o nome do modo utilizado (por exemplo, APPCHOST) para VTAM, utilizando SYS1.SAMPLIB(ATBLJOB) para montar e editar por interligação SYS1.SAMPLIB(ATBLMODE) em SYS1.VTAMLIB. Consulte o membro SYS1.SAMPLIB(ATBLMODE) para obter detalhes.
2. Definir APPC/MVS como um aplicativo VTAM, copiando o membro de amostra SYS1.SAMPLIB(ATBAPPL) para um conjunto de dados na concatenação SYS1.VTAMLST. Consulte o membro SYS1.SAMPLIB(ATBAPPL) para obter detalhes.
3. Utilize o comando do console **v net,act,id=atbappl** para ativar o aplicativo recém-definido (em que net é igual ao nome do VTAM STC). Utilize o

comando do console **d net,appls** para verificar se o aplicativo está ativo. Inclua o nome do membro em SYS1.VTAMLST(ATCCONxx) se quiser que seja ativado quando o VTAM for iniciado.

O ACBNAME de MVSLU01 utilizado no membro de amostra SYS1.SAMPLIB(ATBAPPL) pode ser utilizado para corresponder aos padrões do site, mas deve corresponder às definições no membro 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
```

Figura 15. SYS1.SAMPLIB(ATBAPPL)

Consulte o *Communications Server bookshelf (F1A1BK61 para z/OS 1.7)* para obter mais informações sobre a configuração do VTAM.

SYS1.PARMLIB(APPCPMxx)

Para ativar e suportar o fluxo de conversações entre sistemas, os sites devem definir LUs entre quais sessões podem fazer ligação. Um site precisa definir, pelo menos, uma LU antes que o processamento de APPC/MVS possa ocorrer, mesmo quando o processamento APPC permanecer em um sistema único. As LUs são algumas das definições realizadas em SYS1.PARMLIB(APPCPMxx).

O serviço de comandos do TSO requer que o APPC seja configurado para ter uma LU de base que possa administrar pedidos de entrada e saída.

A definição da LU deve ser incluída no membro SYS1.PARMLIB(APPCPMxx) e precisa incluir os parâmetros BASE e SCHED(ASCH). O membro APPCPMxx também especifica quais conjuntos de dados VSAM de TP e de SI VSAM serão utilizados.

Figura 16 é um membro SYS1.PARMLIB(APPCPMxx) de amostra que pode ser utilizado para o serviço Comandos do TSO.

```
LUADD
  ACBNAME(MVSLU01)
  BASE
  SCHED(ASCH)
  TPDATA(SYS1.APPCTP)
SIDEINFO DATASET(SYS1.APPCSI)
```

Figura 16. SYS1.PARMLIB(APPCPMxx)

Quando um sistema possui vários nomes de LU, pode ser necessário fazer alterações, dependendo de qual LU o sistema seleciona como LU BASE. A LU BASE para o sistema é determinada por:

1. A LU base do sistema é representada pela última instrução LUADD que contém os parâmetros NOSCHED e BASE. Esse tipo de LU base do sistema permitirá que os pedidos de saída sejam processados quando nenhum planejador de transação estiver ativo.
2. Se nenhuma instrução LUADD contiver NOSCHED e BASE, a LU base do sistema será representada pela última instrução LUADD que contém o parâmetro BASE e especifica ASCH como planejador de transações APPC/MVS. Isto pode ser realizado codificando SCHED(ASCH) ou não codificando o parâmetro SCHED (ASCH é o valor padrão para SCHED).

Se seu sistema possui uma LU com parâmetros BASE e NOSCHED, esta LU seria utilizada como a LU BASE mas o serviço de comandos do TSO não funcionará porque esta LU não possui um planejador de transações para administrar pedidos para a transação FEKFRSRV. Se esta LU não puder ser alterada para remoção do parâmetro NOSCHED, a variável de ambiente `rsed.envvars (_FEKFSCMD_PARTNER_LU)` pode ser configurada para a LU que possui BASE e SCHED(ASCH), tal como:

```
_FEKFSCMD_PARTNER_LU=MVSLU01
```

Consulte “Customizar `rsed.envvars`, o Arquivo de Configuração para RSE” na página 35 para obter mais informações sobre `rsed.envvars`.

SYS1.PARMLIB(ASCHPMxx)

O planejador de transações APPC/MVS (nome padrão é ASCH) inicia e planeja programas de transações em resposta aos pedidos de entrada para conversões. O membro `SYS1.PARMLIB(ASCHPMxx)` controla seu funcionamento, por exemplo, com definições de classe da transação.

A classe de transação APPC utilizada para o serviço de comandos do TSO deve ter iniciadores APPC suficientes para permitir um iniciador para cada usuário da função Remote Edit-Compile-Debug.

O serviço de comandos do TSO também precisa das especificações padrão para ser especificado nas seções `OPTIONS` e `TPDEFAULT`.

Figura 17 é um membro `SYS1.PARMLIB(ASCHPMxx)` de amostra que pode ser utilizado para o serviço Comandos do TSO.

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

OPTIONS
  DEFAULT(A)

TPDEFAULT
  REGION(2M)
  TIME(5)
  MSGLEVEL(1,1)
  OUTCLASS(X)
```

Figura 17. SYS1.PARMLIB(ASCHPMxx)

Nota: Para depuração, o IBM Support Center pode solicitar que você aumente o valor de `MSGLIMIT`, para que mais saída seja gravada no arquivo de log.

Ativando Alterações do APPC

As alterações na configuração documentadas nas etapas acima agora podem ser ativadas. Isso pode ser feito de diversas maneiras, dependendo das circunstâncias:

- O APPC ainda não está ativo. Digite os seguintes comandos de console para iniciar o APPC/MVS (em que xx é igual aos últimos 2 caracteres dos membros SYS1.PARMLIB relacionados):

1. S APPC,SUB=MSTR,APPC=xx

2. S ASCH,SUB=MSTR,ASCH=xx

Inclua esses comandos em SYS1.PARMLIB(COMMNDxx) para iniciá-los na inicialização do sistema.

- APPC já está ativo. O APPC pode recarregar dinamicamente os membros SYS1.PARMLIB, utilizando um comando SET do console (em que xx é igual aos últimos 2 caracteres dos membros SYS1.PARMLIB relacionados):

1. SET APPC=xx

2. SET ASCH=xx

Os comandos de console **D APPC** e **D ASCH** podem ser utilizados para verificar a configuração do APPC. Consulte *MVS System Commands (GC28-1781)* para obter mais informações sobre os comandos de console mencionados.

Definindo a Transação de Serviço Comandos do TSO

Quando o APPC/MVS está ativo, o serviço Comandos do TSO do Developer for System z pode ser definido, conforme descrito em “(Opcional) Definir uma Transação APPC para o Serviço de Comandos do TSO” na página 23.

A maneira documentada de definir a transação APPC é customizando e enviando hlq.SFEKSAMP(FEKAPPCC), em que hlq é igual ao qualificador de alto nível utilizado durante a instalação do Developer for System z (padrão FEK).

A transação APPC também pode ser definida interativamente por meio da interface ISPF do APPC, documentada em um white paper. Esse white paper também descreve como configurar a transação APPC a fim de coletar informações de contabilidade específicas ao usuário.

O white paper *APPC and WebSphere Developer for System z (SC23-5885-00)* está disponível na biblioteca na Internet do Developer for System z, <http://www-306.ibm.com/software/awdtools/devzseries/library/>

Nota: O JCL do TP, utilizado pelo APPC para iniciar o serviço Comandos do TSO, foi alterado no Developer for System z versão 7.1. Se você seguir as orientações no white paper para definir o TP, deve incluir a palavra-chave NESTMACS na linha PARM, por exemplo:

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

Glossário

A

Ação de Bloqueio. Bloqueia um membro

Application Server.

1. Um programa que manipula todas as operações do aplicativo entre computadores baseados em navegador e aplicativos ou bancos de dados de negócios de back-end da organização. Essa é uma classe especial de appservers baseados em Java que seguem o padrão J2EE. O código da J2EE pode ser facilmente colocado entre esses appservers. Pode suportar JSPs e servlets para conteúdo da Web dinâmico e EJBs para transações e acesso ao banco de dados.
2. O destino de um pedido de um aplicativo remoto. No ambiente do DB2, a função de servidor de aplicativos é fornecida pelo recurso de dados distribuídos e é utilizada para acessar os dados do DB2 a partir de aplicativos remotos.
3. Um programa do servidor em uma rede distribuída que fornece o ambiente de execução para um programa aplicativo.
4. O destino de um pedido de um solicitador de aplicativo. O sistema de gerenciamento de banco de dados (DBMS) no site do servidor de aplicativos fornece os dados solicitados.
5. Software que manipula a comunicação com o cliente que está solicitando um recurso e consultas do Content Manager.

Arquivo de Resposta.

1. Um arquivo que contém um conjunto de respostas predefinidas para perguntas feitas por um programa e que é utilizado para que não seja necessário digitar esses valores um a um.
2. Um arquivo ASCII que pode ser customizado com os dados de definição e de configuração e que automatiza uma instalação. Os dados de definição e de configuração seriam digitados durante uma instalação interativa, mas com um arquivo de resposta a instalação pode prosseguir sem nenhuma intervenção.

Atributo Bidirecional. Tipo de Texto, Orientação de Texto, Troca Numérica e Troca Simétrica.

B

Banco de Dados. Um conjunto de itens de dados inter-relacionados ou independentes que são armazenados para servir um ou mais aplicativos.

Biblioteca de Carregamento. Uma biblioteca que contém módulos de carregamento.

Bidirecional (bi-di). Pertencente a scripts, como Árabe e Hebraico, que geralmente são executados da direita para a esquerda, exceto números, que são executados da esquerda para a direita. Essa definição é do Glossário LISA (Localization Industry Standards Association).

Buffer de Erro. Uma parte do armazenamento utilizada para conter temporariamente informações de saída de erro.

C

Compilar.

1. Em linguagens ILE (Integrated Language Environment), para converter instruções de origem em módulos que, então, podem ser ligados a programas ou programas de serviços.
2. Para traduzir todo o programa ou parte dele, expresso em um idioma de alto nível em um programa de computador expresso em uma linguagem intermediária, uma linguagem Assembly ou uma linguagem da máquina.

Conjunto de Dados. A principal unidade de armazenamento e recuperação de dados, que consiste em um conjunto de dados em uma das muitas organizações prescritas e descritas pelas informações de controle às quais o sistema tem acesso.

Contêiner.

1. No CoOperative Development Environment/400, um objeto de sistema que contém e organiza arquivos de origem. Uma biblioteca i5/OS ou um conjunto de dados particionado por MVS são exemplos e um contêiner.
2. Na J2EE, uma entidade que fornece serviços de gerenciamento de ciclo de vida, de segurança, de implementação e de tempo de execução para componentes. (Sun) Cada tipo de contêiner (EJB, Web, JSP, servlet, applet e cliente aplicativo) também fornece serviços específicos ao componente.
3. Em Serviços de Recuperação e Mídia de Backup, o objeto físico utilizado para armazenar e mover mídia, como uma caixa, um estojo ou um rack.
4. Em um servidor de fita virtual (VTS), um receptáculo em que um ou mais volumes lógicos exportados (LVOLs) podem ser armazenados. Um volume temporário que contém um ou mais LVOLs e reside fora de uma biblioteca de VTS é considerado o contêiner para esses volumes.

5. Uma localização do armazenamento físico dos dados. Por exemplo, um arquivo, um diretório ou um dispositivo.
6. Uma coluna ou uma linha que é utilizada para organizar o layout de um portlet ou outro contêiner em uma página.
7. Um elemento da interface com o usuário que contém objetos. No gerenciador de pastas, um objeto que pode conter outras pastas ou documentos.

D

Depurar. Para detectar, diagnosticar e eliminar erros em programas.

Desinstalação Silenciosa. Um processo de desinstalação que não envia mensagens para o console, mas armazena mensagens e erros em arquivos de log depois que o comando de desinstalação é invocado.

G

Gateway.

1. Um componente de middleware que faz uma ponte entre a Internet e os ambientes de intranet durante chamadas de serviço da Web.
2. Software que fornece serviços entre os nós de extremidades e o restante do ambiente Tivoli.
3. Um componente de um VoIP (Voice over Internet Protocol) que fornece uma ponte entre o VoIP e ambientes alternados em circuito.
4. Um dispositivo ou um programa utilizado para conectar redes ou sistemas com diferentes arquiteturas de rede. Os sistemas podem ter diferentes características, como protocolos de comunicação diferentes, arquitetura da rede diferente ou políticas de segurança diferentes; nesse caso, o gateway desempenha a função de tradução e também de conexão.

I

ID da Ação. Um identificador numérico para uma ação entre 0 e 999

Instalação Silenciosa. Uma instalação que não envia mensagens para o console, mas armazena mensagens e erros em arquivos de log. Além disso, uma instalação silenciosa pode utilizar arquivos de resposta para entrada de dados.

Instância do Repositório. Um projeto ou um componente que existe em um SCM.

Interactive System Productivity Facility (ISPF). Um programa licenciado IBM que serve como um editor de tela inteira e um gerenciador de diálogos. Utilizado para gravar programas aplicativos, permite gerar

painéis de tela padrão e diálogos interativos entre o programador do aplicativo e o usuário terminal. O ISPF consiste em quatro componentes principais: o DM, o PDF, o SCLM e o C/S. O componente DM é o Dialog Manager, que fornece serviços para diálogos e usuários finais. O componente PDF é o Program Development Facility, que fornece serviços para auxiliar o diálogo ou o desenvolvedor de aplicativos. O componente SCLM é o Software Configuration Library Manager, que fornece serviços para que desenvolvedores de aplicativos gerenciem suas bibliotecas de desenvolvimento de aplicativos. O componente C/S é o Client/Server, que permite executar o ISPF em uma estação de trabalho programável, para exibir os painéis que utilizam a função de exibição do sistema operacional da estação de trabalho e para integrar ferramentas e dados da estação de trabalho a ferramentas e dados do host.

Intérprete. Um programa que traduz e executa cada instrução de uma linguagem de programação de alto nível antes de traduzir e executar a próxima instrução.

Isomórfico. Cada elemento composto (em outras palavras, um elemento contendo outros elementos) do documento da instância XML iniciando a partir da raiz tem um, e apenas um, item do grupo COBOL correspondente cuja profundidade de aninhamento é idêntica à profundidade de aninhamento de seu equivalente XML. Cada elemento não-composto (em outras palavras, um elemento que não contém outros elementos) do documento da instância XML iniciando a partir da parte superior tem um e apenas um item elementar COBOL correspondente cuja profundidade de aninhamento é idêntica ao nível de aninhamento de seu equivalente XML e cujo endereço de memória no tempo de execução pode ser exclusivamente identificado.

L

Lista de Tarefas. Uma lista de procedimentos que podem ser executados por um único fluxo de controle.

N

Não-isomórfico. um mapeamento simples de itens COBOL e elementos XML pertencentes a documentos XML e grupos COBOL que não têm shapes idênticos (não-isomórficos). O mapeamento não-isomórfico também pode ser criado entre elementos não-isomórficos de estruturas isomórficas.

Nome da Shell. O nome da interface shell.

P

Pedido de Construção. Um pedido do cliente para executar uma transação de construção.

Perspectiva. Um grupo de visualizações que mostram vários aspectos dos recursos do workbench. O usuário do workbench pode alternar perspectivas, dependendo da tarefa disponível, e customizar o layout de visualizações e editores na perspectiva.

Perspectiva Sistemas Remotos. Fornece uma interface para gerenciar sistemas remotos utilizando convenções que são semelhantes ao ISPF

R

RAM. Repository Access Manager

Repositório.

1. Uma área de armazenamento de dados. Cada repositório tem um nome e um tipo de item de negócios associado. Por padrão, o nome será igual ao nome do item de negócios. Por exemplo, um repositório para faturas será chamado Faturas. Há dois tipos de repositórios de informações: local (específico ao processo) e global (reutilizável).
2. Um conjunto de dados VSAM em que os estados de processos BTS são armazenados. Quando um processo não está sendo executado sob o controle do BTS, seu estado (e os estados de suas atividades constituintes) é preservado com a gravação em um conjunto de dados do repositório. Os estados de todos os processos de um tipo de processo específico (e de suas instâncias de atividade) são armazenados no mesmo conjunto de dados do repositório. Registros de vários tipos de processo podem ser gravados no mesmo repositório.
3. Uma área de armazenamento persistente para código fonte e outros recursos de aplicativo. Em um ambiente de programação de equipe, um repositório compartilhado permite o acesso multiusuário a recursos de aplicativo.
4. Um conjunto de informações sobre os gerenciadores de filas que são membros de um cluster. Essas informações incluem nomes de gerenciadores de filas, seus locais, seus canais, quais filas eles hospedam, etc.

S

Script da Shell. Um arquivo que contém comandos que podem ser interpretados pela shell. O usuário digita o nome do arquivo de script no prompt do comando shell para fazer com que a shell execute os comandos do script.

Seção de Ligação. A seção da divisão de dados de uma unidade ativada (um programa chamado ou um método invocado) que descreve itens de dados disponíveis da unidade de ativação (um programa ou um método). Esses itens de dados podem ser utilizados como referência tanto pela unidade ativada quanto pela unidade de ativação.

Sessão de Depuração. As atividades de depuração que ocorrem entre o momento em que o desenvolvedor inicia um depurador e o momento em que ele sai dali.

Shell. Uma interface de software entre usuários e o sistema operacional que interpreta comandos e interações com o usuário e comunica-os ao sistema operacional. Um computador pode ter várias camadas de shells para diversos níveis de interação com o usuário.

Siddeck. Uma biblioteca que publica as funções de um programa DLL. Os nomes de entrada e de módulo são armazenados na biblioteca após a compilação do código fonte.

Sistema de Arquivo Remoto. Um sistema de arquivo que reside em um servidor ou sistema operacional separado.

Sistema Remoto. Qualquer outro sistema na rede com o qual o sistema possa se comunicar.

T

Transação de Construção. Uma tarefa iniciada no MVS para executar construções após um pedido de construção ser recebido do cliente.

U

URL. Uniform Resource Locator

V

Visualização Console de Saída. Exibe a saída de um processo e permite fornecer entrada do teclado a um processo.

Visualização Definição de Dados. Contém uma representação local de bancos de dados e seus objetos e fornece recursos para manipular esses objetos e exportá-los para um banco de dados remoto

Visualização Navegador. Fornece uma visualização hierárquica dos recursos do Workbench.

Visualização Repositórios. Exibe os locais de repositório CVS que foram incluídos no Workbench.

Visualização Saída. Exibe mensagens, parâmetros e resultados relacionados aos objetos com os quais você trabalha

Visualização Servidores. Exibe uma lista de todos os servidores e as configurações associadas a eles.

Avisos

Estas informações foram desenvolvidas para produtos e serviços oferecidos nos Estados Unidos.

É possível que a IBM não ofereça os produtos, serviços ou recursos discutidos neste documento em outros países. Consulte o representante IBM local para obter informações sobre os produtos e serviços disponíveis atualmente em sua área. Qualquer referência a produtos, programas ou serviços IBM não significa que apenas produtos, programas ou serviços IBM possa ser utilizados. Qualquer produto, programa ou serviço funcionalmente equivalente, que não infrinja nenhum direito de propriedade intelectual da IBM, poderá ser utilizado em substituição a este produto, programa ou serviço. Entretanto, a avaliação e verificação da operação de qualquer produto, programa ou serviço não-IBM são de responsabilidade do Cliente.

A IBM pode ter patentes ou solicitações de patentes pendentes relativas a assuntos tratados nesta publicação. O fornecimento desta publicação não garante ao Cliente nenhum direito sobre tais patentes. Pedidos de licença podem ser enviados, por escrito, para:

Gerência de Relações Comerciais e Industriais da *IBM Brasil*
Av. Pasteur, 138-146
Botafogo
Rio de Janeiro, RJ
CEP 22290-240

Para pedidos de licença relacionados a informações de DBCS (Conjunto de Caracteres de Byte Duplo), entre em contato com o Departamento de Propriedade Intelectual da IBM em seu país ou envie pedidos de licença, por escrito, para:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

O parágrafo a seguir não se aplica ao Reino Unido ou a qualquer outro país no qual tais provisões sejam inconsistentes com a legislação local: A INTERNATIONAL BUSINESS MACHINES CORPORATION FORNECE ESTA PUBLICAÇÃO "NO ESTADO EM QUE SE ENCONTRA", SEM GARANTIA DE NENHUM TIPO, SEJA EXPRESSA OU IMPLÍCITA, INCLUINDO, MAS NÃO SE LIMITANDO ÀS GARANTIAS IMPLÍCITAS DE MERCADO OU DE ADEQUAÇÃO A UM DETERMINADO PROPÓSITO. Alguns países não permitem a exclusão de garantias explícitas ou implícitas em determinadas transações; portanto, esta disposição pode não se aplicar ao Cliente.

Essas informações podem conter imprecisões técnicas ou erros tipográficos. Periodicamente, são feitas alterações nas informações aqui contidas; tais alterações serão incorporadas em futuras edições desta publicação. A IBM pode, a qualquer momento, aperfeiçoar e/ou alterar os produtos e/ou programas descritos nesta publicação, sem aviso prévio.

Referências nestas informações a Web sites não-IBM são fornecidas apenas por conveniência e não representam de forma alguma um endosso a esses Web sites. Os materiais contidos nesses Web sites não fazem parte dos materiais deste produto IBM e a utilização desses Web sites é de inteira responsabilidade do Cliente.

A IBM pode utilizar ou distribuir as informações fornecidas da forma que julgar apropriada sem incorrer em qualquer obrigação para com o Cliente.

Licenciados deste programa que desejam obter informações sobre este assunto com objetivo de permitir: (i) a troca de informações entre programas criados independentemente e outros programas (incluindo este) e (ii) a utilização mútua das informações trocadas, devem entrar em contato com:

Gerência de Relações Comerciais e Industriais da IBM Brasil
Av. Pasteur, 138-146
Botafogo
Rio de Janeiro, RJ
CEP 22290-240

Tais informações podem estar disponíveis, sujeitas a termos e condições apropriadas, incluindo em alguns casos o pagamento de uma taxa.

O programa licenciado descrito nesta publicação e todo material licenciado disponível são fornecidos pela IBM sob os termos do Contrato com o Cliente IBM, do Contrato de Licença de Programa Internacional IBM ou de qualquer outro contrato equivalente.

Quaisquer dados de desempenho contidos aqui foram determinados em ambientes controlados. Portanto, os resultados obtidos em outros ambientes operacionais poderão variar significativamente. Algumas medidas podem ter sido tomadas em sistemas em fase de desenvolvimento e não há garantia de que tais medidas sejam as mesmas nos sistemas normalmente disponíveis. Além do mais, algumas medidas podem ter sido estimadas por meio da extrapolação. Os resultados reais podem ser diferentes. Os usuários deste documento devem verificar os dados aplicáveis para seu ambiente específico.

As informações relativas a produtos não-IBM foram obtidas junto aos fornecedores dos respectivos produtos, de seus anúncios publicados ou de outras fontes disponíveis publicamente. A IBM não testou estes produtos e não pode confirmar a precisão de seu desempenho, compatibilidade ou qualquer outra reivindicação relacionada a produtos não-IBM. Dúvidas sobre os recursos de produtos não-IBM devem ser dirigidas aos fornecedores destes produtos.

Todas as declarações relacionadas aos objetivos e intenções futuras da IBM estão sujeitas a alterações ou cancelamento sem aviso prévio e representam apenas metas e objetivos.

Estas informações contêm exemplos de dados e relatórios utilizados nas operações diárias de negócios. Para ilustrá-los da forma mais completa possível, os exemplos podem incluir nomes de indivíduos, empresas, marcas e produtos. Todos estes nomes são fictícios e qualquer semelhança a esses nomes e endereços utilizados por uma empresa comercial real é mera coincidência.

LICENÇA DE DIREITOS AUTORAIS:

Estas informações contêm programas de aplicativos de exemplos na linguagem fonte, ilustrando as técnicas de programação em diversas plataformas operacionais. Você pode copiar, modificar e distribuir estes programas de exemplo sem a necessidade de pagar à IBM, com objetivos de desenvolvimento, utilização, marketing ou distribuição de programas aplicativos em conformidade com a interface de programação de aplicativo para a plataforma operacional para a qual os programas de exemplo são criados. Esses exemplos não foram completamente testados em todas as condições. Portanto, a IBM não pode garantir ou implicar confiabilidade, manutenção, ou função destes programas. Você pode copiar, modificar e distribuir estes programas de exemplo de qualquer maneira sem pagamento à IBM, com objetivos de desenvolvimento, utilização, marketing ou distribuição de programas aplicativos em conformidade com interfaces de programação de aplicativos da IBM.

Cada cópia ou parte desses programas de exemplo ou qualquer trabalho derivado deve incluir um aviso de copyright com os dizeres:

© (nome da sua empresa) (ano). Partes deste código são derivadas dos Programas de Exemplo da IBM Corp. © Copyright IBM Corp. _digite o ano ou anos_. Todos os direitos reservados.

Marcas Registradas e Marcas de Serviço

Os termos a seguir são marcas ou marcas registradas da International Business Machines Corporation nos Estados Unidos e/ou em outros países:

- CICS
- CICSplex
- DB2
- IBM
- IMS
- MVS
- OS/390
- RACF
- Rational
- System z
- VTAM
- WebSphere
- z/OS
- zSeries

Java e todas as marcas e logotipos baseados em Java são marcas ou marcas registradas da Sun Microsystems, Inc. nos Estados Unidos e em outros países.

Microsoft, Windows, Windows NT e o logotipo Windows são marcas ou marcas registradas da Microsoft Corporation nos Estados Unidos e/ou em outros países.

UNIX é uma marca registrada da The Open Group.

Outros nomes de empresas, produtos ou serviços, que podem estar indicados por asteriscos duplos (**), podem ser marcas registradas ou marcas de serviço de terceiros.

Comentários do Leitor

IBM Rational Developer para System z
Guia de Configuração do Host
Versão 7.1

Publicação N° S517-9094-00

Neste formulário, faça-nos saber sua opinião sobre este manual. Utilize-o se encontrar algum erro, ou se quiser externar qualquer opinião a respeito (tal como organização, assunto, aparência...) ou fazer sugestões para melhorá-lo.

Para pedir publicações extras, fazer perguntas ou tecer comentários sobre as funções de produtos ou sistemas IBM, fale com o seu representante IBM.

Quando você envia seus comentários, concede direitos, não exclusivos, à IBM para usá-los ou distribuí-los da maneira que achar conveniente, sem que isso implique em qualquer compromisso ou obrigação para com você.

Não se esqueça de preencher seu nome e seu endereço abaixo, se deseja resposta.

Comentários:

Nome

Endereço

Companhia ou Empresa

Telefone

IBM Brasil - Centro de Traduções
Rodovia SP 101 km 09
CEP 13185-900
Hortolândia, SP



Número do Programa: 5724-T07

Impresso em Brazil

S517-9094-00

