

**Rational** IBM Rational Developer para System z  
Versão 7.6.1

## *Guia de Configuração do Host*





**Rational** IBM Rational Developer para System z  
Versão 7.6.1

## *Guia de Configuração do Host*



**Nota**

Antes de usar este documento, leia as informações gerais em “Notas de Documentação para IBM Rational Developer para System z” na página 349.

**Quinta Edição (Maio de 2010)**

Esta edição aplica-se ao IBM Rational Developer para System z Versão 7.6.1 (número do programa 5724-T07) e a todas modificações e releases subsequentes, até indicado de forma diferente 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 Estados Unidos. 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 a seguir.

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

Av. Pasteur, 138-146  
Rodovia SP 101 Km 09  
CEP 13185-900  
Hortolândia,  
SP

É possível enviar um fax com os seus comentários para: 1-800-227-5088 (Estados Unidos e Canadá)

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

© Copyright IBM Corporation 2005, 2010.

# Índice

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

<b>Tabelas . . . . .</b>	<b>xi</b>
--------------------------	-----------

<b>Sobre este Documento . . . . .</b>	<b>xiii</b>
---------------------------------------	-------------

Quem Deve Usar este Documento . . . . .	xiii
Resumo das Mudanças . . . . .	xiii
Descrição do Conteúdo do Documento . . . . .	xiv
Planejamento . . . . .	xiv
Customização Básica . . . . .	xiv
(Opcional) CARMA (Common Access Repository Manager) . . . . .	xiv
(Opcional) Application Deployment Manager . . . . .	xv
(Opcional) SCLM Developer Toolkit . . . . .	xv
(Opcional) Outras Tarefas de Customização . . . . .	xv
Verificação de Instalação . . . . .	xvi
Comandos do operador . . . . .	xvi
Resolução de Problemas de Configuração . . . . .	xvi
Considerações de segurança . . . . .	xvi
Entendendo o Developer para System z . . . . .	xvi
Considerações WLM . . . . .	xvi
Considerações de Ajuste . . . . .	xvii
Considerações de Desempenho . . . . .	xvii
Considerações sobre o CICSTS . . . . .	xvii
Customizando o Ambiente TSO . . . . .	xvii
Executando várias instâncias . . . . .	xvii
Guia de Migração . . . . .	xvii
Configurando o SSL e a Autenticação X.509 . . . . .	xvii
Configurando o TCP/IP . . . . .	xviii
Configurando o INETD . . . . .	xviii
Configurando o APPC . . . . .	xviii
Requisitos . . . . .	xviii

## Parte 1. Customização do Developer para System z . . . . . 1

### Capítulo 1. Planejamento . . . . . 3

Considerações de Migração . . . . .	3
Considerações de Planejamento . . . . .	3
Visão Geral do Produto . . . . .	3
Exigências de Capacidade . . . . .	4
Requisitos de Tempo . . . . .	4
Considerações sobre Pré-Instalação . . . . .	4
Opção de Configuração . . . . .	5
Requisito de produtos . . . . .	5
Recursos Necessários . . . . .	5
Considerações de Pré-configuração . . . . .	8
Gerenciamento de Carga de Trabalho . . . . .	8
Uso de Recursos e Limites do Sistema . . . . .	9
Configuração Necessária de Produtos de Requisito . . . . .	9
Considerações de ID do Usuário . . . . .	9
Considerações do Servidor . . . . .	10
Método de Configuração . . . . .	11
Considerações sobre Pré-implementação . . . . .	11

Lista de Verificação do Cliente . . . . .	12
---	----

## Capítulo 2. Customização Básica . . . . . 15

Requisitos e Lista de Verificação . . . . .	15
Configuração da Customização . . . . .	15
Alterações PARMLIB . . . . .	16
Configurar Limites do z/OS UNIX no BPXPRMxx . . . . .	16
Incluir Tarefas Iniciadas em COMMNDxx . . . . .	17
Definições de LPA em LPALSTxx . . . . .	18
Autorizações APF em PROGxx . . . . .	18
Definições de LINKLIST no PROGxx . . . . .	19
Definições de LINKLIST e LPA de Requisito . . . . .	20
Definições LINKLIST para Outros Produtos . . . . .	21
Alterações do PROCLIB . . . . .	21
JES Job Monitor . . . . .	21
Daemon RSE . . . . .	22
Daemon de Bloqueio . . . . .	23
Limitações da JCL para a Variável PARM . . . . .	24
Procedimentos de Construção Remota ELAXF* . . . . .	25
Definições de Segurança . . . . .	26
FEJJCNF, Arquivo de Configuração do JES Job Monitor . . . . .	27
Arquivo de Configuração do RSE rsed.envvars . . . . .	31
Definindo o PORTANGE Disponível para o Servidor RSE . . . . .	41
Definindo Parâmetros Extras de Inicialização Java com _RSE_JAVAOPTS . . . . .	42
Definindo Parâmetros Extras de Inicialização Java com _RSE_CMDSERV_OPTS . . . . .	46
ISPF.conf, Arquivo de Configuração do TSO/ISPF Client Gateway do ISPF . . . . .	47
Componentes Opcionais . . . . .	48
Verificação de Instalação . . . . .	48

## Capítulo 3. (Opcional) CARMA (Common Access Repository Manager) 49

Requisitos e Lista de Verificação . . . . .	49
Componentes do CARMA . . . . .	50
Notas de Migração VSAM do CARMA . . . . .	51
Interface RSE para CARMA . . . . .	51
Inicialização do Servidor CARMA Usando Submissão em Lote . . . . .	53
Ajustar CRASRV.properties . . . . .	54
Ajustar CRASUBMT . . . . .	54
(Opcional) Inicialização Alternativa do Servidor CARMA Utilizando CRASTART . . . . .	55
Ajustar CRASRV.properties . . . . .	56
Ajustar crastart.conf . . . . .	56
(Opcional) Inicialização Alternativa do Servidor CARMA Usando TSO/ISPF Client Gateway . . . . .	58
Ajustar CRASRV.properties . . . . .	58
Ajustar ISPF.conf . . . . .	59
(Opcional) Ativando os RAM (Repository Access Managers) de Amostra . . . . .	60

Ativando o PDS RAM . . . . .	61
Ativando o RAM do SCLM . . . . .	61
Ativando o RAM da Estrutura . . . . .	61
(Opcional) Ativando o CA Endeavor SCM RAM . . . . .	61
Requisitos e Lista de Verificação . . . . .	62
Defina o CA Endeavor SCM RAM . . . . .	62
Inicialização do CA Endeavor SCM RAM Usando Submissão em Lote . . . . .	63
Inicialização do CA Endeavor SCM RAM Usando CRASTART . . . . .	65
(Opcional) Customize CRANDVRA . . . . .	66
(Opcional) Customize o CA Endeavor SCM RAM . . . . .	67
(Opcional) Suportando Múltiplos RAMs . . . . .	67
Exemplo . . . . .	68
(Opcional) IRXJCL versus CRAXJCL . . . . .	69
Criar CRAXJCL . . . . .	69

## Capítulo 4. (Opcional) Application

### Deployment Manager . . . . . 71

Requisitos e Lista de Verificação . . . . .	71
Repositório do CRD . . . . .	72
CICS Administrative Utility . . . . .	72
RESTful versus Serviços da Web . . . . .	73
Servidor CRD Usando a Interface RESTful . . . . .	73
Região de Conexão Primária do CICS . . . . .	73
Regiões de Conexão Não-primária do CICS . . . . .	74
(Opcional) Customizar IDs de Transação do Servidor CRD . . . . .	74
Servidor CRD Usando a Interface do Serviço da Web . . . . .	75
Manipulador de Mensagens do Pipeline . . . . .	75
região de conexão primária do CICS . . . . .	76
regiões de conexão não-primária do CICS . . . . .	76
(Opcional) Repositório de Manifesto . . . . .	77

## Capítulo 5. (Opcional) SCLM Developer

### Toolkit . . . . . 79

Requisitos e Lista de Verificação . . . . .	79
Pré-requisitos . . . . .	80
Atualizações do ISPF.conf para SCLMDT . . . . .	80
Atualizações do rsed.envvars para SCLMDT . . . . .	81
(Opcional) Tradução de Nome Longo/Abreviado . . . . .	82
Criar LSTRANS.FILE, o VSAM de Conversão de nome Longo/Curto . . . . .	82
Atualizações de rsed.envvars para Conversão de Nomes Longos/Curtos . . . . .	84
(Opcional) Instalar e Customizar Ant . . . . .	84
Atualizações SCLM para SCLMDT . . . . .	85
Remover Arquivos Antigos de WORKAREA . . . . .	86

## Capítulo 6. (Opcional) Outras Tarefas de Customização . . . . . 87

(Opcional) Procedimento Armazenado do DB2 . . . . .	87
Alterações do Workload Manager (WLM) . . . . .	87
Alterações do PROCLIB . . . . .	88
Mudanças no DB2 . . . . .	88
(Opcional) Suporte de Enterprise Service Tools (EST) . . . . .	89
(Opcional) Suporte de Idioma Bidirecional do CICS . . . . .	90
(Opcional) Mensagens de Erro IRZ de Diagnóstico . . . . .	91
(Opcional) Criptografia SSL do RSE . . . . .	91

(Opcional) Rastreamento de RSE . . . . .	94
(Opcional) Grupos de Propriedade Baseados em Host . . . . .	96
(Opcional) Projetos Baseados no Host . . . . .	97
(Opcional) Integração do File Manager . . . . .	98
(Opcional) Caracteres Não-editáveis . . . . .	99
(Opcional) Usando REXEC (ou SSH) . . . . .	100
Ações Remotas (Baseadas em Host) para Subprojetos z/OS UNIX . . . . .	101
Método de Conexão Alternativo do RSE . . . . .	101
Configuração de REXEC (ou SSH) . . . . .	101
(Opcional) Transação APPC para o Serviço de Comandos TSO . . . . .	102
Preparação . . . . .	102
Implementação . . . . .	104
Considerações sobre Uso de APPC . . . . .	105
(Opcional) Limpeza de WORKAREA . . . . .	105

## Capítulo 7. Verificação de Instalação 107

Verificar Tarefas Iniciadas . . . . .	107
JMON, JES Job Monitor . . . . .	107
LOCKD, Daemon de Bloqueio . . . . .	107
RSED, Daemon RSE . . . . .	107
Verificar Serviços . . . . .	109
Inicialização do IVP . . . . .	110
Disponibilidade de Porta . . . . .	111
Configuração de TCP/IP . . . . .	112
Conexão do Daemon RSE . . . . .	113
Conexão do JES Job Monitor . . . . .	113
Conexão do Daemon de Bloqueio . . . . .	113
Conexão do TSO/ISPF Client Gateway do ISPF . . . . .	114
(Opcional) Conexão do Serviço TSO Commands Utilizando APPC . . . . .	115
(Opcional) Conexão SCLMDT . . . . .	116
(Opcional) Conexão REXEC . . . . .	118
(Opcional) Script de Shell REXEC/SSH . . . . .	119

## Parte 2. Informações do Developer para System z . . . . . 121

## Capítulo 8. Comandos do Operador 123

Start (S) . . . . .	123
JES Job Monitor . . . . .	123
Daemon RSE . . . . .	124
Daemon de Bloqueio . . . . .	124
Modify (F) . . . . .	125
JES Job Monitor . . . . .	125
Daemon RSE . . . . .	126
Daemon de Bloqueio . . . . .	129
Stop (P) . . . . .	129
Mensagens do Console . . . . .	130
JES Job Monitor . . . . .	130
Daemon RSE, Servidor do Conjunto de Encadeamento RSE e Daemon de Bloqueio . . . . .	130
Como Ler um Diagrama de Sintaxe . . . . .	132
Símbolos . . . . .	132
Operandos . . . . .	132
Exemplo de Sintaxe . . . . .	133
Caracteres Não Alfanuméricos e Espaços em Branco . . . . .	133

Selecionando Mais de Um Operando . . . . .	133
Mais Longo que Uma Linha . . . . .	133
Fragmentos de Sintaxe . . . . .	133

## Capítulo 9. Resolução de Problemas de Configuração . . . . . 135

Análise de Log e de Configuração Usando FEKLOGS . . . . .	135
Arquivos de Log . . . . .	136
Criação de Logs do JES Job Monitor. . . . .	137
Criação de Log do Daemon de Bloqueio . . . . .	137
Criação de Log de Daemon RSE e de Conjunto de Encadeamento . . . . .	137
Criação de Log de Usuário do RSE . . . . .	138
Criação de Log de Integração do Fault Analyzer . . . . .	139
Criação de Log de Integração do File Manager . . . . .	139
Criação de Log do SCLM Developer Toolkit . . . . .	140
Criação de Logs do CARMA . . . . .	140
Criação de Logs de Transações APPC (Serviço de Comandos do TSO) . . . . .	140
Criação de Log de Teste IVP do fekfivpi . . . . .	141
Criação de Log de Teste IVP do fekfivps . . . . .	141
Arquivos de Dump . . . . .	141
Dumps do MVS . . . . .	141
Dumps de Java. . . . .	141
Locais do Dump do z/OS UNIX . . . . .	143
Rastreio . . . . .	143
Rastreio do JES Job Monitor . . . . .	143
Rastreio do RSE . . . . .	143
Rastreio do Daemon de Bloqueio. . . . .	144
Rastreio do CARMA . . . . .	144
Rastreio de Feedback de Erro . . . . .	145
Bits de Permissão do z/OS UNIX . . . . .	146
Atributo de Sistema de Arquivo SETUID . . . . .	146
Autorização de Controle de Programa . . . . .	146
Autorização APF . . . . .	147
Sticky Bit. . . . .	148
Portas TCP/IP Reservadas . . . . .	149
Tamanho do Espaço de Endereço. . . . .	150
Requisitos da JCL de Inicialização . . . . .	150
Limitações Definidas em SYS1.PARMLIB(BPXPRMxx) . . . . .	150
Limitações Armazenadas no Perfil de Segurança . . . . .	151
Limitações Impostas por Saídas do Sistema . . . . .	151
Limitações para Endereçamento de 64 Bits . . . . .	151
Transação APPC e Serviço de Comandos do TSO . . . . .	152
Informações Diversas. . . . .	153
Limites do Sistema . . . . .	153
Problemas de Requisitos Conhecidos . . . . .	154
Emulador de Conexão do Host . . . . .	155

## Capítulo 10. Considerações sobre Segurança. . . . . 157

Métodos de Autenticação . . . . .	157
ID do Usuário e Senha . . . . .	158
ID do Usuário e Senha Única . . . . .	158
Certificado X.509 . . . . .	158
Autenticação do JES Job Monitor. . . . .	158
Segurança de Conexão . . . . .	159

Limitar Comunicação Externa a Portas Especificadas . . . . .	159
Criptografia de Comunicação Usando SSL. . . . .	159
Verificação de Port Of Entry . . . . .	160
Portas TCP/IP . . . . .	160
Comunicação Externa . . . . .	160
Comunicação Interna. . . . .	161
Portas CARMA e TCP/IP . . . . .	162
Usando os PassTickets . . . . .	162
Criação de Log de Auditoria . . . . .	163
Controle de Auditoria . . . . .	163
Dados de Auditoria . . . . .	163
Segurança do JES . . . . .	164
Ações nas Tarefas - Limitações de Destino. . . . .	164
Ações nas Tarefas - Limitações de Execução . . . . .	166
Acesso aos Arquivos de Spool. . . . .	167
Comunicação Criptografada por SSL . . . . .	167
Autenticação de Cliente Usando Certificados X.509 . . . . .	168
Validação da Autoridade de Certificação (CA) . . . . .	169
(Opcional) Consulte uma Certificate Revocation List (CRL) . . . . .	170
Autenticação por Software de Segurança . . . . .	170
Autenticação por Daemon do RSE . . . . .	171
Verificação de Port Of Entry (POE) . . . . .	172
Segurança do CICSTS . . . . .	172
Repositório do CRD . . . . .	173
Transações do CICS . . . . .	173
Comunicação Criptografada por SSL . . . . .	173
Segurança de SCLM . . . . .	173
Arquivos de Configuração do Developer para System z . . . . .	173
JES Job Monitor - FEJJCENFG . . . . .	173
RSE - rsed.envvars . . . . .	174
RSE - ssl.properties . . . . .	174
Definições de Segurança. . . . .	175
Requisitos e Lista de Verificação . . . . .	175
Ativar Configurações e Classes de Segurança . . . . .	176
Definição de um Segmento OMVS para os Usuários do Developer para System z . . . . .	177
Definir os Perfis do Conjunto de Dados . . . . .	178
Definir as Tarefas Iniciadas do Developer para System z . . . . .	180
Definir a Segurança de Comando do JES . . . . .	182
Definir o RSE como um Servidor z/OS UNIX Seguro . . . . .	183
Definir Bibliotecas Controladas pelo Programa MVS para o RSE . . . . .	184
Definir Proteção de Aplicativo para RSE . . . . .	185
Definir o Suporte PassTicket para o RSE . . . . .	185
Definir Arquivos Controlados pelo Programa z/OS UNIX para o RSE . . . . .	186
Verificar Configurações de Segurança . . . . .	186

## Capítulo 11. Entendimento do Developer para System z . . . . . 189

Visão Geral do Componente . . . . .	189
RSE como um Aplicativo Java. . . . .	191
Donos das Tarefas. . . . .	192
Fluxo de Conexão . . . . .	194
Daemon de Bloqueio . . . . .	196
Liberando um Bloqueio . . . . .	197



Estrutura do Diretório z/OS UNIX . . . . .	198	Opção Java -Xquickstart . . . . .	246
Atualizar Privilégios para Administradores que Não São de Sistema . . . . .	199	Compartilhamento de Classe entre JVMs . . . . .	246
<b>Capítulo 12. Considerações WLM. . . . .</b>	<b>201</b>	Ativar Compartilhamento de Classes . . . . .	246
Classificação de Carga de Trabalho . . . . .	201	Limites de Tamanho de Cache. . . . .	247
Regras de Classificação . . . . .	202	Segurança do Cache . . . . .	247
Configurando Objetivos . . . . .	203	SYS1.PARMLIB(BPXPRMxx) . . . . .	247
Considerações para Seleção de Objetivos . . . . .	204	Espaço em Disco . . . . .	248
STC . . . . .	205	Utilitários de Gerenciamento de Cache . . . . .	248
OMVS. . . . .	206	<b>Capítulo 15. Considerações sobre o CICSTS . . . . .</b>	<b>251</b>
JES. . . . .	207	RESTful versus Serviços da Web . . . . .	252
ASCH. . . . .	208	Regiões de Conexão Primária versus Não-primária	252
CICS . . . . .	208	Log de Instalação do Recurso do CICS . . . . .	253
<b>Capítulo 13. Considerações de Ajuste</b>	<b>211</b>	segurança do Application Deployment Manager	253
Uso de Recursos . . . . .	211	Segurança do Repositório do CRD . . . . .	253
Visão Geral . . . . .	212	Segurança de Pipeline . . . . .	253
Contagem do Espaço de Endereço . . . . .	213	Segurança da Transação . . . . .	253
Contagem de Processos . . . . .	216	Comunicação Criptografada por SSL . . . . .	254
Contagem de Encadeamentos . . . . .	219	Segurança do Recurso . . . . .	255
Uso de Armazenamento. . . . .	222	Administrative Utility . . . . .	255
Limite de Tamanho de Heap Java . . . . .	222	Notas de Migração do Utilitário Administrativo	259
Limite de Tamanho do Espaço de Endereço . . . . .	223	Mensagens do Administrative Utility . . . . .	260
Diretrizes de Estimativa de Tamanho . . . . .	223	<b>Capítulo 16. Customizando o Ambiente TSO . . . . .</b>	<b>263</b>
Análise do Uso de Armazenamento de Amostra	224	O Serviço TSO Commands . . . . .	263
Uso do Espaço do Sistema de Arquivos z/OS		Métodos de Acesso . . . . .	263
UNIX . . . . .	228	Usando o Método de Acesso do TSO/ISPF Client	
Definições de Recursos Principais . . . . .	231	Gateway . . . . .	264
/etc/rdz/rsed.envvars . . . . .	231	Customização Básica – ISPF.conf . . . . .	264
SYS1.PARMLIB(BPXPRMxx) . . . . .	231	Avançado – Usar Perfis do ISPF Existentes . . . . .	264
Definições de Vários Recursos. . . . .	234	Avançado – Usando um Exec de Alocação. . . . .	265
Placa EXEC na JCL do Servidor . . . . .	234	Avançado – Usar Vários Execs de Alocação . . . . .	265
FEK.#CUST.PARMLIB(FEJJCNF) . . . . .	234	Avançado – Vários Arquivos ISPF.conf com várias Configurações do Developer para System	
SYS1.PARMLIB(IEASYSxx) . . . . .	235	z . . . . .	265
SYS1.PARMLIB(IVTPRMxx) . . . . .	235	Usando o Método de Acesso do APPC . . . . .	266
SYS1.PARMLIB(ASCHPMxx) . . . . .	235	Customização Básica – JCL de Transação APPC	266
Monitoramento. . . . .	236	Avançado – Usar Perfis do ISPF Existentes . . . . .	267
Monitoramento de RSE . . . . .	236	Avançado – Usando um Exec de Alocação. . . . .	267
Monitorando z/OS UNIX . . . . .	237	Avançado – Várias Transações APPC com Várias Configurações do Developer para System z . . . . .	268
Monitoramento da Rede. . . . .	239	<b>Capítulo 17. Executando Várias Instâncias . . . . .</b>	<b>271</b>
Monitorando Sistemas de Arquivos z/OS UNIX	239	Configuração Idêntica em um Sysplex . . . . .	271
Configuração de Amostra . . . . .	239	Arquivos de Configuração Diferentes de Níveis de	
Contagem do Conjunto de Encadeamento. . . . .	239	Software Idênticos. . . . .	272
Determinar Limites Mínimos . . . . .	240	Todas as Outras Situações . . . . .	273
Definindo Limites . . . . .	240	<b>Capítulo 18. Guia de Migração . . . . .</b>	<b>277</b>
Uso de Recurso de Monitor . . . . .	241	Considerações de Migração. . . . .	277
<b>Capítulo 14. Considerações de Desempenho . . . . .</b>	<b>243</b>	Fazendo Backup de Arquivos Configurados	
Usar Sistemas de Arquivos zFS . . . . .	243	Anteriormente . . . . .	277
Evite o Uso de STEPLIB. . . . .	243	Notas de Migração da Versão 7.6.1 . . . . .	279
Aprimorar o Acesso às Bibliotecas do Sistema . . . . .	243	Migrar da Versão 7.5 para a Versão 7.6 . . . . .	280
Bibliotecas de Tempo de Execução do Ambiente		IBM Rational Developer para System z, FMID	
de Linguagem (LE) . . . . .	244	HHOP760 . . . . .	280
Desenvolvimento do Aplicativo . . . . .	244		
Aprimorando o Desempenho da Verificação de			
Segurança . . . . .	245		
Gerenciamento de Carga de Trabalho . . . . .	245		
Tamanho de Heap Java Fixo . . . . .	245		



Arquivos Configuráveis . . . . .	282
Migrar da Versão 7.1 para a Versão 7.5 . . . . .	286
IBM Rational Developer para System z, FMID HHOP750 . . . . .	286
Arquivos Configuráveis . . . . .	287
Migrar da Versão 7.0 para a Versão 7.1 . . . . .	289
IBM Rational Developer para System z, FMID HHOP710 . . . . .	289
IBM CARMA, FMID HCMA710 . . . . .	290
Arquivos Configuráveis . . . . .	290

## **Apêndice A. Configurando o SSL e a Autenticação X.509 . . . . . 293**

Decidir Onde Armazenar Chaves Privadas e Certificados . . . . .	293
Criar um Anel de Chave com o RACF . . . . .	295
(Opcional) Usando um Certificado Assinado	296
Clonar a Configuração RSE Existente . . . . .	296
Atualizar rsed.envvars para Ativar a Coexistência	297
Atualizar ssl.properties para Ativar SSL . . . . .	297
Ativar SSL Criando um Novo Daemon RSE . . . . .	297
Testar a Conexão . . . . .	298
(Opcional) Incluir Suporte de Autenticação de Cliente X.509 . . . . .	301
(Opcional) Criar um Banco de Dados de Chaves com gskkyman . . . . .	301
(Opcional) Criar um Keystore com keytool . . . . .	304

## **Apêndice B. Configurando o TCP/IP 307**

Dependência do Nome do Host . . . . .	307
Compreendendo os Resolvedores. . . . .	308
Compreendendo as Ordens de Procura das Informações de Configuração . . . . .	308
Ordens de Procura Usadas no Ambiente do z/OS UNIX . . . . .	309
Arquivos de Base da Configuração do Resolvedor . . . . .	309
Tabelas de Conversão . . . . .	309
Tabelas do Host Local . . . . .	310
Aplicando estas Informações de Configuração no Developer para System z . . . . .	311
O Endereço do Host Não É Resolvido Corretamente . . . . .	313

## **Apêndice C. Configurando o INETD 315**

inetd.conf . . . . .	315
ETC.SERVICES . . . . .	316
Ordem de Procura Usada no Ambiente z/OS UNIX . . . . .	317
Ordem de Procura Usada no Ambiente MVS Ambiente. . . . .	317
Definições de Portas PROFILE.TCPIP . . . . .	318
/etc/inetd.pid . . . . .	319
Inicialização . . . . .	319
/etc/rc . . . . .	319

/etc/inittab . . . . .	319
BPXBATCH . . . . .	320
Sessão de Shell . . . . .	320
Segurança . . . . .	321
Requisitos do Developer para System z. . . . .	322
INETD . . . . .	322
REXEC (ou SSH) . . . . .	322

## **Apêndice D. Configurando o APPC 323**

VSAM. . . . .	323
VTAM. . . . .	324
SYS1.PARMLIB(APPCPMxx) . . . . .	325
SYS1.PARMLIB(ASCHPMxx) . . . . .	326
Ativando Alterações do APPC. . . . .	327
Definindo a Transação de Serviço Comandos do TSO . . . . .	327
(Opcional) Opções de Configuração Alternativas	328
Nome de Transação Alternativo . . . . .	328
Várias LUs . . . . .	328
Segurança da LU . . . . .	328

## **Apêndice E. Requisitos . . . . . 329**

Pré-requisitos de Host do z/OS . . . . .	329
z/OS . . . . .	329
SMP/E . . . . .	330
SDK para z/OS Java 2 Technology Edition . . . . .	331
Co-requisitos de Host do z/OS . . . . .	331
z/OS . . . . .	332
Compilador de COBOL . . . . .	334
Compilador PL/I . . . . .	334
Debug Tool para z/OS . . . . .	335
CICS Transaction Server. . . . .	335
IMS . . . . .	336
DB2 para z/OS. . . . .	337
Rational Team Concert para System z . . . . .	337
File Manager . . . . .	338
Fault Analyzer . . . . .	338
REXX . . . . .	338
Ported tools . . . . .	338
Ant. . . . .	339
Endeavor . . . . .	339

## **Bibliografia . . . . . 341**

Publicações Referenciadas . . . . .	341
Publicações Informativas . . . . .	343

## **Glossário . . . . . 345**

## **Notas de Documentação para IBM Rational Developer para System z . . 349**

Licença de Copyright. . . . .	350
Reconhecimentos de Marca Registrada . . . . .	351

## **Índice Remissivo . . . . . 353**



## Figuras

1.	JMON - Tarefa Iniciada do JES Job Monitor	22
2.	RSED - tarefa iniciada do daemon RSE	23
3.	LOCKD - Tarefa iniciada do daemon de bloqueio . . . . .	23
4.	RSED - inicialização do daemon RSE alternativo . . . . .	24
5.	rsed.stdin.sh - inicialização do daemon RSE alternativo . . . . .	24
6.	FEJCNFG, arquivo de configuração do JES Job Monitor . . . . .	28
7.	rsed.envvars - Arquivo de configuração do RSE . . . . .	34
8.	(continuação) . . . . .	35
9.	ISPF.conf - Arquivo de Configuração do ISPF	47
10.	CRASRV.properties - Arquivo de configuração CARMA . . . . .	52
11.	CRASRV.properties - inicialização do CARMA usando submissão em lote . . . . .	54
12.	CRASUBMT - inicialização do CARMA utilizando submissão em lote . . . . .	55
13.	CRASRV.properties - Inicialização alternativa do CARMA *CRASTART . . . . .	56
14.	crastart.conf - Inicialização alternativa do CARMA *CRASTART . . . . .	57
15.	CRASRV.properties - Inicialização alternativa do CARMA *ISPF . . . . .	59
16.	ISPF.conf - Inicialização alternativa do CARMA *ISPF. . . . .	60
17.	Figura x1. CRASRV.properties - inicialização do CA Endeavor SCM RAM usando submissão em lote . . . . .	63
18.	Figura x2. Inicialização do CRASUBMT - CA Endeavor SCM RAM usando submissão em lote	64
19.	Figura x3. CRASRV.properties - inicialização do CA Endeavor SCM RAM usando CRASTART . . . . .	65
20.	crastart.conf - CA Endeavor inicialização do SCM RAM usando CRASTART . . . . .	66
21.	Atualizações do ISPF.conf para SCLMDT	81
22.	Atualizações do rsed.envvars para SCLMDT	82
23.	FLM02LST - JCL de Configuração da Conversão de Nomes Longos/Curtos . . . . .	83
24.	ELAXMSAM - Tarefa de Procedimento Armazenado do DB2 . . . . .	88
25.	ELAXMJCL - Definição do procedimento armazenado do DB2 . . . . .	89
26.	ssl.properties - Arquivo de configuração SSL	93
27.	rsecomm.properties - Arquivo de configuração da criação de log. . . . .	95
28.	propertiescfg.properties - Arquivo de configuração de grupos de propriedades baseado no host . . . . .	96
29.	projectcfg.properties - Arquivo de configuração de projetos baseados no host . . . . .	98
30.	FMIEXT.properties - Arquivo de configuração do File Manager . . . . .	99
31.	uchars.settings - Arquivo de configuração de caracteres não editáveis . . . . .	100
32.	REXX para Painéis ISPF do APPC. . . . .	103
33.	Comando do operador START JMON	123
34.	Comando do operador START RSED	124
35.	Comando do operador START LOCKD	124
36.	Comando do operador MODIFY JMON	125
37.	Comando do operador MODIFY RSED	126
38.	Comando do operador MODIFY LOCKD	129
39.	Comando do operador STOP . . . . .	130
40.	Portas TCP/IP . . . . .	160
41.	Visão geral do componente . . . . .	189
42.	RSE como um aplicativo Java . . . . .	191
43.	donos das tarefas . . . . .	193
44.	Fluxo de conexão . . . . .	194
45.	Fluxo do daemon de bloqueio . . . . .	196
46.	estrutura do diretório z/OS UNIX . . . . .	198
47.	classificação WLM . . . . .	201
48.	Número máximo de espaços de endereço	215
49.	Número de espaços de endereços por cliente	216
50.	Número máximo de processos . . . . .	217
51.	Número de processos por cliente . . . . .	218
52.	Número máximo de encadeamentos do conjunto de encadeamento do RSE . . . . .	221
53.	Número máximo de encadeamentos do JES Job Monitor . . . . .	221
54.	Uso de recursos com 5 logons . . . . .	225
55.	Uso de recursos com 5 logons (continuação)	226
56.	Uso de recursos ao editar um membro PDS	227
57.	Uso do espaço do sistema de arquivos z/OS UNIX . . . . .	229
58.	Uso do recurso de configuração de amostra	242
59.	ADNJSPAU - Administrative utility do CICSTS . . . . .	257
60.	FEKAPPCC - criar uma segunda transação APPC . . . . .	269
61.	RSESSL - Tarefa do usuário do daemon RSE para SSL . . . . .	298
62.	Diálogo Importar Certificado do Host	299
63.	Diálogo Preferências - SSL . . . . .	300
64.	JCL de Inicialização do INETD. . . . .	320
65.	JCL para Criar VSAMs do APPC . . . . .	324
66.	SYS1.SAMPLIB(ATBAPPL) . . . . .	325
67.	SYS1.PARMLIB(APPCPMxx) . . . . .	325
68.	SYS1.PARMLIB(ASCHPMxx) . . . . .	326



## Tabelas

1. Recursos Necessários. . . . .	6	25. Mecanismos de armazenamento de certificado SSL . . . . .	168
2. Recursos Opcionais . . . . .	6	26. Variáveis de configuração de segurança . . . . .	175
3. Administradores necessários para as tarefas necessárias . . . . .	7	27. Comandos do Operador do JES2 Job Monitor . . . . .	182
4. Administradores Necessários para Tarefas Opcionais . . . . .	7	28. Comandos do Operador do JES3 Job Monitor . . . . .	183
5. Lista de Verificação do Cliente - Partes Obrigatórias . . . . .	13	29. subsistemas de ponto de entrada WLM . . . . .	202
6. Lista de Verificação do Cliente - Partes Opcionais . . . . .	13	30. qualificadores de trabalho WLM . . . . .	203
7. Procedimento ELAXF* de amostra . . . . .	25	31. cargas de trabalho WLM . . . . .	204
8. Lista de Verificação para Qualificadores de Alto Nível ELAXF* . . . . .	26	32. cargas de trabalho WLM STC . . . . .	205
9. Matriz de Permissão do Comando LIMIT_COMMANDS . . . . .	30	33. Cargas de trabalho WLM OMVS . . . . .	206
10. Variáveis de crastart.conf . . . . .	57	34. Carga de trabalhos WLM JES . . . . .	207
11. IDs de Transação do Servidor CRD Padrão . . . . .	74	35. Cargas de trabalho WLM ASCH . . . . .	208
12. IDs de Transação do Servidor CRD Padrão . . . . .	75	36. Cargas de trabalho WLM CICS . . . . .	208
13. Lista de Verificação do Administrador de SCLM . . . . .	85	37. Uso de recursos comuns . . . . .	212
14. Mecanismos de armazenamento de certificado SSL . . . . .	92	38. Uso de recursos obrigatórios específicos do usuário . . . . .	212
15. Tipos de keystore válidos . . . . .	94	39. Uso de recursos específicos do usuário . . . . .	213
16. Lista de verificação de transações APPC . . . . .	103	40. Contagem do espaço de endereço . . . . .	213
17. IVPs para Serviços . . . . .	109	41. Limites de espaço de endereço . . . . .	216
18. Status do erro do conjunto de encadeamento . . . . .	127	42. Contagem de processos . . . . .	216
19. Mensagens de Console do RSE. . . . .	130	43. Limites do processo . . . . .	218
20. Variáveis de JAVA_DUMP_TDUMP_PATTERN. . . . .	142	44. Contagem de encadeamentos . . . . .	219
21. Comandos do Console do JES Job Monitor . . . . .	164	45. Limites de encadeamento . . . . .	222
22. Matriz de Permissão do Comando LIMIT_COMMANDS . . . . .	165	46. Diretivas de saída do log . . . . .	230
23. Perfis JESSPOOL Estendidos . . . . .	165	47. Customizações da Versão 7.6 . . . . .	282
24. Matriz de permissão de navegação LIMIT_VIEW . . . . .	167	48. Customizações da Versão 7.5 . . . . .	287
		49. Customizações da Versão 7.1 . . . . .	290
		50. Mecanismos de armazenamento de certificado SSL . . . . .	294
		51. Definições locais disponíveis para o resolvedor . . . . .	312
		52. Publicações Referenciadas . . . . .	341
		53. Web Sites Referidos . . . . .	343
		54. Publicações Informativas. . . . .	343





---

## Sobre este Documento

Este documento descreve a configuração das funções do IBM Rational Developer para System z. Ele inclui instruções sobre como configurar o IBM Rational Developer para System z Versão 7.6.1 em seu sistema host z/OS.

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

- *IBM Rational Developer para System z* é chamado *Developer para System z*.
- *Common Access Repository Manager* é abreviado para *CARMA*.
- *Software Configuration and Library Manager Developer Toolkit* é chamado *SCLM Developer Toolkit*, abreviado como *SCLMDT*.
- O *z/OS UNIX® System Services* é chamado de *z/OS UNIX*.
- O *Customer Information Control System Transaction Server* é chamado de *CICSTS*, abreviado como *CICS*.

Para releases anteriores, incluindo IBM WebSphere Developer para System z, IBM WebSphere Developer para zSeries e IBM® WebSphere Studio Enterprise Developer, use as informações de configuração encontradas no Guia de Configuração do Host e nos Diretórios de Programas desses releases.

---

## Quem Deve Usar este Documento

Este documento foi planejado para programadores de sistema que instalam e configuram o IBM Rational Developer para System z Versão 7.6.1, FMID HHOP760, em seus sistemas host z/OS.

Ele lista com detalhes as diferentes etapas necessárias para uma configuração completa do produto, incluindo alguns cenários não padrão. Para utilizar este documento, você precisa estar familiarizado com os Serviços do Sistema z/OS UNIX e os sistemas host MVS.

---

## Resumo das Mudanças

Esta seção resume as alterações do Guia de Configuração do Host do *Rational Developer para System z Versão 7.6.1*, S517-9094-04 (atualizado em Maio de 2010).

Mudanças técnicas e adições ao texto e ilustrações são indicadas por uma linha vertical à esquerda da mudança.

Este documento contém informações previamente apresentadas no Guia de Configuração do Host do *Rational Developer para System z Versão 7.6*, SC23-7658-03.

Novas informações:

- Correções e informações adicionais apresentadas nas Notas sobre o Release de Configuração do Host do *Rational Developer para System z v7.6* estão incorporadas.
- Notas de migração específicas da versão 7.6.1. Consulte o “Notas de Migração da Versão 7.6.1” na página 279.
- Visão geral do documento incluído. Consulte o “Descrição do Conteúdo do Documento” na página xiv.

- Suporte para Java Java™. Consulte o Apêndice E, “Requisitos”, na página 329.
- Configuração do produto através de painéis ISPF. Consulte o “Considerações de Pré-configuração” na página 8.
- Novas diretivas no rsed.envvars. Consulte o “Arquivo de Configuração do RSE rsed.envvars” na página 31.
- Novos membros proclib. Consulte o “Procedimentos de Construção Remota ELAXF\*” na página 25.
- Novo layout VSAM do CARMA. Consulte o “Notas de Migração VSAM do CARMA” na página 51.
- Suportar múltiplos RAMs do CARMA. Consulte o “(Opcional) Suportando Múltiplos RAMs” na página 67.
- Novas opções do Application Deployment Manager. Consulte o “Administrative Utility” na página 255.
- Novo layout VSAM do Application Deployment Manager. Consulte o “Notas de Migração do Utilitário Administrativo” na página 259
- Novos comandos do operador. Consulte o Capítulo 8, “Comandos do Operador”, na página 123.
- Novas mensagens do console. Consulte o “Mensagens do Console” na página 130
- Informações sobre o Gerenciamento de Carga de Trabalho. Consulte o Capítulo 12, “Considerações WLM”, na página 201.

---

## Descrição do Conteúdo do Documento

Esta seção resume as informações apresentadas neste documento.

### Planejamento

Use as informações deste capítulo para planejar a instalação e a implementação do Developer para System z.

### Customização Básica

As etapas de customização abaixo são para uma configuração básica do Developer para System z:

- Configuração de customização
- Alterações do PARMLIB
- alterações do PROCLIB
- Definições de segurança
- FEJCNFG, arquivo de configuração do JES Job Monitor
- rsed.envvars, arquivo de configuração do RSE
- ISPF.conf, Arquivo de configuração do ISPF's TSO/ISPF Client Gateway

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

O Common Access Repository Manager (CARMA) é um auxílio de produtividade para os desenvolvedores criam Repository Access Managers (RAMs). Um RAM é uma interface de programação de aplicativos (API) para z/OS baseada em Software Configuration Managers (SCMs).

Em seguida, os aplicativos gravados pelo usuário podem iniciar um servidor CARMA que carrega os RAMS(s) e fornece uma interface padrão para acessar o SCM.

O IBM® Rational® Developer para System z Interface para CA Endevor® Software Configuration Manager fornece aos clientes do Developer para System z acesso direto ao CA Endevor® SCM.

## **(Opcional) Application Deployment Manager**

O Developer para System z utiliza determinadas funções do Application Deployment Manager como uma abordagem de implementação comum para vários componentes. A customização opcional ativa mais recursos do Gerenciador de Implementação do Aplicativo e pode incluir os seguintes serviços no Developer para System z:

- O IBM CICS Explorer fornece uma infraestrutura baseada em Eclipse para visualizar e gerenciar recursos CICS e ativa maior integração entre as ferramentas CICS.
- O cliente e o servidor do CICS Resource Definition (CRD) fornecem as seguintes funções:
  - Editor do CICS Resource Definition
  - Permite que os desenvolvedores de aplicativos definam recursos CICS de forma limitada, controlada e segura.
  - Evita acesso no desenvolvimento do CICS a conjuntos de dados VSAM desautorizados ou incorretos, fornecendo ao administrador do CICS controle sobre o atributo físico do nome do conjunto de dados nas definições de Arquivo.
  - Diversos auxílios no desenvolvimento do CICS
  - Diversos auxílios no desenvolvimento de Serviço da Web do CICS

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

O SCLM Developer Toolkit fornece as ferramentas necessárias para estender os recursos do SCLM para o cliente. O SCLM por si só é um gerenciador de código de origem baseado em host enviado como parte do ISPF.

O SCLM Developer Toolkit possui um plug-in baseado em Eclipse que faz a interface com o SCLM e fornece acesso a todos os processos do SCLM para desenvolvimento de código legado, bem como suporte para o desenvolvimento completo do Java e do J2EE na estação de trabalho com sincronização para SCLM no mainframe, incluindo compilação, montagem e implementação do código J2EE a partir do mainframe.

## **(Opcional) Outras Tarefas de Customização**

Esta seção combina uma variedade de tarefas opcionais de customização. Siga as instruções na seção apropriada para configurar o serviço desejado.

- procedimento armazenado do DB2
- Suporte de Enterprise Service Tools (EST)
- suporte à linguagem bidirecional do CICS
- IRZ de diagnóstico, mensagens de erro
- criptografia SSL do RSE
- rastreamento RSE
- grupos de propriedades baseados no host
- Projetos Baseados em Host
- Integração do File Manager
- caracteres não editáveis

- Usando REXEC (ou SSH)
- transação APPC para o serviço TSO Commands
- limpeza da WORKAREA

## Verificação de Instalação

Após concluir a customização do produto, é possível usar os Installation Verification Programs (IVPs) descritos neste capítulo para verificar o êxito da configuração dos componentes principais do produto.

## Comandos do operador

Este capítulo fornece uma visão geral dos comandos disponíveis do operador (ou console) para o Developer para System z.

## Resolução de Problemas de Configuração

Este capítulo é fornecido para ajudá-lo com alguns problemas comuns que você pode encontrar durante a configuração do Developer para System z e contém as seguintes seções:

- Análise de log e de configuração usando FEKLOGS
- Arquivos de Log
- Arquivos de dump
- Rastreo
- Bits de permissão do z/OS UNIX
- Portas TCP/IP reservadas
- Tamanho do espaço de endereço
- Transação APPC e serviço TSO Commands
- Informações Variadas

## Considerações de segurança

O Developer para System fornece acesso do mainframe para usuários de uma estação de trabalho sem mainframe. A validação dos pedidos de conexão, o fornecimento de comunicação segura entre o host e a estação de trabalho, e a atividade de autorização e auditoria são aspectos importantes da configuração do produto.

## Entendendo o Developer para System z

O host Developer para System z consiste em vários componentes que interagem para fornecer ao cliente acesso aos dados e serviços do host. Entender o design desses componentes pode ajudar você a tomar as decisões de configuração corretas.

## Considerações WLM

Ao contrário dos aplicativos z/OS tradicionais, o Developer para System z não é um aplicativo monolítico que pode ser facilmente identificado para o Workload Manager (WLM). O Developer para System z consiste em diversos componentes que interagem para dar ao cliente acesso aos serviços e dados de host. Alguns destes serviços estão ativos em diferentes espaços de endereço, resultando em diferentes classificações de WLM.

## Considerações de Ajuste

O RSE (Explorador de Sistema Remoto) é o núcleo do Developer para System z. Para gerenciar as conexões e as cargas de trabalho a partir dos clientes, o RSE é formado por um espaço de endereço do daemon, que controla os espaços de endereços do conjunto de encadeamento. O daemon atua como um ponto focal para fins de conexão e gerenciamento, enquanto os conjuntos de encadeamento processam as cargas de trabalho do cliente.

Isso torna o RSE um alvo principal para o ajuste da configuração do Developer para System z. Entretanto, a manutenção de centenas de usuários, cada um usando 16 ou mais encadeamentos, uma determinada quantidade de armazenamento e possivelmente um ou mais espaços de endereço exige configuração adequada do Developer para System z e do z/OS.

## Considerações de Desempenho

z/OS é um sistema operacional altamente customizável, e (às vezes pequenas) mudanças no sistema podem ter um enorme impacto no desempenho geral. Este capítulo destaca algumas das mudanças que podem ser feitas para melhorar o desempenho do Developer para System z.

## Considerações sobre o CICSTS

Este capítulo contém informações úteis para um administrador do CICS Transaction Server.

## Customizando o Ambiente TSO

Este capítulo ajuda você a imitar um procedimento de logon do TSO incluindo instruções DD e conjuntos de dados no ambiente do TSO no Developer para System z.

## Executando várias instâncias

Há vezes em que você quer várias instâncias do Developer para System z ativas no mesmo sistema, por exemplo, ao testar um upgrade. Entretanto, alguns recursos, como portas TCP/IP, não podem ser compartilhados, portanto, os padrões nem sempre são aplicáveis. Use as informações neste capítulo para planejar a coexistência de instâncias diferentes do Developer para System z, após a qual você poderá usar esse guia de configuração para customizá-las.

## Guia de Migração

Esta seção destaca as alterações na instalação e na configuração em comparação a releases anteriores do produto. Também fornece algumas diretrizes gerais para migrar para este release. Consulte as seções relacionadas neste manual para obter informações adicionais.

## Configurando o SSL e a Autenticação X.509

Este apêndice é fornecido para ajudar você com alguns dos problemas comuns que você pode encontrar ao configurar Secure Socket Layer (SSL) ou durante a verificação ou modificação de uma configuração existente. Este apêndice também fornece uma configuração de amostra para dar suporte aos usuários se autenticando com um certificado X.509.

## **Configurando o TCP/IP**

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

## **Configurando o INETD**

Este apêndice é fornecido para ajudá-lo com alguns problemas comuns que você pode encontrar ao configurar o INETD ou durante a verificação ou a modificação de uma configuração existente. INETD é usado pelo Developer para System z para a funcionalidade REXEC/SSH.

## **Configurando o APPC**

Este apêndice é fornecido para ajudá-lo com alguns problemas comuns que você pode encontrar ao configurar o APPC (Advanced Program-to-Program Communication) ou durante a verificação e a modificação de uma configuração existente.

## **Requisitos**

Este apêndice lista os pré-requisitos e co-requisitos do host para esta versão do Developer para System z.



---

## **Parte 1. Customização do Developer para System z**



---

## Capítulo 1. Planejamento

Use as informações deste capítulo, junto com as informações do Apêndice E, “Requisitos”, na página 329, para planejar a instalação e implementação do Developer para System z. Os seguintes assuntos são descritos:

- “Considerações de Migração”
- “Considerações de Planejamento”
- “Considerações sobre Pré-Instalação” na página 4
- “Considerações de Pré-configuração” na página 8
- “Considerações sobre Pré-implementação” na página 11
- “Lista de Verificação do Cliente” na página 12

---

### Considerações de Migração

O Capítulo 18, “Guia de Migração”, na página 277 descreve as alterações de instalação e configuração comparadas aos releases anteriores do produto. Use essas informações para planejar a migração para o release atual do Developer para System z.

#### Notas:

1. Se você for um usuário anterior do IBM Rational Developer para System z, IBM WebSphere Developer para System z, IBM WebSphere Developer para zSeries ou IBM WebSphere Studio Enterprise Developer, é recomendável salvar os arquivos customizados relacionados ANTES de instalar o IBM Rational Developer para System z Versão 7.6.1. Consulte o Capítulo 18, “Guia de Migração”, na página 277 para obter uma visão geral dos arquivos que precisam ser customizados.
2. Consulte o Capítulo 17, “Executando Várias Instâncias”, na página 271, se planeja executar várias instâncias do Developer para System z.

---

### Considerações de Planejamento

#### Visão Geral do Produto

O Developer para System z consiste em um cliente, instalado no computador pessoal do usuário, e em um servidor, instalado em um ou mais hosts. Esta documentação enfocará no host que está sendo um sistema z/OS. Porém, outros sistemas operacionais como AIX e Linux® no System z também são suportados.

O cliente fornece aos desenvolvedores um ambiente de desenvolvimento baseado no Eclipse que facilita uma interface gráfica uniforme com o host, e que, entre outras coisas, pode transferir trabalho do host para o cliente, economizando recursos no host.

A parte do host consiste em várias tarefas ativas permanentemente e tarefas que são iniciadas ad-hoc. Essas tarefas permitem que o cliente trabalhe com os vários componentes do seu host z/OS, tais como conjuntos de dados MVS, comandos TSO, arquivos e comandos z/OS UNIX, envio de tarefas e saída de tarefas.

O Developer para System z pode também interagir com subsistemas e outros softwares aplicativos no host, tal como o CICS, a Ferramenta de Depuração e

SCMs (Software Configuration Managers), se o Developer para System z estiver configurado para isso e se esses produtos (correquisitos) estiverem disponíveis.

Consulte o Capítulo 11, “Entendimento do Developer para System z”, na página 189 para obter um entendimento básico do design do Developer para System z.

Consulte o Web site do Developer para System z, <http://www-01.ibm.com/software/awdtools/rdz/>, ou o seu representante local IBM para saber mais sobre a funcionalidade oferecida pelo Developer para System z.

## Exigências de Capacidade

Capacidades mínimas de SMP/E são necessárias para a instalação de um host do Developer para System z.

A configuração do Developer para System z exige mais do que as permissões e o conhecimento típicos de programação de sistema, portanto, é possível que haja necessidade de uma mínima assistência de terceiros. A Tabela 3 na página 7 e a Tabela 4 na página 7 listam os administradores necessários para as tarefas de customização necessárias e opcionais.

## Requisitos de Tempo

O tempo necessário para instalar e configurar os componentes de host do Developer para System z depende de vários fatores, tais como:

- a configuração atual do z/OS UNIX e de TCP/IP
- a disponibilidade de software obrigatório e de manutenção
- se os segmentos OMVS estão ou não definidos para usuários do Developer para System z
- a disponibilidade do usuário, que instalou com êxito o cliente, para testar a instalação e relatar quaisquer problemas que possam ocorrer

A experiência tem mostrado que o processo de instalação e de configuração do host do Developer para System z exige de um a quatro dias para ser concluído. Esse requisito de tempo é para que uma instalação limpa seja executada por um programador de sistema experiente. Se ocorrerem problemas ou se as qualificações necessárias não estiverem disponíveis, então a configuração demorará mais tempo.

---

## Considerações sobre Pré-Instalação

Consulte o *Program Directory para IBM Rational Developer para System z* (GI11-8298) para obter instruções detalhadas sobre a instalação SMP/E do produto.

**Nota:** O sistema de arquivos (HFS ou zFS) em que o Developer para System z está instalado deve ser montado com o bit de permissão SETUID ativado (este é o padrão do sistema). A montagem do sistema de arquivos com o parâmetro NOSETUID impedirá o Developer para System z de criar o ambiente de segurança do usuário e falhará no pedido de conexão do cliente.

Consulte o Capítulo 17, “Executando Várias Instâncias”, na página 271 se desejar executar várias instâncias do Developer para System z.

## Opção de Configuração

O Developer para System z fornece uma opção para acessar o serviço TSO Commands. A opção escolhida aqui impacta a configuração necessária de pré-requisitos. Um dos seguintes métodos deve ser selecionado e configurado:

- O serviço TSO/ISPF Client Gateway do ISPF, que requer um nível mínimo de serviço do ISPF. Este é o método padrão usado nas amostras fornecidas.
- Uma transação APPC (como em releases pré-versão 7.1)

**Nota:** O TSO/ISPF Client Gateway do ISPF também é usado pelo SCLM Developer Toolkit e, opcionalmente, por um método de inicialização alternativo para o CARMA (Common Access Repository Manager).

## Requisito de produtos

O Apêndice E, “Requisitos”, na página 329 possui uma lista de softwares obrigatórios que devem ser instalados e estar em funcionamento para que o Developer para System z funcione. Há também uma lista de software de correquisito para suportar recursos específicos do Developer para 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 *Rational Developer para System z Prerequisites* (SC23-7659) na biblioteca on-line do Developer para System z, em <http://www-01.ibm.com/software/awdtools/rdz/library/> para obter uma lista atualizada de produtos obrigatórios e de correquisitos para a sua versão do Developer para System z. Planeje antecipadamente para ter esses produtos de requisito disponíveis, uma vez que isso pode levar algum tempo, dependendo das políticas em seu site. Os principais requisitos para uma configuração básica são os seguintes:

- z/OS 1.8 ou superior
- ISPF APAR OA29489 (TSO/ISPF Client Gateway)
- Java 5.0 ou versão mais recente

**Nota:** O PTF APAR PM073 para o Developer para System z deve ser aplicado ao usar uma versão 64 bits do Java. O PTF está disponível por meio da página de serviço recomendada do Developer para System z, <http://www-01.ibm.com/support/docview.wss?rs=2294&context=SS2QJ2&uid=swg27006335>.

## Recursos Necessários

O Developer para System z requer a alocação dos recursos do sistema listados na Tabela 1 na página 6. Os recursos listados na Tabela 2 na página 6 são necessários para serviços opcionais. Planeje antecipadamente para ter esses recursos disponíveis, uma vez que pode levar algum tempo para obtê-los, dependendo das políticas em seu site.

**Nota:** O Developer para System z consiste em múltiplas tarefas que se comunicam entre si e o cliente. Essas tarefas usam vários cronômetros para detectar perda de comunicação com seu parceiro ou seus parceiros. Isso significa que podem surgir problemas de tempo limite (devido à falta de tempo de CPU durante a janela de tempo limite) nos sistemas com uma carga de CPU pesada ou configurações incorretas de Gerenciamento de Carga de Trabalho (WLM) no Developer para system z.

*Tabela 1. Recursos Necessários*

Recurso	Valor Padrão	Informações
Conjunto de dados autorizado APF	FEK.SFEKAUTH	"Autorizações APF em PROGxx" na página 18
tarefa iniciada	JMON, RSED e LOCKD	"Considerações do Servidor" na página 10
porta para uso limitado por host	6715	"FEJCNFG, Arquivo de Configuração do JES Job Monitor" na página 27
porta para uso limitado por host	4036	"Arquivo de Configuração do RSE rsed.envvars" na página 31
porta para comunicação cliente-host	4035	"Alterações do PROCLIB" na página 21
intervalo de portas para comunicação cliente-host	qualquer porta disponível é usada	"Definindo o PORTRANGE Disponível para o Servidor RSE" na página 41
Definição da segurança do aplicativo	Acesso universal de LEITURA para FEKAPPL	"Definir Proteção de Aplicativo para RSE" na página 185
Definições de segurança PassTicket	nenhum padrão	"Definir o Suporte PassTicket para o RSE" na página 185

*Tabela 2. Recursos Opcionais*

Recurso	Valor Padrão	Informações
conjunto de dados LINKLIST	FEK.SFEKAUTH e FEK.SFEKLOAD	Capítulo 5, "(Opcional) SCLM Developer Toolkit", na página 79
conjunto de dados LPA	FEK.SFEKLPA	Capítulo 3, "(Opcional) CARMA (Common Access Repository Manager)", na página 49
intervalo de portas para uso confinado no host	5227-5326 (100 portas)	Capítulo 3, "(Opcional) CARMA (Common Access Repository Manager)", na página 49
portas para uso limitado por host	qualquer porta disponível é usada	"(Opcional) Transação APPC para o Serviço de Comandos TSO" na página 102
porta para comunicação cliente-host	nenhum padrão	Capítulo 4, "(Opcional) Application Deployment Manager", na página 71
atualização da CSD do CICS	diversos valores	Capítulo 4, "(Opcional) Application Deployment Manager", na página 71
atualização do CICS JCL	FEK.SFEKLOAD	<ul style="list-style-type: none"> <li>Capítulo 4, "(Opcional) Application Deployment Manager", na página 71</li> <li>"(Opcional) Suporte de Idioma Bidirecional do CICS" na página 90</li> </ul>



A configuração do Developer para System z exige mais do que as permissões e o conhecimento típicos de programação de sistema, portanto, é possível que haja necessidade de uma mínima assistência de terceiros. A Tabela 3 e a Tabela 4 listam os administradores necessários para as tarefas de customização necessárias e opcionais.

*Tabela 3. Administradores necessários para as tarefas necessárias*

Administrador	Tarefa	Informações
Sistema	Ações típicas do programador de sistema são necessárias para todas as tarefas de customização	N/D
Security	<ul style="list-style-type: none"> <li>Definir segmento OMVS para usuários do Developer para System z</li> <li>Definir perfis do conjunto de dados</li> <li>Definir tarefas iniciadas</li> <li>Definir segurança de comando do operador</li> <li>Definir os perfis do servidor z/OS UNIX</li> <li>Definir segurança do aplicativo</li> <li>Definir suporte PassTicket</li> <li>Definir conjuntos de dados controlados pelo programa</li> <li>Definir os arquivos do z/OS UNIX controlados pelo programa</li> </ul>	Capítulo 10, “Considerações sobre Segurança”, na página 157
TCP/IP	Definir novas portas TCP/IP	“Portas TCP/IP” na página 160
WLM	Designar os objetivos da tarefa iniciada para os processos de servidores e seus filhos	Capítulo 12, “Considerações WLM”, na página 201

*Tabela 4. Administradores Necessários para Tarefas Opcionais*

Administrador	Tarefa	Informações
Sistema	Ações típicas do programador de sistema são necessárias para todas as tarefas de customização	N/D

Tabela 4. Administradores Necessários para Tarefas Opcionais (continuação)

Administrador	Tarefa	Informações
Security	<ul style="list-style-type: none"> <li>Definir perfis do conjunto de dados</li> <li>Definir conjuntos de dados controlados pelo programa</li> <li>Definir permissão para enviar tarefas xxx*</li> <li>Definir segurança da transação do CICS</li> <li>Incluir certificado para SSL</li> <li>Definir suporte de certificado de cliente X.509</li> </ul>	<ul style="list-style-type: none"> <li>Capítulo 10, “Considerações sobre Segurança”, na página 157</li> <li>Capítulo 15, “Considerações sobre o CICSTS”, na página 251</li> <li>Apêndice A, “Configurando o SSL e a Autenticação X.509”, na página 293</li> </ul>
TCP/IP	Definir novas portas TCP/IP	“Portas TCP/IP” na página 160
SCLM	<ul style="list-style-type: none"> <li>Definir tradutores de idiomas SCLM para suporte JAVA/J2EE</li> <li>Definir tipos de SCLM para suporte JAVA/J2EE</li> </ul>	Capítulo 5, “(Opcional) SCLM Developer Toolkit”, na página 79
CICS TS	<ul style="list-style-type: none"> <li>Atualização de JCL da região do CICS</li> <li>Atualização da CSD da região do CICS</li> <li>Definir grupo do CICS</li> <li>Definir nomes de transação do CICS</li> <li>Definir um programa para o CICS</li> </ul>	<ul style="list-style-type: none"> <li>Capítulo 4, “(Opcional) Application Deployment Manager”, na página 71</li> <li>“(Opcional) Suporte de Idioma Bidirecional do CICS” na página 90</li> </ul>
DB2	Definir um procedimento armazenado do DB2	“(Opcional) Procedimento Armazenado do DB2” na página 87
WLM	<ul style="list-style-type: none"> <li>Designar metas a um procedimento armazenado do DB2</li> <li>Designar metas do tipo TSO para uma transação APPC</li> </ul>	<ul style="list-style-type: none"> <li>“(Opcional) Procedimento Armazenado do DB2” na página 87</li> <li>Capítulo 12, “Considerações WLM”, na página 201</li> </ul>
APPC	Definir uma transação APPC	“(Opcional) Transação APPC para o Serviço de Comandos TSO” na página 102

## Considerações de Pré-configuração

### Gerenciamento de Carga de Trabalho

Ao contrário dos aplicativos tradicionais do z/OS, o Developer para System z não é um aplicativo monolítico que pode ser facilmente identificado para Workload Manager (WLM). O Developer para System z consiste de vários componentes que

interagem para fornecer ao cliente acesso aos dados e serviços do host. Consulte o Capítulo 12, “Considerações WLM”, na página 201 para planejar apropriadamente a sua configuração WLM.

## Uso de Recursos e Limites do Sistema

Quando estiver em uso, o Developer para System z usará um número variável de recursos do sistema, como espaços de endereços e processos e encadeamentos z/OS UNIX. A disponibilidade desses recursos é limitada por várias definições do sistema. Consulte o Capítulo 13, “Considerações de Ajuste”, na página 211 para estimar o uso de recursos-chave, para que você possa planejar a configuração do sistema de acordo.

## Configuração Necessária de Produtos 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. Algumas tarefas de customização de requisito que são facilmente negligenciadas estão listadas a seguir:

- Todos os usuários do Developer para System z devem ter acesso READ e EXECUTE aos diretórios Java.
- Todos os usuários do Developer para System z devem ter acesso READ, WRITE e EXECUTE ao diretório /tmp/.
- Ações remotas (baseadas em host) para subprojetos do z/OS UNIX requerem que a versão z/OS UNIX do REXEC ou SSH esteja ativada no host.

## Considerações de ID do Usuário

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

- Acesso ao TSO (com um tamanho de região normal).

**Nota:** Um tamanho de região grande é necessário para o ID do usuário que executa os IVPs (Programas de Verificação da Instalação), pois as funções exigem muita memória (como Java) serão executadas. Você deve definir o tamanho da região como 131072 kilobytes (128 megabytes) ou mais alto.

- Um segmento OMVS definido para o sistema de segurança (por exemplo, RACF), ambos para o ID do usuário e seu grupo padrão.
  - O campo HOME deve referir-se a um diretório inicial alocado para o usuário (com acesso READ, WRITE e EXECUTE).
  - O campo PROGRAM no segmento OMVS deve ser /bin/sh ou outro shell z/OS UNIX válido, como /bin/tcsh.
  - O campo ASSIZEMAX não deve ser definido, para esses padrões do sistema serão usados.
  - 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
```

```
FILEPROCMAx= NONE
PROCUSERMAx= NONE
THREADSMAx= NONE
MMAPAREAMAx= NONE
```

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

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

```
GROUP group
```

```
OMVS INFORMATION
```

```
-----
GID= 0000003243
```

- Acesso READ e EXECUTE aos diretórios e arquivos de instalação e configuração do Developer para System z, padrão /usr/lpp/rdz/\*, /etc/rdz/\* e /var/rdz/\*.
- Acesso READ, WRITE e EXECUTE ao diretório Developer para System z WORKAREA, padrão /var/rdz/WORKAREA.
- Acesso READ aos conjuntos de dados de instalação do Developer para System z, padrão FEK.SFEK\*.

## Considerações do Servidor

O Developer para System z consiste em 3 servidores ativos permanentemente, que podem ser tarefas iniciadas ou tarefas do usuário. Esses servidores fornecem os serviços necessários entre si ou iniciam outros servidores (como encadeamentos z/OS UNIX ou tarefas do usuário) para fornecer o serviço.

- O JES Job Monitor (JMON) fornece todos os serviços relacionados ao JES.
- O Daemon de Bloqueio (LOCKD) fornece serviços de rastreamento para bloqueios de conjuntos de dados.
- O RSE (Explorador de Sistemas Remotos) fornece os serviços principais, como conectar o cliente ao host e iniciar outros servidores para serviços específicos. O RSE consiste em 2 entidades lógicas:
  - O Daemon RSE (RSED), que gerencia a configuração da conexão e que é responsável pela execução em modo de servidor único.
  - Servidor RSE, que manipula pedidos individuais do cliente.

O JES Job Monitor (JMON) fornece todos os serviços relacionados ao JES.

- Os mecanismos de segurança usados pelo JES Job Monitor dependem de os conjuntos de dados em que eles residem serem seguros. Isso significa que apenas os administradores de sistemas confiáveis deverão estar aptos a atualizarem as bibliotecas e os arquivos de configuração.

Explorador de Sistemas Remotos O (RSE) é o componente do Developer para System z que fornece serviços principais, como conectar o cliente ao host.

- Desde a versão 7.5, o daemon RSE não é mais um processo gerenciado pelo INETD, mas uma tarefa iniciada.
- Desde a versão 7.5, o servidor RSE utiliza um único modelo de servidor, enquanto em versões anteriores, cada conexão entre cliente e host tinha um servidor RSE privado.
- Diferentes níveis de segurança de comunicação são suportados pelo RSE:
  - A comunicação externa (cliente-host) pode ser limitada a portas especificadas. Esse recurso é desativado por padrão.
  - A comunicação externa (cliente-host) pode ser criptografada utilizando SSL. Esse recurso é desativado por padrão.

- A verificação de Port Of Entry (POE) pode ser utilizada para permitir acesso apenas a endereços TCP/IP confiáveis. Esse recurso é desativado por padrão.
- O RSE também suporta vários métodos de autenticação de cliente:
  - ID do Usuário e Senha
  - ID do Usuário e Senha Única
  - Certificado X.509
- Os mecanismos de segurança usados pelo RSE dependem de o sistema de arquivo em que eles residem ser seguro. Isso significa que apenas os administradores de sistemas confiáveis deverão estar aptos a atualizarem as bibliotecas e os arquivos de configuração.

Conforme documentado no “Portas TCP/IP” na página 160, certos serviços do host, e suas portas, devem estar disponíveis para conexão com o cliente e devem ser definidos para seu firewall que protege o host. Todas as outras portas usadas pelo Developer para System z possuem tráfego apenas do host. A seguir há uma lista das portas necessárias para uma configuração básica do Developer para System z.

- Daemon RSE para a configuração da comunicação de cliente-host (utilizando o protocolo tcp), porta padrão 4035.
- Servidor RSE para comunicação de cliente-host (utilizando o protocolo tcp). Por padrão, qualquer porta disponível é usada, mas isto pode ser limitado a um intervalo específico.

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

## Método de Configuração

Começando com a versão 7.6.1, o Developer para System z fornece um método alternativo, usando um aplicativo do painel ISPF, para configurar o lado de host do produto. Isso fornece a você uma escolha dos seguintes métodos:

- Usar um aplicativo do painel ISPF. Isto o guia através das etapas de customização requeridas e etapas de customização opcionais. Para obter informações adicionais, consulte a White Paper do *Utilitário de Configuração do Host*, disponível na biblioteca da Internet do Developer para System z, <http://www-306.ibm.com/software/awdtools/rdz/library/>.
- Usar o *O Guia de Iniciação Rápida da Configuração do Host*. Isso o guia através das etapas de customização requeridas. O escopo deste guia é limitado a uma configuração básica.
- Usar o *Guia de Configuração do Host*. Isso o guia através das etapas de customização requeridas e todas as etapas de customização opcionais. Todas as opções configuráveis são cobertas neste guia, incluindo alguns cenários não padrão

---

## Considerações sobre Pré-implementação

O Developer para System z suporta a clonagem de uma instalação para um sistema diferente, evitando a necessidade de instalar o SMP/E em cada sistema.

Os seguintes conjuntos de dados, diretórios e arquivos são obrigatórios para a implementação em outros sistemas. Se você copiou um arquivo para um local diferente, esse arquivo deverá substituir seu correspondente nas listas a seguir.

**Nota:** A lista a seguir não inclui as necessidades de implementação do software obrigatório e de correquisito.

- FEK.SFEKAUTH(\*)
- FEK.SFEKLOAD(\*)
- FEK.SFEKPROC(\*)
- FEK.#CUST.PARMLIB(\*)
- FEK.#CUST.PROCLIB(\*)
- /usr/lpp/rdz/\*
- /etc/rdz/\*
- /var/rdz/\* (apenas estrutura do diretório)
- partes opcionais:
  - FEK.SFEKLPA(\*)
  - FEK.#CUST.CNTL(\*)
  - definições, conjuntos de dados, arquivos e diretórios resultantes das tarefas de customização em FEK.#CUST.JCL

**Notas:**

1. FEK e /usr/lpp/rdz são o qualificador e o caminho de alto nível usados durante a instalação do produto. FEK.#CUST, /etc/rdz e /var/rdz são os locais padrão usados durante a customização do produto (consulte “Configuração da Customização” na página 15 para obter informações adicionais).
2. Você deve instalar o Developer para System z em um sistema de arquivos privado (HFS ou zFS) para facilitar a implementação e partes do produto z/OS UNIX.
3. Se você não puder usar um sistema de arquivos privado, deverá usar uma ferramenta de arquivamento, como o comando z/OS UNIX tar para transportar os diretórios do z/OS UNIX de sistema para sistema. Isso para preservar os atributos (como controle de programas) para os arquivos e diretórios do Developer para System z.

Consulte *UNIX System Services Command Reference* (SA22-7802) para obter informações adicionais sobre os seguintes comandos de amostra para arquivar e restaurar o diretório de instalação do Developer para System z.

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

---

## Lista de Verificação do Cliente

Os usuários do cliente do Developer para System z devem conhecer o resultado de determinadas customizações do host, como números de portas TCP/IP, para que o cliente funcione corretamente. Use essas listas de verificação para reunir as informações necessárias.

A lista de verificação na Tabela 5 na página 13 lista os resultados necessários de etapas de customização obrigatórias. A Tabela 6 na página 13 lista os resultados necessários das etapas de customização opcionais.



*Tabela 5. Lista de Verificação do Cliente - Partes Obrigatórias*

Customização	Valor
Número da porta do servidor do JES Job Monitor (padrão 6715):  Consulte SERV_PORT em “FEJJCNFG, Arquivo de Configuração do JES Job Monitor” na página 27.	
Número da porta TCP/IP do daemon RSE (padrão 4035):  Consulte o “Daemon RSE” na página 22.	

*Tabela 6. Lista de Verificação do Cliente - Partes Opcionais*

Customização	Valor
Local dos procedimentos de ELAXF* se não estiverem em uma biblioteca de procedimentos do sistema:  Consulte a nota sobre JCLLIB em “Procedimentos de Construção Remota ELAXF*” na página 25.	
Procedimento ou nomes de etapas dos procedimentos de ELAXF* se tiverem sido alterados:  Consulte a nota sobre a alteração deles em “Procedimentos de Construção Remota ELAXF*” na página 25.	
Nome do procedimento armazenado do DB2 (padrão ELAXMSAM):  Consulte informações sobre procedimentos armazenados do DB2 no Capítulo 17, “Executando Várias Instâncias”, na página 271.	
Local do procedimento armazenado do DB2, se não estiver em uma biblioteca de procedimentos do sistema:  Consulte o “(Opcional) Procedimento Armazenado do DB2” na página 87.	
(correquiso) Número da porta TN3270 para Emulador de Conexão do Host (padrão 23).  Consulte o Capítulo 10, “Considerações sobre Segurança”, na página 157	
(correquiso) Número da porta REXEC ou SSH (padrão 512 ou 22, respectivamente):  Consulte o “(Opcional) Usando REXEC (ou SSH)” na página 100.	
Local do arquivo server.zseries se o método de conexão REXEC/SSH for usado (padrão /etc/rdz).  Consulte o “(Opcional) Usando REXEC (ou SSH)” na página 100.	
Local do CRA#ASLM JCL para alocações de conjuntos de dados SCLM RAM do CARMA (padrão FEK.#CUST.JCL):  Consulte a nota sobre CRA#ASLM em “Ativando o RAM do SCLM” na página 61.	



---

## Capítulo 2. Customização Básica

As etapas de customização a seguir são para uma configuração básica do Developer para System z. Consulte os capítulos sobre os componentes opcionais para seus requisitos de customização.

---

### Requisitos e Lista de Verificação

Você precisará da assistência de um administrador de segurança e de um administrador de TCP/IP para concluir essa tarefa de customização, que requer os seguintes recursos e tarefas de customização especiais:

- Conjunto de dados autorizado APF
- Várias atualizações PARMLIB
- Várias atualizações de software de segurança
- Várias portas TCP/IP para comunicação interna e de cliente-host

Para verificar a instalação e começar a usar o Developer para System z no seu site, você deve executar as seguintes tarefas. A menos que especificado o contrário, todas as tarefas são obrigatórias.

1. Crie cópias customizáveis de amostras e crie o ambiente de trabalho para o Developer para system z. Para obter detalhes, consulte “Configuração da Customização”.
2. Atualize os limites do sistema z/OS UNIX, inicie as tarefas iniciadas, defina os conjuntos de dados LINKLIST e autorizados por APF e, opcionalmente, os conjuntos de dados LPA. Para obter detalhes, consulte “Alterações PARMLIB” na página 16.
3. Crie procedimentos de tarefa iniciada e procedimentos de compilação/link. Para obter detalhes, consulte “Alterações do PROCLIB” na página 21.
4. Atualize as definições de segurança. Para obter detalhes, consulte “Definições de Segurança” na página 26. Você deve também estar ciente e compreender como os PassTickets são usados para estabelecer a segurança do encadeamento. Consulte “Usando os PassTickets” na página 162 para obter detalhes.
5. Customize os arquivos de configuração do Developer para System z. Para obter detalhes, consulte:
  - “FEJCNFG, Arquivo de Configuração do JES Job Monitor” na página 27
  - “Arquivo de Configuração do RSE rsed.envvars” na página 31
  - “ISPF.conf, Arquivo de Configuração do TSO/ISPF Client Gateway do ISPF” na página 47

---

### Configuração da Customização

O Developer para System z é fornecido com vários arquivos de configuração de amostra e a JCL de amostra. Para evitar sobrescrever suas customizações ao realizar manutenção, é aconselhável copiar todos esses membros e arquivos z/OS UNIX para um local diferente e customizar a cópia.

Algumas funções do Developer para System z também requerem a existência de determinados diretórios no z/OS UNIX, que devem ser criados durante a customização do produto. Para facilitar o esforço de instalação, uma tarefa de amostra, FEKSETUP, é fornecida para criar as cópias e os requisitos necessários.

Customize e envie o membro de amostra FEKSETUP no conjunto de dados FEK.SFEKSAMP para criar cópias customizáveis de arquivos de configuração e JCL de configuração e para criar diretórios do z/OS UNIX necessários. As etapas necessárias de customização são descritas dentro do membro.

Esta tarefa executa as seguintes ações:

- Criar FEK.#CUST.PARMLIB e preenchê-lo com arquivos de configuração de amostra.
- Criar FEK.#CUST.PROCLIB e preenchê-lo com membros SYS1.PROCLIB de amostra.
- Criar FEK.#CUST.JCL e preenchê-lo com JCL de configuração de amostra.
- Criar FEK.#CUST.CNTL e preencher com scripts de inicialização do servidor de amostra.
- Criar FEK.#CUST.ASM e preenchê-lo com o código-fonte do assembler de amostra.
- Criar FEK.#CUST.COBOL e preenchê-lo com o código de origem do COBOL de amostra.
- Criar /etc/rdz/\* e preencher com arquivos de configuração de amostra.
- Criar /var/rdz/\* como diretórios de trabalho para várias funções do Developer para System z.

#### Notas:

1. As etapas de configuração nesta publicação usam os locais de membro/arquivo criados pela tarefa FEKSETUP, a menos que indicado de outra maneira. As amostras originais, que não devem ser atualizadas, podem ser localizadas em FEK.SFEKSAMP e /usr/lpp/rdz/samples/.
2. Se desejar manter todos os arquivos Developer para System z z/OS UNIX no mesmo sistema de arquivo (HFS ou zFS), e também desejar colocar os arquivos de configuração no /etc/rdz, utilize os links simbólicos para resolver este problema. Os seguintes comandos de amostra z/OS UNIX criam um novo diretório no sistema de arquivo existente (/usr/lpp/rdz/cust) e definem um link simbólico (/etc/rdz) para ele:

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

---

## Alterações PARMLIB

Consulte *MVS Referência de Ajuste e Inicialização* (SA22-7592) para obter informações adicionais sobre as definições PARMLIB listadas a seguir. Consulte o *MVS System Commands* (SA22-7627) para obter informações adicionais sobre os comandos do console de amostra.

## Configurar Limites do z/OS UNIX no BPXPRMxx

O Explorador de Sistemas Remotos (RSE), que fornece os serviços principais, como conexão do cliente com o host, é um processo baseado no z/OS UNIX. Portanto, é importante configurar valores corretos para os limites do sistema z/OS UNIX no BPXPRMxx, com base no número de usuários simultaneamente ativos do Developer para System z e em sua carga de trabalho média.

Consulte o Capítulo 13, “Considerações de Ajuste”, na página 211 para obter mais informações sobre os diferentes limites definidos por BPXPRMxx e seu impacto no Developer para System z.

MAXASSIZE especifica o tamanho máximo da região do espaço de endereço (processo). Configure MAXASSIZE em SYS1.PARMLIB(BPXPRMxx) como 2G. Esse é o

valor máximo permitido. Esse é um limite amplo do sistema e, dessa forma, ativo em todos os espaços de endereço do z/OS UNIX. Se isso não for o que você deseja, então poderá definir o limite também apenas para o Developer para System z em seu software de segurança, conforme descrito em “Definir as Tarefas Iniciadas do Developer para System z” na página 180.

MAXTHREADS especifica o número máximo de encadeamentos ativos para um único processo. Configure MAXTHREADS em SYS1.PARMLIB(BPXPRMxx) como 1500 ou mais alto. Esse é um limite amplo do sistema e, dessa forma, ativo em todos os espaços de endereço do z/OS UNIX. Se isso não for o que você deseja, então poderá definir o limite também apenas para o Developer para System z em seu software de segurança, conforme descrito em “Definir as Tarefas Iniciadas do Developer para System z” na página 180.

MAXTHREADTASKS especifica o número máximo de tarefas MVS ativas para um único processo. Configure MAXTHREADTASKS em SYS1.PARMLIB(BPXPRMxx) como 1500 ou mais alto. Esse é um limite amplo do sistema e, dessa forma, ativo em todos os espaços de endereço do z/OS UNIX. Se isso não for o que você deseja, então poderá definir o limite também apenas para o Developer para System z em seu software de segurança, conforme descrito em “Definir as Tarefas Iniciadas do Developer para System z” na página 180.

MAXPROCUSER especifica o número máximo de processos que um único ID de usuário do z/OS UNIX pode ter simultaneamente ativo. Configure MAXPROCUSER em SYS1.PARMLIB(BPXPRMxx) como 50 ou superior. Essa configuração deve ser um limite amplo do sistema, uma vez que deve estar ativo para cada cliente que usa o Developer para System z.

Esses valores podem ser verificados e configurados dinamicamente (até o próximo IPL) com os seguintes comandos do console:

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

#### Notas:

1. Consulte o “Tamanho do Espaço de Endereço” na página 150 para obter informações adicionais sobre outros locais em que tamanhos de espaço de endereço possam ser definidos ou limitados.
2. O valor de MAXPROCUSER usado acima baseia-se nos usuários que têm um UID (ID do Usuário) exclusivo do z/OS UNIX. Aumente esse valor se os usuários compartilharem o mesmo UID.
3. Certifique-se de que outros valores de BPXPRMxx, como aqueles para MAXPROCSYS e MAXUIDS, são suficientes para manipular a quantidade esperada de usuários do Developer para System z atualmente ativos. Consulte Capítulo 13, “Considerações de Ajuste”, na página 211 para obter mais detalhes.

## Incluir Tarefas Iniciadas em COMMNDxx

Inclua comandos de inicialização para os servidores Developer para System z RSED, LOCKD e JMON para SYS1.PARMLIB(COMMANDxx) para iniciá-los automaticamente no próximo sistema IPL.

Após os servidores serem definidos e configurados, eles podem ser iniciados dinamicamente (até o próximo IPL) com os seguintes comandos do console:

- S RSED
- S LOCKD
- S JMON

**Nota:** O daemon de bloqueio deve ser iniciado antes de usuários do Developer para System z efetuarem logon no daemon RSE. Isso é para que o daemon de bloqueio possa rastrear os pedidos de bloqueio do conjunto de dados por esses usuários. Portanto, é necessário iniciar o daemon de bloqueio na inicialização do sistema.

## Definições de LPA em LPALSTxx

O serviço Common Access Repository Manager (CARMA) (opcional) suporta métodos de inicialização de servidor alternativos que não requerem o uso de um iniciador JES. A alternativa mais flexível requer que o módulo CRASTART na biblioteca de carregamento FEK.SFEKLPA esteja na LPA (Área do Pacote de Links).

Os conjuntos de dados de LPA são definidos em SYS1.PARMLIB(LPALSTxx).

As definições de LPA podem ser configuradas dinamicamente (até o próximo IPL) com os seguintes comandos do console:

- SETPROG LPA,ADD,DSN=FEK.SFEKLPA

## Autorizações APF em PROGxx

Para que o JES Job Monitor acesse arquivos de spool do JES, o módulo FEJMON na biblioteca de carregamento FEK.SFEKAUTH e nas bibliotecas de tempo de execução LE (Language Environment) (CEE.SCEERUN\*) deve ser autorizado por APF.

Para que o serviço SCLM Developer Toolkit (opcional) funcione, o módulo BWBTSOW na biblioteca de carregamento FEK.SFEKAUTH e na biblioteca de tempo de execução do REXX (REXX.\*.SEAGLPA) deve ser autorizado por APF.

Para que o ISPF crie o Gateway do Cliente TSO/ISPF, o módulo ISPZTS0 em SYS1.LINKLIB deve ser autorizado pela APF. O TSO/ISPF Client Gateway é usado pelo serviço TSO Commands do Developer para System z', SCLM Developer Toolkit e opcionalmente CARMA.

As autorizações APF são definidas em SYS1.PARMLIB(PROGxx), se seu site seguiu as recomendações da IBM.

As autorizações APF podem ser configuradas dinamicamente (até o próximo IPL) com os seguintes comandos do console, em que volser é o volume onde o conjunto de dados reside se não for gerenciado pelo SMS:

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

**Notas:**

1. Quando você utiliza a Biblioteca Alternativa para o pacote de produto REXX, o nome da biblioteca de tempo de execução REXX padrão é REXX.\*.SEAGALT, em vez de REXX.\*.SEAGLPA, conforme usado na amostra anterior.
2. Bibliotecas LPA, como REXX.\*.SEAGLPA, são autorizadas automaticamente por APF quando localizadas em LPA e, portanto, não requerem definições explícitas.
3. Alguns dos produtos de correquisito, como a Ferramenta de Depuração IBM, exigem também autorização APF. Consulte os guias de customização do produto relacionados para obter informações adicionais sobre isso.

## Definições de LINKLIST no PROGxx

Definições LINKLIST para o Developer para System z podem ser agrupadas em 3 categorias:

- Bibliotecas de carregamento do Developer para System z necessárias para funções do Developer para System z. Essas definições são descritas nesta seção.
- Bibliotecas de carregamento de requisitos necessárias para funções do Developer para System z. Essas definições são descritas em “Definições de LINKLIST e LPA de Requisito” na página 20.
- Bibliotecas de carregamento do Developer para System z necessárias por outros produtos. Essas definições são descritas em “Definições LINKLIST para Outros Produtos” na página 21.

Para que o serviço SCLM Developer Toolkit (opcional) funcione, todos os módulos BWB\* nas bibliotecas de carregamento FEK.SFEKAUTH e FEK.SFEKLOAD devem ser disponibilizados por meio do STEPLIB ou LINKLIST.

Se você optar por utilizar o STEPLIB, deverá definir as bibliotecas não disponíveis por meio do LINKLIST na diretiva STEPLIB de rsed.envvars, no arquivo de configuração do RSE. Porém, lembre-se de que:

- o uso de STEPLIB no z/OS UNIX provoca um impacto negativo no desempenho.
- Se uma biblioteca STEPLIB for autorizada pelo APF, todas deverão ser autorizadas. As bibliotecas perderão sua autorização do APF se forem combinadas com bibliotecas não-autorizadas pelo STEPLIB.

Os conjuntos de dados LINKLIST são definidos em SYS1.PARMLIB(PROGxx), se seu site seguiu as recomendações da IBM.

As definições necessárias serão semelhantes às seguintes, em que listname é o nome do conjunto LINKLIST que será ativado e volser é o volume no qual o conjunto de dados residirá se não estiver catalogado no catálogo principal:

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

As definições de LINKLIST podem ser criadas dinamicamente (até o próximo IPL) com o seguinte grupo de comandos de console, em que listname é o nome do conjunto LINKLIST atual e volser é o volume no qual o conjunto de dados residirá se não estiver catalogado no catálogo principal:

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

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

## Definições de LINKLIST e LPA de Requisito

Explorador de Sistemas Remotos (RSE) é um processo z/OS UNIX que requer acesso às bibliotecas de carregamento do MVS. As seguintes bibliotecas (pré-requisito) devem ser disponibilizadas, através de STEPLIB ou LINKLIST/LPALIB:

- Biblioteca de carregamento do sistema
  - SYS1.LINKLIB
- Tempo de execução do Ambiente de Linguagem
  - CEE.SCEERUN
  - CEE.SCEERUN2
- Biblioteca de classes DLL do C++
  - CBC.SCLBDLL
- TSO/ISPF Client Gateway do ISPF
  - ISP.SISPLoad
  - ISP.SISPLPA

As seguintes bibliotecas adicionais devem ser disponibilizadas, por meio de STEPLIB ou de LINKLIST/LPALIB, para suportar o uso de serviços opcionais. Essa lista não inclui conjuntos de dados que são específicos para um produto com o qual o Developer para System z interage, como a Ferramenta de Depuração IBM:

- Biblioteca de tempo de execução REXX (para o SCLM Developer Toolkit)
  - REXX.\*.SEAGLPA
- Biblioteca de carregamento do sistema (para criptografia de SSL)
  - SYS1.SIEALNKE
- Biblioteca de carregamento TCP/IP (ao usar APPC para o serviço TSO Commands)
  - TCPIP.SEZALOAD

### Notas:

1. Quando você utiliza a Biblioteca Alternativa para o pacote de produto REXX, o nome da biblioteca de tempo de execução REXX padrão é REXX.\*.SEAGALT, em vez de REXX.\*.SEAGLPA, conforme usado na amostra anterior.
2. As bibliotecas projetadas para colocação de LPA, como REXX.\*.SEAGLPA, podem exigir controle de programa adicional e/ou autorizações do APF se forem acessadas através de LINKLIST ou STEPLIB.
3. Alguns dos produtos de correquisito, como a Ferramenta de Depuração IBM, exigem também definições STEPLIB ou LINKLIST/LPALIB. Consulte os guias de customização do produto relacionados para obter informações adicionais sobre isso.
4. Se CEE.SCEELKED estiver em LINKLIST ou em STEPLIB, TCPIP.SEZALOAD deverá ser colocado antes de CEE.SCEELKED. Se isso não for feito, o sistema 0C1 encerrará de forma anormal para as chamadas do soquete TCP/IP REXX.

Os conjuntos de dados LINKLIST são definidos em SYS1.PARMLIB(PROGxx), se seu site seguiu as recomendações da IBM. Os conjuntos de dados de LPA são definidos em SYS1.PARMLIB(LPALSTxx).



Se você optar por utilizar o STEPLIB, deverá definir as bibliotecas não disponíveis por meio do LINKLIST/LPALIB na diretiva STEPLIB de `rsed.envvars`, no arquivo de configuração do RSE. Porém, lembre-se de que:

- A utilização de STEPLIB no z/OS UNIX tem um impacto de desempenho negativo.
- Se uma biblioteca STEPLIB for autorizada pelo APF, todas deverão ser autorizadas. As bibliotecas perderão sua autorização do APF se forem combinadas com bibliotecas não-autorizadas pelo STEPLIB.
- As bibliotecas incluídas na STEPLIB DD em uma JCL não são propagadas para os processos z/OS UNIX iniciados pela JCL.

## Definições LINKLIST para Outros Produtos

O cliente do Developer para System z possui um componente de geração de códigos chamado Enterprise Service Tools (EST). Para que o código gerado emita mensagens de erro de diagnóstico, todos os módulos IRZ\* e IIRZ\* da biblioteca de carregamento FEK.SFEKLOAD devem ser disponibilizados por meio do STEPLIB ou LINKLIST.

Os conjuntos de dados LINKLIST são definidos em `SYS1.PARMLIB(PROGxx)`, se seu site seguiu as recomendações da IBM.

Se você optar por usar o STEPLIB, deverá definir as bibliotecas não disponíveis por meio do LINKLIST na diretiva STEPLIB da tarefa que executa o código (IMS ou tarefas em lote). Entretanto, lembre-se do seguinte:

- Se uma biblioteca STEPLIB for autorizada pelo APF, todas deverão ser autorizadas. As bibliotecas perderão sua autorização do APF se forem combinadas com bibliotecas não-autorizadas no STEPLIB.

---

## Alterações do PROCLIB

Os procedimentos de construção remota e a tarefa iniciada listados a seguir devem residir em uma biblioteca de procedimentos do sistema definida para seu subsistema JES. Nas instruções a seguir, a biblioteca de procedimentos padrão da IBM, `SYS1.PROCLIB`, é usada.

### JES Job Monitor

Customize o membro de tarefa iniciada de amostra `FEK.#CUST.PROCLIB(JMON)`, conforme descrito no membro, e copie-o para `SYS1.PROCLIB`. Conforme mostrado na amostra de código abaixo, você deve fornecer o seguinte:

- O qualificador de alto nível da biblioteca de carregamento (autorizada), padrão FEK
- O arquivo de configuração do JES Job Monitor, padrão `FEK.#CUST.PARMLIB(FEJJCNG)`

```

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

```

Figura 1. JMON - Tarefa Iniciada do JES Job Monitor

#### Notas:

1. Consulte o Capítulo 8, “Comandos do Operador”, na página 123 para obter informações adicionais sobre os parâmetros de inicialização.
2. A JCL de amostra é enviada inicialmente como FEK.SFEKSAMP(FEJJJCL) e é renomeada para FEK.#CUST.PROCLIB(JMON) em “Configuração da Customização” na página 15.
3. O rastreamento também poderá ser controlado pelos comandos de console, conforme descrito no Capítulo 8, “Comandos do Operador”, na página 123.
4. Esta tarefa deve ser designada para SYSSTC ou ter objetivo equivalente no Workload Manager (WLM).
5. A variável de ambiente LE \_CEE\_ENVFILE\_S requer z/OS 1.8 ou versão mais alta. A variável pode ser substituída por \_CEE\_ENVFILE em níveis mais antigos do z/OS, mas devido a um erro no tempo de execução C, a variável TZ no arquivo de configuração do JES Job Monitor (FEJJCNFG) pode não ser interpretada corretamente.

## Daemon RSE

Customize o membro de tarefa iniciada de amostra FEK.#CUST.PROCLIB(RSED), conforme descrito no membro, e copie-o para o SYS1.PROCLIB. Conforme mostrado na amostra de código abaixo, você deve fornecer o seguinte:

- A porta daemon do RSE, padrão 4035.
- O diretório inicial no qual o Developer para System z está instalado, padrão /usr/lpp/rdz.
- O local dos arquivos de configuração, padrão /etc/rdz

```

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

```

Figura 2. RSED - tarefa iniciada do daemon RSE

**Nota:**

- Consulte o Capítulo 8, “Comandos do Operador”, na página 123 para obter informações adicionais sobre os parâmetros de inicialização.
- A JCL de amostra é enviada inicialmente como FEK.SFEKSAMP(FEKRSED) e é renomeada para FEK.#CUST.PROCLIB(RSED) em “Configuração da Customização” na página 15.
- Limite o comprimento do nome da tarefa a 7 caracteres ou menos. Os comandos de operador **modify** e **stop** falharão com a mensagem "IEE342I MODIFY REJECTED-TASK BUSY" se um nome com 8 caracteres for usado. Esse comportamento é causado pelo design do z/OS UNIX para processos-filho.
- Esta tarefa e os processos-filho que ela cria, devem ser designados para SYSTC ou ter objetivo equivalente no Workload Manager (WLM). Os processos-filho têm o mesmo nome da tarefa-pai, RSED, anexada a um número aleatório de 1 dígito, por exemplo RSED8.

## Daemon de Bloqueio

Customize o membro de tarefa iniciada de amostra FEK.#CUST.PROCLIB(LOCKD), conforme descrito no membro, e copie-o para SYS1.PROCLIB. Conforme mostrado na amostra de código abaixo, você deve fornecer o seguinte:

- O diretório inicial em que o Developer para System z está instalado, padrão /usr/lpp/rdz.
- O local dos arquivos de configuração, padrão /etc/rdz.
- O nível de detalhe de log inicial, padrão 1.

```

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

```

Figura 3. LOCKD - Tarefa iniciada do daemon de bloqueio

**Notas:**

1. Consulte o Capítulo 8, “Comandos do Operador”, na página 123 para obter informações adicionais sobre os parâmetros de inicialização.
2. A JCL de amostra é enviada inicialmente como FEK.SFEKSAMP(FEKLOCKD) e é renomeada para FEK.#CUST.PROCLIB(LOCKD) em “Configuração da Customização” na página 15.
3. Esta tarefa deve ser designada para SYSSTC ou ter objetivo equivalente no Workload Manager (WLM).

## Limitações da JCL para a Variável PARM

O comprimento máximo para a variável PARM é de 100 caracteres, o que poderá causar problemas se você utilizar nomes de diretórios customizados. Para ignorar esse problema, você pode:

- Usar links simbólicos

Os links simbólicos podem ser usados como abreviação de um nome de diretório longo. O comando de amostra z/OS UNIX a seguir define um link simbólico (/usr/lpp/rdz) para outro diretório (/long/directory/name/usr/lpp/rdz).

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

- Usar STDIN

Quando o campo PARM estiver vazio, **BPXBATSL** iniciará um shell z/OS UNIX e executará o script de shell fornecido pelo STDIN. Observe que STDIN deve ser um arquivo z/OS UNIX (alocado como ORDONLY) e que usar STDIN desativa o uso de variáveis PROC para a porta, etc. Observe também que o shell executará os scripts de logon de shell /etc/profile e \$HOME/.profile.

Para utilizar esse método, primeiro você deve atualizar a JCL de inicialização para corresponder a algo como a seguinte amostra:

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

*Figura 4. RSED - inicialização do daemon RSE alternativo*

Segundo, você deve criar o script de shell (/etc/rdz/rsed.stdin.sh neste exemplo), que iniciará o daemon RSE. O conteúdo desse script será semelhante ao seguinte:

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

*Figura 5. rsed.stdin.sh - inicialização do daemon RSE alternativo*

**Nota:** É recomendável alocar rsed.envvars para STDENV na JCL de inicialização do daemon, uma vez que isso define algumas diretivas do z/OS UNIX que ajudam a economizar recursos do sistema ao usar esse método de inicialização.

## Procedimentos de Construção Remota ELAXF\*

O Developer para System z fornece procedimentos JCL de amostra que podem ser usados para a geração de JCL, construções de projetos remotos e recursos de verificação de sintaxe remotos de mapas BMS do CICS, de telas MFS do IMS, de programas COBOL, PL/I, Assembler e C/C++. Esses procedimentos permitem que instalações apliquem seus próprios padrões e garante que os desenvolvedores utilizem os mesmos procedimentos com as mesmas opções e níveis do compilador.

Os procedimentos de amostra e suas funções são listados na Tabela 7.

*Tabela 7. Procedimento ELAXF\* de amostra*

Membro	Propósito
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++.
ELAXFCP1	Procedimento de amostra para compilações COBOL com instruções de pré-processador SCM (-INC e ++INCLUDE).
ELAXFDCL	Procedimento de amostra para executar um programa no modo TSO.
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.
ELAXFPP1	Procedimento de amostra para compilações PL/I com instruções de pré-processador SCM (-INC e ++INCLUDE).
ELAXFTSO	Procedimento de amostra para executar/depurar código do DB2 gerado no modo TSO.
ELAXFUOP	Procedimento de amostra para a geração da etapa UOPT ao construir programas que executam nos subsistemas CICS ou IMS.

Os nomes dos procedimentos e os nomes das etapas nos procedimentos correspondem às propriedades padrão que são fornecidas com o cliente Developer para System z. Se você decidir alterar o nome de um procedimento ou o nome de

uma etapa em um procedimento, o arquivo de propriedades correspondente em todos os clientes também deverá ser atualizado. É recomendável que você não altere os nomes dos procedimentos e das etapas.

Customize os membros do procedimento de construção de amostra, FEK.#CUST.PROCLIB(ELAXF\*), conforme descrito nos membros, e copie-os para SYS1.PROCLIB. Você deve fornecer os qualificadores de alto nível corretos para diferentes bibliotecas de produto, conforme descrito em Tabela 8.

*Tabela 8. Lista de Verificação para Qualificadores de Alto Nível ELAXF\**

Produto	HLQ Padrão	Valor
Developer para System z	FEK	
CICS	CICSTS32.CICS	
DB2	DSN910	
IMS	IMS	
COBOL	IGY.V4R1M0	
PL/I	IBMZ.V3R8M0	
C/C++	CBC	
LE	CEE	
LINKLIB do sistema	SYS1	
MACLIB do sistema	SYS1	

Se os procedimentos ELAXF\* não puderem ser copiados para uma biblioteca de procedimentos de sistema, solicite aos usuários do Developer para System z incluírem uma placa JCLLIB (logo após a placa JOB) nas propriedades da tarefa no cliente.

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

---

## Definições de Segurança

Customize e envie o membro de amostra FEKRACF no conjunto de dados FEK.#CUST.JCL para criar as definições de segurança para o Developer para System z. O usuário que enviar essa tarefa deve ter privilégios de administrador de segurança, como sendo RACF SPECIAL.

### Nota:

- Para esses sites que usam CA ACF2™ para z/OS, consulte o seguinte link, <https://support.ca.com/irj/portal/kbtech?ipLogNrow=0&docid=492389&searchID=TEC492389>, para obter detalhes sobre os comandos de segurança necessários para configurar corretamente o Developer para System z.
- Para aqueles sites que usam CA Top Secret® para z/OS, consulte a sua página do produto no site de suporte do CA (<https://support.ca.com>) e verifique o Documento de Conhecimento relacionado do Developer para System z. Este Documento de Conhecimento possui detalhes dos comandos de segurança necessários para configurar apropriadamente o Developer para System z.

A seguinte lista das definições obrigatórias relacionadas a segurança para Developer para System z são discutidas em detalhe em Capítulo 10,

“Considerações sobre Segurança”, na página 157. Esse capítulo discute também aspectos gerais relacionados à segurança do Developer para System z, incluindo aspectos de segurança de produtos de requisito que não são abordados pela tarefa de amostra FEKRACF.

- Ativar configurações e classes de segurança
- Definir um segmento OMVS para usuários do Developer para System z
- Definir perfis do conjunto de dados
- Definir as tarefas iniciadas JMON, RSED e LOCKD
- Definir a segurança de comando do JES
- Definir RSE como um servidor z/OS UNIX seguro
- Definir bibliotecas controladas pelo programa MVS para RSE
- Definir segurança do aplicativo para RSE
- Definir suporte PassTicket para RSE
- Definir arquivos controlados pelo programa z/OS UNIX para RSE

**Nota:** A tarefa FEKRACF de amostra contém mais que apenas comandos do RACF. A última etapa das definições de segurança consiste em tornar um arquivo do z/OS UNIX controlado por programa. Dependendo das políticas em seu site, essa pode ser uma tarefa para o programador de sistema e não para o administrador de segurança.

<p><b>Atenção:</b> O pedido de conexão com o cliente falhará se a segurança do aplicativo e os PassTickets não estiverem configurados corretamente.</p>
---

---

## FEJJCNFG, Arquivo de Configuração do JES Job Monitor

O JES Job Monitor (JMON) fornece todos os serviços relacionados ao JES. O comportamento do JES Job Monitor pode ser controlado com as definições em FEJJCNFG.

O FEJJCNFG está localizado em FEK.#CUST.PARMLIB, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes.

Customize o membro de configuração do JES Job Monitor de amostra FEJJCNFG, conforme mostrado na seguinte amostra. Linhas de comentários são iniciadas com um sinal de sustenido (#) ao utilizar uma página de códigos US. As linhas de dados só podem ter uma diretiva e seu valor designado; não são permitidos comentários na mesma linha.

**Nota:** A tarefa iniciada JMON deve ser reiniciada para assimilar as mudanças feitas.

```

SERV_PORT=6715
TZ=EST5EDT
#_BPXK_SETIBMOPT_TRANSPORT=TCPIP
#APPLID=FEKAPPL
#AUTHMETHOD=SAF
#CODEPAGE=UTF-8
#CONCHAR=$
#CONSOLE_NAME=JMON
#GEN_CONSOLE_NAME=OFF
#HOST_CODEPAGE=IBM-1047
#LIMIT_COMMANDS=NOLIMIT
#LIMIT_VIEW=USERID
#LISTEN_QUEUE_LENGTH=5
#MAX_DATASETS=32
#MAX_THREADS=200
#TIMEOUT=3600
#TIMEOUT_INTERVAL=1200
#SUBMITMETHOD=TSO
#TSO_TEMPLATE=FEK.#CUST.CNTL(FEJTS0)

```

Figura 6. FEJCNFG, arquivo de configuração do JES Job Monitor

## 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, ambos servidor e os clientes Developer para System z devem ser configurados com o mesmo número de porta. Se você alterar o número da porta do servidor, todos os clientes também deverão 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**.
- Ao utilizar um cliente da versão 7.1 ou posterior, toda a comunicação nesta porta é confinada para a sua máquina host z/OS.

**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 Estados Unidos). Altere isso para representar seu fuso horário. Informações adicionais podem ser localizadas no *UNIX System Services Command Reference* (SA22-7802).

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

## \_BPXK\_SETIBMOPT\_TRANSPORT

Especifica o nome da pilha TCPIP a ser utilizada. O padrão é TCPIP. Remova o comentário e altere para o nome de pilha TCPIP solicitado, conforme definido na instrução TCPIPJOBNAME no TCPIP.DATA relacionado.

### Nota:

- Codificar uma instrução SYSTCPD DD no servidor JCL não configura a afinidade de pilha solicitada.
- Quando esta diretiva não está ativa, o JES Job Monitor se liga a cada pilha disponível no sistema (BIND INADDRANY).

## APLICADO

Especifica o identificador do aplicativo usado para identificar o JES Job



Monitor para seu software de segurança. O padrão é FEKAPPL. Remova o comentário e altere para o ID do aplicativo desejado.

**Nota:** Esse valor deve corresponder ao ID do aplicativo configurado para o RSE no arquivo de configuração `rsed.envvars`. Se esses valores forem diferentes, o RSE não pode conectar o cliente ao JES Job Monitor.

#### **AUTHMETHOD**

O padrão é SAF, o que significa que a interface de segurança System Authorization Facility (SAF) é usada. Não faça alterações, a menos que você seja orientado pelo centro de suporte da IBM.

#### **CODEPAGE**

A página de códigos da estação de trabalho. O padrão é UTF-8. A página de códigos da estação de trabalho é configurada como UTF-8 e geralmente não deve ser alterada. Talvez seja necessário remover o comentário da diretiva e alterar 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.

#### **CONCHAR**

Especifica o caractere de comando do console do JES. CONCHAR tem CONCHAR=\$ como padrão para JES2, ou CONCHAR=\* para JES3. Remova o comentário e altere para o caractere do comando solicitado.

#### **CONSOLE\_NAME**

Especifica o nome do console do EMCS usado para emitir comandos com relação a tarefas (Suspender, Liberar, Cancelar e Limpar). O padrão é JMON. Remova o comentário e altere para o nome de console desejado, utilizando as diretrizes a seguir.

- CONSOLE\_NAME deve ser um nome de console que consiste em 2 a 8 caracteres alfanuméricos ou em '&SYSUID' (sem as aspas).
- Se um nome de console for especificado, um único console com esse nome é usado para todos os usuários. Se o console com esse nome estiver em uso, o comando emitido pelo cliente falhará.
- Se &SYSUID for especificado, o ID de usuário cliente será usado como o nome do console. Assim, um console diferente é usado para cada usuário. Se o console com esse nome estiver em uso (por exemplo, o usuário estiver usando o SDSF ULOG), o comando emitido pelo cliente pode falhar, dependendo da configuração de GEN\_CONSOLE\_NAME.

Independentemente do nome de console usado, o ID de usuário cliente que solicita o comando é usado como a LU do console, deixando um rastro nas mensagens IEA630I e IEA631I do syslog.

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

#### **GEN\_CONSOLE\_NAME**

Ativa ou desativa a geração automática de nomes de consoles alternativos. O padrão é OFF. Remova o comentário e altere para ON para ativar nomes de console alternativos.

Esta diretiva somente é usada quando CONSOLE\_NAME é igual a &SYSUID e o ID do usuário não está disponível no console.

Se GEN\_CONSOLE\_NAME=ON, um nome de console alternativo é gerado, anexando um único dígito numérico ao ID do usuário. Os dígitos 0 a 9 são tentados. Se nenhum console disponível for localizado, o comando emitido pelo cliente falha.

Se GEN\_CONSOLE\_NAME=OFF, o comando emitido pelo cliente falha.

**Nota:** As únicas configurações válidas são ON e OFF.

## HOST\_CODEPAGE

A página de códigos do host. O padrão é IBM-1047. Remova o comentário e altere para estabelecer correspondência com a página de códigos do host.

A partir da versão 7.6.1, os clientes do Developer para System z ignoram o valor do HOST\_CODEPAGE especificado aqui e usam a página de códigos especificada localmente nas propriedades do subsistema “MVS Files”.

**Nota:** Até mesmo para clientes recentes, o JES Job Monitor usará a página de códigos do host especificada no HOST\_CODEPAGE durante a configuração inicial de comunicação do cliente inicial.

## LIMIT\_COMMANDS

Define em que tarefas o usuário pode emitir comandos JES selecionados (Mostrar JCL, Suspend, Liberar, Cancelar e Limpar). O padrão (LIMIT\_COMMANDS=USERID) limita os comandos às tarefas que pertencem ao usuário. Remova o comentário dessa diretiva e especifique LIMITED ou NOLIMIT para permitir que o usuário emita comandos em relação a todos os arquivos em spool, se for permitido pelo produto de segurança.

Tabela 9. Matriz de Permissão do Comando LIMIT\_COMMANDS

LIMIT_COMMANDS	Proprietário da tarefa	
	Usuário	Outros
USERID (padrão)	Permitido	Não permitido
LIMITED	Permitido	Permitido somente se for permitido explicitamente por perfis de segurança
NOLIMIT	Permitido	Permitido se for permitido pelos perfis de segurança ou quando a classe JESSPOOL não estiver ativa

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

## LIMIT\_VIEW

Define qual saída o usuário pode visualizar. O padrão (LIMIT\_VIEW=NOLIMIT) permite que o usuário visualize todas as saídas do JES, se for permitido pelo produto de segurança. Remova o comentário dessa diretiva e especifique USERID para limitar a visualização à saída que pertence ao usuário.

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

## LISTEN\_QUEUE\_LENGTH

O comprimento da fila de atendimento do TCP/IP. O padrão é 5. Não faça alterações, a menos que você seja orientado pelo centro de suporte da IBM.

## MAX\_DATASETS

O número máximo de conjuntos de dados de saída em spool que o JES Job

Monitor retornará para o cliente (por exemplo, SYSOUT, SYSPRINT, SYS00001 e assim por diante). O padrão é 32. O valor máximo é 2147483647.

#### **MAX\_THREADS**

Número máximo de usuários que podem utilizar um Monitor de Tarefas do JES por vez. O padrão é 200. O valor máximo é 2147483647. Aumentar este número pode exigir o aumento do tamanho do espaço de endereços do JES Job Monitor.

#### **TIMEOUT**

O período de tempo, em segundos, antes que um encadeamento seja eliminado devido à falta de interação com o cliente. O padrão é 3600 (1 hora). O valor máximo é 2147483647. TIMEOUT=0 desativa a função.

#### **TIMEOUT\_INTERVAL**

O número de segundos entre as verificações de tempo limite. O padrão é 1200. O valor máximo é 2147483647.

#### **SUBMITMETHOD=TSO**

Enviar tarefas por meio do TSO. O padrão (SUBMITMETHOD=JES) envia jobs diretamente para o JES. Remova o comentário dessa diretiva e especifique TSO para enviar a tarefa através do comando TSO **SUBMIT**. Esse método permite que saídas do TSO sejam invocadas, porém, ele prejudica o desempenho e, por essa razão, não é recomendado.

##### **Nota:**

- As únicas configurações válidas são TSO e JES.
- Se SUBMITMETHOD=TSO for especificado, então TSO\_TEMPLATE também deve ser definido.

#### **TSO\_TEMPLATE**

JCL do wrapper para envio de tarefa através de TSO. O valor padrão é FEK.#CUST.CNTL(FEJTSO). Esta instrução faz referência ao nome completo do membro da JCL a ser usado como um wrapper para o envio TSO. Consulte a instrução SUBMITMETHOD para obter informações adicionais.

##### **Nota:**

- Uma tarefa do wrapper de amostra é fornecida em FEK.#CUST.CNTL(FEJTSO). Consulte este membro para obter informações adicionais sobre a customização necessária.
- TSO\_TEMPLATE não possui nenhum efeito a menos que SUBMITMETHOD=TSO também seja especificado.

---

## **Arquivo de Configuração do RSE rsed.envvars**

Os processos de daemon de bloqueio RSE e de servidor RSE (daemon RSE, conjunto de encadeamentos do RSE e servidor RSE) usam as definições em rsed.envvars. Serviços opcionais do Developer para System z e de terceiros também podem utilizar esse arquivo de configuração para definir variáveis de ambiente para seu uso.

O RSE (Explorador de Sistemas Remotos) fornece os serviços principais, como conectar o cliente ao host e iniciar outros servidores para serviços específicos. O daemon de bloqueio fornece serviços de rastreamento para bloqueios de conjuntos de dados.

O `rsed.envvars` está localizado em `/etc/rdz/`, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa `FEK.SFEKSAMP(FEKSETUP)`. Consulte “Configuração da Customização” na página 15 para obter mais detalhes. É possível editar o arquivo com o comando do TSO **OEDIT**.

Consulte o seguinte arquivo de amostra `rsed.envvars`, que deve ser customizado para corresponder ao ambiente do sistema. Linhas de comentários são iniciadas com um sinal de sustenido (#) ao utilizar uma página de códigos US. As linhas de dados só podem ter uma diretiva e seu valor designado; não são permitidos comentários na mesma linha. As continuações de linha e os espaços ao redor do sinal de igual (=) não são suportadas.

**Nota:** As tarefas iniciadas **RSED** e **LOCKD** devem ser reiniciadas para assimilar as mudanças feitas.



```

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

```

Figura 7. *rse.envvars* - Arquivo de configuração do RSE

```

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

```

Figura 8. (continuação)

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

As seguintes definições são requeridas:

**JAVA\_HOME**

Diretório inicial Java. O padrão é `/usr/lpp/java/J5.0`. Altere para corresponder com a sua instalação do Java.

**RSE\_HOME**

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

**\_RSE\_LOCKD\_PORT**

Número da porta do daemon de bloqueio RSE. O padrão é 4036. Pode ser alterado, se desejado.

**Nota:**

- Antes de selecionar uma porta, verifique se ela está disponível em seu sistema com os comandos do TSO **NETSTAT** e **NETSTAT PORTL**.
- Toda a comunicação nesta porta está confinada à máquina host do z/OS.

**\_RSE\_HOST\_CODEPAGE**

A página de códigos do host. O padrão é IBM-1047. Altere para corresponder à página de códigos do host.

**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 Estados Unidos). Altere para corresponder ao fuso horário.

Informações adicionais podem ser localizadas no *UNIX System Services Command Reference* (SA22-7802).

**LANG**

Especifica o nome do código do idioma padrão. O padrão é `C`. O `C` especifica o código do idioma POSIX e (por exemplo) `Ja_JP` especifica o código do idioma japonês. Altere para corresponder ao código do idioma.

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

**\_CEE\_DMPTARG**

Local de dump do Language Environment (LE) do z/OS UNIX usado pela Java Virtual Machine (JVM). O padrão é `/tmp`.

**STEPLIB**

Acesse conjuntos de dados MVS que não estão no LINKLIST/LPALIB. O padrão é `NONE`.

Você pode ignorar a necessidade de ter bibliotecas (pré-requisito) no LINKLIST/LPALIB removendo o comentário e customizando uma ou mais das seguintes diretivas STEPLIB. Consulte o “Alterações PARMLIB” na página 16 para obter informações adicionais sobre o uso das bibliotecas listadas abaixo:

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

**Nota:**



- A utilização de STEPLIB no z/OS UNIX tem um impacto de desempenho negativo.
- Se uma biblioteca STEPLIB for autorizada pelo APF, todas deverão ser autorizadas. As bibliotecas perderão sua autorização do APF se forem combinadas com bibliotecas não-autorizadas pelo STEPLIB.
- As bibliotecas que são designadas para a colocação de LPA podem exigir controle de programa adicional e autorizações de APF se elas forem acessadas por meio de LINKLIST ou STEPLIB.
- Codificar uma instrução STEPLIB DD no servidor JCL não configura a concatenação de STEPLIB necessária.

#### **RSE\_SAF\_CLASS**

Especifica a interface Java para seu produto de segurança. O padrão é `/usr/include/java_classes/IRRRacf.jar`. Altere para corresponder à configuração do software de segurança.

**Nota:** A partir do z/OS 1.10, o `/usr/include/java_classes/IRRRacf.jar` que faz parte do SAF, fornecido com o z/OS de base, também estará disponível para clientes não-RACF.

#### **RSE\_JVAOPTS**

Opções Java adicionais específicas do RSE. . Consulte o “Definindo Parâmetros Extras de Inicialização Java com `_RSE_JVAOPTS`” na página 42 para obter informações adicionais sobre esta definição.

O Developer para System z utiliza o TSO/ISPF Client Gateway do ISPF por padrão para o serviço TSO Commands. Uma transação APPC é usada quando a seguinte opção `_RSE_JVAOPTS` está sem comentários:

```
RSE_JVAOPTS="$ _RSE_JVAOPTS -DTSO_SERVER=APPC"
```

As definições a seguir são necessárias se o TSO/ISPF Client Gateway do ISPF for usado para o serviço TSO Commands, SCLM Developer Toolkit ou CARMA.

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

Diretório inicial para o código do ISPF que fornece o serviço TSO/ISPF Client Gateway. O padrão é `/usr/lpp/ispf`. Altere para corresponder à instalação do ISPF. Essa diretiva será necessária apenas se o TSO/ISPF Client Gateway do ISPF for usado.

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

Diretório de configuração base do ISPF. O padrão é `/etc/rdz`. Altere para corresponder ao local do ISPF.conf, o arquivo de customização do Gateway do Cliente TSO/ISPF. Essa diretiva será necessária apenas se o TSO/ISPF Client Gateway do ISPF for usado.

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

Diretório de trabalho base do ISPF. O padrão é `/var/rdz`. Altere para corresponder ao local do diretório WORKAREA usado pelo TSO/ISPF Client Gateway. Essa diretiva será necessária apenas se o TSO/ISPF Client Gateway do ISPF for usado.

Notas:

- O TSO/ISPF Client Gateway incluirá `/WORKAREA` ao caminho especificado em `_CMDSERV_WORK_HOME`. Não o inclua você próprio.
- Se você não usar a tarefa de amostra SFEKSAMP(FEKSETUP) para construir o ambiente customizável, então deverá verificar se o diretório

WORKAREA existe no caminho especificado em `_CMDSEV_WORK_HOME`. Os bits de permissão do diretório devem ser 777.

#### **STEPLIB**

STEPLIB está descrito anteriormente na seção de definições necessárias.

#### **RSE\_CMDSEV\_OPTS**

Opções Java adicionais específicas do TSO/ISPF Client Gateway. O padrão é `""`. Consulte o “Definindo Parâmetros Extras de Inicialização Java com `_RSE_CMDSEV_OPTS`” na página 46 para obter informações adicionais sobre esta definição. Essa diretiva será necessária apenas se o TSO/ISPF Client Gateway do ISPF for usado.

As definições a seguir são necessárias se o SCLM Developer Toolkit for utilizado.

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

Diretório de configuração base do SCLM Developer Toolkit. O padrão é `/var/rdz/scldmt`. Altere para corresponder ao local do diretório CONFIG usado pelo SCLMDT para armazenar informações do projeto SCLM. Essa diretiva é necessária apenas quando o SCLMDT for usado.

**Nota:** SCLMDT incluirá `/CONFIG` e `/CONFIG/PROJECT` ao caminho especificado em `SCLMDT_CNFG_HOME`. Não o inclua você próprio.

#### **STEPLIB**

STEPLIB está descrito anteriormente na seção de definições necessárias.

#### **\_SCLMDT\_TRANTABLE**

Nome do VSAM de tradução de nomes completos/abreviados. O padrão é `FEK.#CUST.LSTRANS.FILE`. Remova o comentário e altere para corresponder ao nome usado na tarefa de amostra do SCLM ISP.SISPSAMP(FLM02LST). Essa diretiva será necessária apenas se a tradução do nome completo/abreviado no SCLM Developer Toolkit for usada.

#### **ANT\_HOME**

Diretório inicial da instalação Ant. O padrão é `/usr/lpp/Apache/Ant/apache-ant-1.7.1`. Altere para corresponder à instalação Ant. Essa diretiva será necessária apenas se o suporte para construção JAVA/J2EE for utilizado com o SCLM Developer Toolkit.

As definições a seguir são opcionais. Se forem omitidas, os valores padrão serão usados:

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

Especifica o intervalo de portas que o servidor RSE pode abrir para comunicação com um cliente. Qualquer porta pode ser usada por padrão. Consulte o “Definindo o PORTRANGE Disponível para o Servidor RSE” na página 41 para obter informações adicionais sobre esta definição. Essa é uma diretiva opcional.

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

Especifica o nome da pilha TCP/IP a ser usada. O padrão é `TCPIP`. Remova o comentário e altere para o nome da pilha TCP/IP solicitado, conforme definido na instrução `TCPIPJOBNAME` no `TCPIP.DATA` relacionado. Essa é uma diretiva opcional.

#### **Nota:**

- Codificar uma instrução SYSTCPD DD no servidor JCL não configura a afinidade de pilha solicitada.

- Quando esta diretiva não está ativa, o RSE se liga a cada pilha disponível no sistema (BIND INADDRANY).

#### **\_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. Essa é uma diretiva opcional.

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

Force o servidor RSE a utilizar esta LU de parceiro APPC. O padrão é a LU de base especificada durante a configuração de APPC. Esta é uma diretiva opcional.

#### **GSK\_CRL\_SECURITY\_LEVEL**

Especifica o nível de segurança que os aplicativos SSL usarão ao contatar servidores LDAP para verificar CRLs para certificados revogados durante a validação de certificado. O padrão é MEDIUM. Remova o comentário e altere para impingir o uso do valor especificado. Essa é uma diretiva opcional. Os valores a seguir são válidos:

- LOW - A validação do certificado não falhará se o servidor LDAP não puder ser contatado.
- MEDIUM - A validação do certificado requer que o servidor LDAP possa ser contatado, mas não requer que uma CRL seja definida. Esse é o padrão
- HIGH - A validação do certificado requer que o servidor LDAP possa ser contatado e que uma CRL seja definida.

**Nota:** Esta diretiva exige z/OS 1.9 ou superior.

#### **GSK\_LDAP\_SERVER**

Especifica um ou mais nomes de hosts de servidores LDAP separados por espaços em branco. Remova o comentário e altere para impingir o uso dos servidores LDAP especificados para obter sua CRL. Essa é uma diretiva opcional.

O nome do host pode ser um endereço TCP/IP ou uma URL. Cada nome de host pode conter um número de porta opcional separado do nome do host por dois pontos (:).

#### **GSK\_LDAP\_PORT**

Especifica a porta do servidor LDAP. O padrão é 389. Remova o comentário e altere para impingir o uso do valor especificado. Essa é uma diretiva opcional.

#### **GSK\_LDAP\_USER**

Especifica o nome distinto a usar ao conectar ao servidor LDAP. Remova o comentário e altere para impingir o uso do valor especificado. Essa é uma diretiva opcional.

#### **GSK\_LDAP\_PASSWORD**

Especifica a senha a usar ao conectar ao servidor LDAP. Remova o comentário e altere para impingir o uso do valor especificado. Essa é uma diretiva opcional.

As definições a seguir são necessárias e não devem ser alteradas, a menos que você seja orientado pelo centro de suporte da IBM:

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

Opções de tempo de execução do LE (Language Environment). O padrão é "ALL31(ON) HEAP(32M,32K,ANYWHERE,KEEP,,) TRAP(ON)". 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.

#### **JAVA\_PROPAGATE**

Propaga o contexto de segurança e de carga de trabalho durante a criação do encadeamento (apenas Java versão 1.4 e mais antiga). O padrão é NO. Não modifique.

#### **RSE\_LIB**

Caminho da biblioteca RSE. O padrão é \$RSE\_HOME/lib. Não modifique.

**PATH** Caminho de comandos. O padrão é .:\$JAVA\_HOME/bin:\$RSE\_HOME/bin:\$\_CMDSERV\_BASE\_HOME/bin:\$PATH. Não modifique.

#### **LIBPATH**

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

#### **CLASSPATH**

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

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

Opções Java adicionais específicas do serviço Comandos do TSO. O padrão é "&SESSION=SPAWN\$\_RSE\_CMDSERV\_OPTS". Não modifique.

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

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

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

Classe Java do servidor RSE. O padrão é org.eclipse.dstore.core.server.Server. Não modifique.

#### **\_RSE\_DAEMON\_CLASS**

Classe Java para o daemon RSE. O padrão é com.ibm.etoools.zos.server.RseDaemon. Não modifique.

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

Classe Java para o conjunto de encadeamentos do RSE. O padrão é com.ibm.etoools.zos.server.ThreadPoolProcess. Não modifique.

#### **\_RSE\_LOCKD\_CLASS**

Classe Java para o daemon de bloqueio RSE. O padrão é com.ibm.ftt.rse.mvs.server.miners.MVSLockDaemon. 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.

#### **SCLMDT\_BASE\_HOME**

Diretório inicial para o código do SCLM Developer Toolkit. O padrão é \$RSE\_HOME. Não modifique.

## SCLMDT\_WORK\_HOME

Diretório de trabalho base do SCLM Developer Toolkit. O padrão é `$_CMDSERV_WORK_HOME`. Não modifique.

## CGI\_DTWORK

Suporte do SCLM Developer Toolkit para clientes antigos. O padrão é `$_SCLMDT_WORK_HOME`. Não modifique.

## Definindo o PORTRANGE Disponível para o Servidor 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. Esse intervalo de portas não possui conexão com a porta do daemon RSE.

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 do host 4035, daemon do RSE.
2. O daemon do RSE cria um encadeamento do servidor RSE.
3. O servidor 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 daemon RSE retorna o número da porta para o cliente.
5. O cliente se conecta à porta do host.

### Nota:

- O processo é semelhante para o método de conexão alternativo (opcional) utilizando REXEC/SSH.
- Consulte o Capítulo 11, “Entendimento do Developer para System z”, na página 189 para obter informações adicionais.

Para especificar o intervalo de portas para o cliente se comunicar com o z/OS, remova o comentário e customize 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**.

O formato do PORTRANGE é: `_RSE_PORTRANGE=min-max` (max é não-inclusivo; por exemplo, `_RSE_PORTRANGE=8108-8118` significa que números de porta de 8108 a 8117 são utilizáveis). O número da porta usado 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 de porta especificado será usado. Se a porta não estiver disponível, a conexão falhará. Esta configuração não é recomendada.

**Nota:** O host pode negar este tipo de pedido de conexão ao especificar a diretiva `deny.nonzero.port=true` em `rsed.envvars`. Consulte o “Definindo Parâmetros Extras de Inicialização Java com `_RSE_JAVAOPTS`” na página 42 para obter informações adicionais sobre esta diretiva.

2. Se o 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 usado. Se nenhuma porta no intervalo estiver disponível, a conexão falhará.
3. Se o número de porta nas propriedades do subsistema for 0, e `_RSE_PORTRANGE` não for especificado no `rsed.envvars`, nenhuma porta disponível será usada.

**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.

## Definindo Parâmetros Extras de Inicialização Java com `_RSE_JAVAOPTS`

Com as diferentes diretivas `_RSE_*OPTS`, `rsed.envvars` oferece a possibilidade de fornecer parâmetros extras para Java ao iniciar os processos RSE. As opções de amostra incluídas em `rsed.envvars` podem ser ativadas pela remoção dos comentários.

`_RSE_JAVAOPTS` define opções Java padrão e específicas do RSE.

`_RSE_JAVAOPTS=""`

Inicialização variável. Não modifique.

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

Defina o tamanho de heap inicial (Xms) e máximo (Xmx). Os padrões são 1M e 256M, respectivamente. Altere para aplicar os valores de tamanho de heap desejados. Se esta diretiva for comentada a linha, os valores padrão Java, que são 4M e 512M respectivamente, (1M e 64M para Java 5.0) serão usados.

**Nota:** Consulte o “Definições de Recursos Principais” na página 231 para determinar os valores ideais para essa diretiva.

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

Diretório que contém o daemon RSE, a criação de log de servidor e os dados de auditoria do RSE. O padrão é `/var/rdz/logs`. Altere para aplicar o local desejado. Se essa diretiva for comentada, o diretório inicial do ID de usuário designado ao daemon RSE será usado. O diretório inicial é definido no segmento de segurança OMVS do ID do usuário.

**Nota:** Se esta diretiva (ou sua contraparte, o diretório inicial) não especificar um caminho absoluto (o caminho não inicia com uma barra (/)), o local real do log é relativo ao diretório de configuração (pelo padrão `/etc/rdz`).

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

Diretório que conduz aos logs específicos do usuário. O padrão é `/var/rdz/logs`. Altere para aplicar o local desejado. Se essa diretiva for comentada, o diretório inicial do ID do usuário cliente será usado. O diretório inicial é definido no segmento de segurança OMVS do ID do usuário.

**Nota:**

- Se esta diretiva (ou sua contraparte, o diretório inicial) não especificar um caminho absoluto (o caminho não inicia com uma

barra (/)), o local real do log é relativo ao diretório de configuração (pelo padrão /etc/rdz).

- O caminho completo para os logs do usuário é userlog/dstorelog/\$LOGNAME/, em que userlog é o valor da diretiva user.log, dstorelog é o valor da diretiva DSTORE\_LOG\_DIRECTORY e \$LOGNAME é o ID do usuário do cliente, em maiúsculas.
- Assegure-se de que os bits de permissão para userlog/dstorelog estejam configurados de modo que cada cliente possa criar \$LOGNAME.

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

Esse diretório está anexado ao caminho especificado na diretiva user.log. Juntos, eles criam o caminho que conduz aos logs específicos do usuário. O padrão é null-string. Altere para impingir o uso do diretório especificado. Se essa diretiva for comentada, .eclipse/RSE/ será usado.

**Nota:**

- O caminho completo para os logs do usuário é userlog/dstorelog/\$LOGNAME/, em que userlog é o valor da diretiva user.log, dstorelog é o valor da diretiva DSTORE\_LOG\_DIRECTORY e \$LOGNAME é o ID do usuário do cliente, em maiúsculas.
- O diretório especificado aqui é relativo ao diretório especificado em user.log e, dessa forma, pode não começar com uma barra (/).
- Assegure-se de que os bits de permissão para userlog/dstorelog estejam configurados de modo que cada cliente possa criar \$LOGNAME.

As diretivas a seguir são comentadas por padrão.

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

Quantidade máxima de clientes atendidos por um conjunto de encadeamentos. O padrão é 60. Remova o comentário e customize para limitar o número de clientes por conjunto de encadeamentos. Observe que outros limites podem impedir que o RSE atinja esse limite.

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

Valor máximo dos encadeamentos ativos em um conjunto de encadeamentos para permitir clientes novos. O padrão é 1000. Remova o comentário e customize para limitar o número de clientes por conjunto de encadeamentos baseado no número de encadeamentos em uso. Observe que cada conexão do cliente usa vários encadeamentos (16 ou mais) e que outros limites podem impedir que o RSE atinja esse limite.

**Nota:** Esse valor deve ser inferior ao configurado para MAXTHREADS e MAXTHREADTASKS, em SYS1.PARMLIB(BPXPRMxx).

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

O número mínimo de conjuntos de encadeamentos ativos. O padrão é 1. Remova o comentário e customize para iniciar pelo menos o número listado de processos do conjunto de encadeamentos. Os processos do conjunto de encadeamentos são usados para balanceamento de carga dos encadeamentos do servidor RSE. Mais novos processos serão iniciados



quando forem necessários. Iniciar os novos processos diretamente ajuda a evitar atrasos na conexão, mas usa mais recursos durante os tempos inativos.

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

O número máximo de conjuntos de encadeamentos ativos. O padrão é 100. Remova o comentário e customize para limitar o número de processos do conjunto de encadeamentos. Os processos do conjunto de encadeamentos são usados para balanceamento de carga dos encadeamentos do servidor RSE e a limitação deles limitará a quantidade de conexões do cliente ativas.

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

Versão TCP/IP. O padrão é false, significando que uma interface IPv4 será usada. Remova o comentário e especifique true para utilizar uma interface IPv6.

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

Mantenha uma cópia dos arquivos de log do host pertencentes à sessão anterior. O padrão é false. Remova o comentário e especifique true para renomear os arquivos de log anteriores para \*.last durante a inicialização do servidor e a conexão do cliente.

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

Grave os fluxos stdout e stderr dos conjuntos de encadeamento em um arquivo de log. O padrão é false. Remova o comentário e especifique true para salvar os fluxos stdout e stderr. Os arquivos de log resultantes estão localizados no diretório mencionado pela diretiva daemon.log.

**Nota:**

- O comando do operador **MODIFY RSESTANDARDLOG** pode ser usado para parar e iniciar dinamicamente a atualização dos arquivos de log de fluxo.
- Não haverá arquivos de log stdout.log e stderr.log específicos do usuário quando a diretiva enable.standard.log estiver ativa. Os dados específicos do usuário são gravados agora no fluxo do conjunto de encadeamento do RSE correspondente.

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

Opção de verificação de Port Of Entry (POE). O padrão é false. Remova o comentário e especifique true para aplicar a verificação de POE para conexões do cliente. Durante a verificação de POE, o endereço IP do cliente é mapeado para uma zona de segurança de rede por seu software de segurança. O ID de usuário cliente deve ter permissão para utilizar o perfil que define a zona de segurança.

**Nota:**

- A verificação de POE também deve ser ativada no produto de segurança.
- A ativação da verificação de POE também ficará disponível para outros serviços z/OS UNIX, como INETD.

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

Use seu software de segurança para autenticar um logon com um certificado X.509. O padrão é true. Remova o comentário e especifique false para que o daemon RSE faça a autenticação sem depender do suporte do X.509 de seu software de segurança.



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

| Suporte diretórios iniciais criados pela montagem automática do z/OS  
| UNIX. O padrão é false. Remova o comentário e especifique true para  
| assegurar que a montagem automática do z/OS UNIX usa o ID de usuário  
| cliente como proprietário do diretório.

| **Nota:** A montagem automática do z/OS UNIX usa o ID do usuário do  
| processo que invocou o serviço ao criar um sistema de arquivo. Se  
| esta opção estiver desativada, este processo será o servidor do  
| conjunto de encadeamentos RSE (STCRSE do ID do usuário). Se esta  
| opção estiver ativada, um novo processo temporário será criado  
| usando o ID de usuário cliente antes de invocar o serviço.

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

| Opção de auditoria. O padrão é false. Remova o comentário e especifique  
| true para aplicar o log de auditoria de ações feitas pelos clientes. Os logs  
| de auditoria são gravados no local do log do daemon RSE. Consulte a  
| opção daemon.log da variável \_RSE\_JAVAOPTS para saber onde ela está.

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

| Número de dias armazenado no arquivo de log de auditoria 1. O padrão é  
| 30. Remova o comentário e customize para controlar quantos dados de  
| auditoria serão gravados no arquivo de log de auditoria 1. O valor máximo  
| é 365.

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

| Número de dias durante os quais os logs de auditoria são mantidos. O  
| padrão é 0 (sem limite). Remova o comentário e customize para excluir  
| logs de auditoria após um determinado número de dias. O valor máximo é  
| 365.

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

| Desaprove o cliente escolher um número de porta de comunicação. O  
| padrão é false. Remova o comentário e especifique true para recusar  
| conexões em que o cliente especifica qual porta do host deve ser usada  
| pelo servidor RSE para a conexão. Consulte o "Definindo o PORTRANGE  
| Disponível para o Servidor RSE" na página 41 para obter informações  
| adicionais.

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

| Desaprove um ID do usuário efetuar logon em múltiplos horários. O  
| padrão é true. Remova o comentário e especifique false para permitir que  
| um usuário efetue logon múltiplas vezes em um único daemon RSE.

| **Nota:** Uma segunda tentativa de efetuar logon fará com que a primeira  
| seja cancelada pelo host, se esta diretiva não estiver ativa ou  
| configurada como false. Este cancelamento será acompanhado pela  
| mensagem do console FEK210I.

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

| Remova automaticamente os conjuntos de encadeamentos RSE que estão  
| em um estado de erro irreversível. Pelo padrão, conjuntos de  
| encadeamentos RSE com erro não são removidos automaticamente.  
| Remova o comentário e customize para remover automaticamente  
| servidores dos conjuntos de encadeamentos RSE errados a cada intervalo  
| (unidade de intervalos em segundos). Especificar 0 desativa a função.

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

ID do aplicativo do servidor RSE. O padrão é FEKAPPL. Remova o comentário e customize essa opção para impingir o uso do ID do aplicativo desejado.

**Nota:**

- O ID do aplicativo deve ser definido para seu software de segurança. Se isso não for feito, o cliente não poderá efetuar logon.
- Consulte "Usando os PassTickets" na página 162 para obter informações sobre implicações de segurança ao alterar este valor.
- O ID do aplicativo deve corresponder ao ID do aplicativo usado pelo JES Job Monitor. Consulte "FEJJCNFG, Arquivo de Configuração do JES Job Monitor" na página 27 para aprender como definir o ID do aplicativo para JES Job Monitor.

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

Opção de salvamento de senha. O padrão é false. Remova o comentário e especifique true para evitar que os usuários salvem suas senhas de host no cliente. As senhas salvas anteriormente serão removidas. Essa opção funciona apenas com clientes da versão 7.1 e superior.

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -DHIDE\_ZOS\_UNIX=true"**

Oculte a opção z/OS UNIX. O padrão é false. Remova o comentário e especifique true para evitar que os usuários vejam os elementos z/OS UNIX (estrutura de diretórios e linha de comandos) no cliente. Essa opção funciona apenas com clientes da versão 7.6 e superior.

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS  
-DDSTORE\_IDLE\_SHUTDOWN\_TIMEOUT=3600000"**

Desconecte clientes inativos. Por padrão, os clientes inativos não são desconectados. Remova o comentário e customize para desconectar os clientes que estão inativos pelo total listado de milissegundos (3600000 igual a 1 hora).

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

Inicie o rastreamento do dstore. Utilize apenas quando orientado pelo IBM Support Center. Observe que o arquivo de log .dstoreTrace resultante é criado em Unicode (ASCII), e não em EBCDIC.

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

Inicie o rastreamento de memória do dstore. Utilize apenas quando orientado pelo IBM Support Center. Observe que o arquivo de log resultante .dstoreMemLogging é criado em Unicode (ASCII), e não em EBCDIC.

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

Use uma transação APPC para o serviço TSO Commands. Por padrão, o TSO/ISPF Client Gateway do ISPF é usado. Remova o comentário para utilizar uma transação APPC. Não altere o valor designado.

## **Definindo Parâmetros Extras de Inicialização Java com \_RSE\_CMDSERV\_OPTS**

Com as diferentes diretivas `_RSE_*OPTS`, `rsed.envvars` oferece a possibilidade de fornecer parâmetros extras para Java ao iniciar os processos RSE. As opções de amostra incluídas em `rsed.envvars` podem ser ativadas pela remoção dos comentários.

As diretivas `_RSE_CMDSERV_OPTS` são opções Java específicas do RSE e estarão em vigor apenas quando o TSO/ISPF Client Gateway do ISPF for usado pelo Developer para System z. (Esse é o padrão).

`_RSE_CMDSERV_OPTS=""`

Inicialização variável. Não modifique.

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

Use um perfil ISPF existente para a inicialização do ISPF. Remova o comentário e altere o nome do conjunto de dados para utilizar o perfil ISPF especificado.

As seguintes variáveis podem ser usadas no nome do conjunto de dados:

- `&SYSUID`, para substituir o ID do usuário do desenvolvedor
- `&SYSPREF`, para substituir o prefixo TSO do desenvolvedor

---

## ISPF.conf, Arquivo de Configuração do TSO/ISPF Client Gateway do ISPF

O TSO/ISPF Client Gateway do ISPF utiliza as definições em `ISPF.conf` para criar um ambiente válido para executar comandos do TSO e do ISPF em lote. O Developer para System z utiliza esse ambiente para executar alguns serviços baseados em MVS. Esses serviços incluem o serviço TSO Commands, o serviço SCLM Developer Toolkit e um método de inicialização alternativo do CARMA.

O `ISPF.conf` está localizado em `/etc/rdz/`, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa `FEK.SFEKSAMP(FEKSETUP)`. Consulte “Configuração da Customização” na página 15 para obter mais detalhes. É possível editar o arquivo com o comando do TSO **OEDIT**.

Linhas de comentários são iniciadas com um asterisco (\*) ao utilizar uma página de códigos US. As linhas de dados só podem ter uma diretiva e seu valor designado. Não são permitidos comentários na mesma linha. Continuações de linha não são suportadas. Ao concatenar nomes de conjuntos de dados, inclua-os na mesma linha e separe os nomes com uma vírgula (,).

Além de fornecer os nomes corretos para os conjuntos de dados do ISPF, você deve também incluir o nome do conjunto de dados de serviço TSO Commands, `FEK.SFEKPROC`, na instrução `SYSPROC` ou `SYSEXEC`, conforme mostrado no exemplo a seguir.

```
* REQUIRED:
sysproc=ISP.SISPCLIB,FEK.SFEKPROC
isplib=ISP.SISPMENU
isptlib=ISP.SISPTENU
ispplib=ISP.SISPPENU
ispslib=ISP.SISPSLIB
ispllib=ISP.SISPLLOAD

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

*Figura 9. ISPF.conf - Arquivo de Configuração do ISPF*

### Nota:

- Você pode incluir suas próprias instruções equivalentes ao DD e as concatenações do conjunto de dados para customizar o ambiente TSO,

assemelhando-se a um procedimento de logon do TSO. Consulte o Capítulo 16, “Customizando o Ambiente TSO”, na página 263 para obter mais detalhes.

- O TSO/ISPF Client Gateway poderá não funcionar adequadamente se você usar um produto (de terceiro) que intercepte comandos ISPF, como **ISPSTART**. Verifique a documentação desse produto sobre como ele pode ser desativado para o Developer for System z. Se o produto exigir a alocação de uma instrução DD específica para DUMMY, você pode simular isso no ISPF.conf alocando essa instrução DD para nullfile.

Por exemplo:

```
ISPTRACE=nullfile
```

- Ao usar a diretiva allocjob, tenha cuidado para não desfazer as definições DD feitas anteriormente em ISPF.conf.
- Deve-se esperar o encerramento de forma anormal 522 do módulo ISPZTS0 se o parâmetro JWT do membro parmlib SMFPRMxx estiver configurado com um valor abaixo do ISPF\_timeout em ISPF.conf. Isso não impacta as operações do Developer para System z, uma vez que o TSO/ISPF Client Gateway é reiniciado automaticamente quando necessário.
- As mudanças ficam ativas para todas as novas chamadas. A reinicialização do servidor não é necessária.

---

## Componentes Opcionais

As etapas de customização acima são para uma configuração básica do Developer para System z. Consulte os capítulos sobre os componentes opcionais para seus requisitos de customização:

- Capítulo 3, “(Opcional) CARMA (Common Access Repository Manager)”, na página 49
- Capítulo 4, “(Opcional) Application Deployment Manager”, na página 71
- Capítulo 5, “(Opcional) SCLM Developer Toolkit”, na página 79
- “(Opcional) Procedimento Armazenado do DB2” na página 87
- “(Opcional) Suporte de Idioma Bidirecional do CICS” na página 90
- “(Opcional) Criptografia SSL do RSE” na página 91
- “(Opcional) Rastreo de RSE” na página 94
- “(Opcional) Grupos de Propriedade Baseados em Host” na página 96
- “(Opcional) Projetos Baseados no Host” na página 97
- “(Opcional) Integração do File Manager” na página 98
- “(Opcional) Caracteres Não-editáveis” na página 99
- “(Opcional) Usando REXEC (ou SSH)” na página 100
- “(Opcional) Transação APPC para o Serviço de Comandos TSO” na página 102
- “(Opcional) Limpeza de WORKAREA” na página 105

---

## Verificação de Instalação

A descrição de vários programas de verificação de instalação (IVPs) está localizada em Capítulo 7, “Verificação de Instalação”, na página 107, porque alguns dos IVPs são para os componentes opcionais.

---

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

O Common Access Repository Manager (CARMA) é um auxílio de produtividade para os desenvolvedores criam Repository Access Managers (RAMs). Um RAM é uma API (Interface de Programação do Aplicativo) para os SCMs (Software Configuration Managers) baseados no z/OS.

Em seguida, os aplicativos gravados pelo usuário podem iniciar um servidor CARMA que carrega os RAMs e fornece uma interface padrão para acesso ao SCM.

O Developer para System z suporta vários métodos para iniciar um servidor CARMA, cada um com seus próprios benefícios e desvantagens.

- O método de “submissão em lote” inicia o servidor CARMA enviando uma tarefa. Esse é o método padrão usado nos arquivos de configuração de amostra fornecidos. O benefício desse método é que os logs do CARMA são facilmente acessados na saída de tarefas. Ele também permite o uso de JCL de servidor customizado para cada desenvolvedor, que é mantido pelo próprio desenvolvedor. Entretanto, esse método utiliza um iniciador de JES por desenvolvedor que iniciar um servidor CARMA.
- O método “CRASTART” inicia o servidor CARMA como uma sub tarefa dentro do RSE. Ele fornece uma configuração bastante flexível utilizando um arquivo de configuração separado, que define alocações de conjuntos de dados, e chamadas de programas necessárias para iniciar um servidor CARMA. Esse método fornece o melhor desempenho e utiliza o mínimo de recursos, mas requer que o módulo CRASTART esteja localizado no LPA.
- O método “TSO/ISPF Client Gateway” usa o TSO/ISPF Client Gateway do ISPF para criar um ambiente TSO ou ISPF, no qual o servidor CARMA é iniciado. Ele permite alocações de conjuntos de dados flexíveis utilizando as possibilidades do ISPF.conf. Entretanto, esse método não é adequado para acessar SCMs que interferem em operações normais do TSO ou ISPF.

---

### Requisitos e Lista de Verificação

Você precisará da ajuda de um administrador de segurança e um administrador de TCP/IP para concluir esta tarefa de customização, que exige os seguintes recursos ou tarefas de customização especiais:

- Intervalo de portas TCP/IP para comunicação interna
- regra de segurança para permitir que os desenvolvedores atualizem para arquivos CARMA VSAM
- (Opcional) Regra de segurança para permitir que usuários enviem tarefas CRA\*
- (opcional) Atualização de LPA

Para iniciar o uso do CARMA em seu site, é necessário executar as tarefas a seguir. A menos que especificado o contrário, todas as tarefas são obrigatórias.

1. Crie componentes necessários do CARMA. Para obter detalhes, consulte “Componentes do CARMA” na página 50.

2. Customização inicial dos arquivos de configuração do RSE para fazer interface com o CARMA. A customização completa depende do método escolhido para iniciar o CARMA. Para obter detalhes, consulte “Interface RSE para CARMA” na página 51.
3. Escolha um método para iniciar o CARMA e execute a customização necessária dos arquivos de configuração relacionados. Para obter detalhes, consulte:
  - “Inicialização do Servidor CARMA Usando Submissão em Lote” na página 53
  - “(Opcional) Inicialização Alternativa do Servidor CARMA Utilizando CRAFT” na página 55
  - “(Opcional) Inicialização Alternativa do Servidor CARMA Usando TSO/ISPF Client Gateway” na página 58
4. Ative, opcionalmente, os Repository Access Managers (RAMs) de amostra. Para obter detalhes, consulte “(Opcional) Ativando os RAM (Repository Access Managers) de Amostra” na página 60.
5. Opcionalmente, ative o CA Endevor® RAM. Para obter detalhes, consulte “(Opcional) Ativando o CA Endevor® SCM RAM” na página 61.
6. Crie, opcionalmente, CRAXJCL como substituição para IRXJCL. Para obter detalhes, consulte “(Opcional) IRXJCL versus CRAXJCL” na página 69.

**Nota:** Os membros de amostra referidos neste capítulo estão localizados em FEK.#CUST.\* e /etc/rdz, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP) . Consulte “Configuração da Customização” na página 15 para obter mais detalhes.

---

## Componentes do CARMA

Os seguintes componentes do CARMA devem ser customizados, independentemente do método de inicialização escolhido. Os membros de amostra referenciados abaixo estão localizados em FEK.#CUST.JCL, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes.

1. Customize e envie a JCL FEK.#CUST.JCL(CRA\$VDEF). Consulte a documentação em CRA\$VDEF para obter instruções de customização. CRA\$VDEF cria e prepara o conjunto de dados VSAM de configuração do CARMA, CRADEF.
2. Customize e envie a JCL FEK.#CUST.JCL(CRA\$VMSG). Consulte a documentação em CRA\$VMSG para obter instruções de customização. CRA\$VMSG cria e prepara o conjunto de dados VSAM de mensagens do CARMA, CRAMSG.
3. Customize e envie a JCL FEK.#CUST.JCL(CRA\$VSTR). Consulte a documentação em CRA\$VSTR para obter instruções de customização. CRA\$VSTR cria e prepara o conjunto de dados VSAM de informações customizadas do CARMA, CRASTRS.

**Nota:**

- Os VSAMs do CARMA criados com essas tarefas definem os RAMs de amostra. Consulte “(Opcional) Ativando o CA Endevor® SCM RAM” na página 61 para definir o CA Endevor® RAM.
- Consulte a tarefa de amostra FEK.#CUST.JCL(CRA\$UADD) se precisar mesclar as definições de um RAM (customizado) em uma configuração VSAM existente. Essa tarefa deve ser customizada e enviada para cada VSAM do CARMA que for alterado. Consulte o *Rational Developer para System z*



*Common Access Repository Manager Developer's Guide (SC23-7660)* para obter informações adicionais sobre a estrutura de registro usada pelos VSAMs diferentes do CARMA.

- Use a tarefa de amostra FEK.#CUST.JCL(CRA#UQRY) para extrair as definições ativas de um VSAM para um conjunto de dados sequencial.

## Notas de Migração VSAM do CARMA

O Developer para System z A versão 7.6.1 suporta um novo layout de estrutura de dados para o conjunto de dados VSAM de informações customizadas do CARMA, CRASTRS, para remover as limitações do comprimento das mensagens.

Anterior a versão 7.6.1. Developer para System z, as cadeias definidas no conjunto de dados VSAM de informações customizadas do CARMA são limitadas para comprimentos pré-definidos. Esta limitação força os desenvolvedores RAM a diminuir as cadeias descritivas ou usar plug-ins do lado do cliente para exibir as cadeias de duração longa.

Developer para System z A versão 7.6.1 suporta um novo layout de estrutura de dados de comprimento variável para o conjunto de dados VSAM de informações customizadas do CARMA CRASTRS, onde as cadeias são separadas por um caractere delimitador em vez de serem de comprimento fixo.

Customize e submeta o FEK.SFEKSAMP(CRA#VS2) JCL para converter o seu conjunto de dados VSAM de informações customizadas do CARMA de comprimento fixo existente CRASTRS para o novo formato de comprimento variável.

### Nota:

- Iniciando com a versão 7.6.1, o conjunto de dados VSAM de informações customizadas do CARMA de amostra é enviado em formato de comprimento variável.
- Iniciando com a versão 7.6.1, o módulo de carregamento do CARMA CRASERV suporta os formatos de comprimento fixo e o de comprimento variável para o conjunto de dados VSAM de informações customizadas do CARMA.
- Versões anteriores do módulo de carregamento do CARMA não suportam o formato de comprimento variável e produzirão cadeias distorcidas quando usadas com um conjunto de dados VSAM de informações customizadas do CARMA com comprimento variável.

---

## Interface RSE para CARMA

O servidor CARMA fornece uma API padrão para outros produtos baseados em host para acessar um ou mais SCMs (Software Configuration Managers). Entretanto, ele não fornece métodos para comunicação direta com um PC cliente. Para isso, ele depende de outros produtos, como o servidor RSE. O servidor RSE utiliza as configurações de CRASRV.properties para iniciar e se conectar a um servidor CARMA.

O CRASRV.properties está localizado em /etc/rdz/, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte "Configuração da Customização" na página 15 para obter mais detalhes. É possível editar o arquivo com o comando do TSO OEDIT.

**Nota:** A tarefa iniciada RSED deve ser reiniciada para assimilar as mudanças feitas.

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

*Figura 10. CRASRV.properties – Arquivo de configuração CARMA*

#### **port.start**

Primeira porta utilizada para comunicação entre o CARMA e o servidor RSE. 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 149 para obter informações adicionais.

#### **port.range**

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

#### **startup.script.name**

Define o caminho absoluto do script de inicialização do CARMA. O padrão é `/usr/lpp/rdz/bin/carma.startup.rex`. Esse REXX exec acionará a inicialização de um servidor CARMA.

#### **clist.dsname**

Define o método de inicialização para o servidor CARMA.

- `*CRASTART` indica que o servidor CARMA deve ser iniciado como uma subtarefa dentro do RSE utilizando `CRASTART`. Consulte “(Opcional) Inicialização Alternativa do Servidor CARMA Utilizando `CRASTART`” na página 55 para obter mais detalhes. Se você especificar `*CRASTART`, será necessário especificar também as diretivas `crastart.*`.
- `*ISPF` indica que o servidor CARMA deve ser iniciado utilizando o TSO/ISPF Client Gateway do ISPF. Consulte o “(Opcional) Inicialização Alternativa do Servidor CARMA Usando TSO/ISPF Client Gateway” na página 58 para obter mais detalhes.
- Qualquer outro valor define o local de `CRASUBMT CLIST`, utilizando convenções de nomenclatura ao estilo TSO. Com aspas (‘), o nome do conjunto de dados é uma referência absoluta, sem aspas (‘), o nome do conjunto de dados é prefixado com o ID do usuário do cliente, e não com o prefixo do TSO. Esse último requer que todos os usuários do CARMA mantenham seu próprio `CRASUBMT CLIST`.

O padrão é `'FEK.#CUST.CNTL(CRASUBMT)'`. Esse `CLIST` iniciará um servidor CARMA ao abrir uma conexão utilizando o método de submissão em lote.



**crastart.stub**

Stub do z/OS UNIX para chamar CRAFTART. O padrão é /usr/lpp/rdz/bin/CRAFTART. Esse stub disponibiliza o módulo de carregamento CRAFTART baseado em MVS para os processos do z/OS UNIX. Essa diretiva será usada apenas se a diretiva `clist.dsname` tiver \*CRAFTART como valor.

**crastart.configuration.file**

Especifica o nome do arquivo de configuração CRAFTART. O padrão é /etc/rdz/crastart.conf. Esse arquivo especifica as alocações de conjuntos de dados e chamadas de programas necessárias para iniciar um servidor CARMA. Essa diretiva será usada apenas se a diretiva `clist.dsname` tiver \*CRAFTART como valor.

**crastart.syslog**

Especifica quantas informações são gravadas no log do sistema, enquanto CRAFTART inicia um servidor CARMA. O padrão é Partial. Valores válidos são:

A (Tudo)	Todas as informações de rastreamento são impressas no SYSLOG
P (Parcial)	Apenas informações de conexão, desconexão e erro são impressas no SYSLOG
qualquer outra coisa	Apenas condições de erro são impressas no SYSLOG

Essa diretiva será usada apenas se a diretiva `clist.dsname` tiver \*CRAFTART como valor.

**crastart.timeout**

O período de tempo, em segundos, antes de um servidor CARMA ser encerrado devido à falta de atividade. O padrão é 420 (7 minutos). Essa diretiva será usada apenas se a diretiva `clist.dsname` tiver \*CRAFTART como valor.

**crastart.steplib**

O local do módulo CRAFTART quando acessado por meio da diretiva STEPLIB em `rsed.envvars`. O padrão é FEK.SFEKLPA. Remova o comentário e customize essa diretiva se o módulo CRAFTART não puder fazer parte de LPA ou LINKLIST. Observe que problemas de controle do programa e de APF podem surgir se o módulo CRAFTART não estiver em LPA. Essa diretiva será utilizada apenas se a diretiva `clist.dsname` tiver \*CRAFTART como valor.

**crastart.tasklib**

Nome alternativo para o nome TASKLIB DD em `crastart.conf`. O padrão é TASKLIB. Remova o comentário e customize essa diretiva se o nome DD TASKLIB tiver um significado especial para SCM ou RAM e não puder ser usado como substituto de STEPLIB. Essa diretiva será usada apenas se a diretiva `clist.dsname` tiver \*CRAFTART como valor.

---

## Inicialização do Servidor CARMA Usando Submissão em Lote

As informações nesta seção descrevem como configurar o método padrão para o Developer para System z iniciar um servidor CARMA. Essa etapa de customização poderá ser ignorada se você utilizar outro método de inicialização.

O Developer para System z usa, por padrão, o método de inicialização do servidor CARMA de submissão em lote que não exige que o módulo CRAFTART esteja no

LPA e que não depende do TSO/ISPF Client Gateway. O método envia o servidor CARMA como uma tarefa em lote de longa execução no JES.

## Ajustar CRASRV.properties

O servidor RSE utiliza as configurações em `/etc/rdz/CRASRV.properties` para iniciar e se conectar a um servidor CARMA, conforme documentado em “Interface RSE para CARMA” na página 51. É possível editar o arquivo com o comando do TSO **OEDIT**. Observe que o RSE deve ser reiniciado para que as alterações sejam efetivadas.

Altere o valor da diretiva `clist.dsname` para o conjunto de dados e nome do membro do CLIST de inicialização do servidor CARMA CRASUBMT, como mostrado no seguinte exemplo. Consulte “Interface RSE para CARMA” na página 51 para obter informações adicionais sobre diretivas diferentes.

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

*Figura 11. CRASRV.properties - inicialização do CARMA usando submissão em lote*

## Ajustar CRASUBMT

Customize a CLIST CRASUBMT, conforme mostrado na amostra de código a seguir. Consulte a documentação em CRASUBMT para obter instruções de customização. A CLIST CRASUBMT envia um servidor CARMA.

O CRASUBMT está localizado em `FEK.#CUST.CNTL`, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa `FEK.SFEKSAMP(FEKSETUP)`. Consulte “Configuração da Customização” na página 15 para obter mais detalhes.

```

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

```

Figura 12. CRASUBMT - inicialização do CARMA utilizando submissão em lote

#### Nota:

- Você pode incluir suas próprias instruções DD e as concatenações do conjunto de dados para customizar o ambiente CARMA TSO, simulando um procedimento de logon do TSO.
- Opcionalmente, você pode alterar o valor de tempo limite do CARMA modificando a linha PROC 1 PORT TIMEOUT(420) em FEK.#CUST.CNTL(CRASUBMT) CLIST. 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).
- Detalhes do processo de inicialização do CARMA são mostrados em rsecomm.log. Consulte “(Opcional) Rastreio de RSE” na página 94 para obter informações adicionais sobre a configuração do nível de detalhes de rsecomm.log.
- As mudanças são efetivadas para todos os servidores CARMA iniciados após a atualização.

## (Opcional) Inicialização Alternativa do Servidor CARMA Utilizando CRASTART

As informações nesta seção descrevem como configurar um método alternativo para o Developer para System z iniciar um servidor CARMA. Essa etapa de customização poderá ser ignorada se você utilizar outro método de inicialização.

O Developer para System z suporta um método de inicialização alternativo do servidor CARMA que não depende do TSO/ISPF Client Gateway e que não envia uma tarefa do servidor utilizando um iniciador JES. O método utiliza CRASTART para iniciar o servidor CARMA como uma sub tarefa com RSE e é semelhante ao serviço TSO/ISPF Client Gateway.

**Nota:** Detalhes do processo de inicialização do CARMA são mostrados em `rsecomm.log`. Consulte “(Opcional) Rastreamento de RSE” na página 94 para obter informações adicionais sobre a configuração do nível de detalhes de `rsecomm.log`.

## Ajustar CRASRV.properties

O servidor RSE utiliza as configurações em `/etc/rdz/CRASRV.properties` para iniciar e se conectar a um servidor CARMA, conforme documentado em “Interface RSE para CARMA” na página 51. É possível editar o arquivo com o comando do TSO **OEDIT**. Observe que o RSE deve ser reiniciado para que as alterações sejam efetivadas.

Altere o valor da diretiva `clist.dsname` para `*CRASTART` e forneça os valores corretos para as diretivas `crastart.*`, como mostrado no seguinte exemplo. Consulte “Interface RSE para CARMA” na página 51 para obter informações adicionais sobre diretivas diferentes.

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

*Figura 13. CRASRV.properties - Inicialização alternativa do CARMA \*CRASTART*

**Nota:** O encerramento de forma anormal 522 do sistema para módulo CRASERV ocorrerá se o parâmetro `JWT` no membro `SMFPRMxx parmlib` estiver configurado como um valor mais baixo do que o valor do tempo limite em `CRASRV.properties`. Isso não causa impacto nas operações do CARMA, pois o servidor é reiniciado automaticamente, se necessário.

## Ajustar crastart.conf

Mantenha uma impressão do CRASUBMT customizado (consulte “Inicialização do Servidor CARMA Usando Submissão em Lote” na página 53 ) pronta para facilitar a referência durante essa etapa de customização. A impressão será importante mesmo se você não tiver customizado o membro.

CRASTART utiliza as definições em `crastart.conf` para criar um ambiente válido para executar comandos TSO e ISPF em lote. O Developer para System z utiliza esse ambiente para executar o servidor CARMA chamado CRASERV.

O `crastart.conf` está localizado em `/etc/rdz/`, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa `FEK.SFEKSAMP(FEKSETUP)`. Consulte “Configuração da Customização” na página 15 para obter mais detalhes. É possível editar o arquivo com o comando do TSO **OEDIT**.

**Nota:** As mudanças são efetivadas para todos os servidores CARMA iniciados após a atualização.

As etapas de customização a seguir são necessárias para ajustar o arquivo de configuração exibido na amostra de código abaixo.

- Inclua os conjuntos de dados alocados para a concatenação STEPLIB do procedimento CRASUBMT para a instrução TASKLIB em crastart.conf.
- Crie entradas para os DDs obrigatórios CRADEF, CRAMSG e CRSTRS do VSAM do CARMA. Use os nomes de conjuntos de dados fornecidos no procedimento CRASUBMT (customizado).
- Inclua qualquer instrução DD customizada e as concatenações de conjuntos de dados relacionadas, disponíveis no procedimento CRASUBMT (customizado). Por exemplo, inclua a instrução CRARAM1 DD e o nome do conjunto de dados, caso utilize a amostra PDS RAM. Observe que você pode utilizar nomes de conjuntos de dados (alocados com DISP=SHR), SYSOUT e constructos DUMMY.
- Opcionalmente, inclua qualquer comando BPXWDYN utilizando a instrução -COMMAND. Pode haver várias instruções -COMMAND. BPXWDYN permite que você faça alocações mais complexas, incluindo a criação de conjuntos de dados temporários, disposições diferentes de SHR, alocações para outros subsistemas, etc. Consulte *Usando REXX e os Serviços do Sistema z/OS UNIX (SA22-7806)* para obter informações adicionais do BPXWDYN.
- Selecione o método de chamada de programa desejado utilizando a instrução PROGRAM. O método recomendado é "PROGRAM=IKJEFT01 CRASERV &CRAPRM1. &CRAPRM2.", já que ele fornece um ambiente TSO que pode manipular uma combinação de conjuntos de dados APF e não-APF. Consulte a amostra crastart.conf para conhecer outros métodos.

**Nota:** As definições de crastart.conf não podem ser divididas em várias linhas.

\* crastart.conf - Opções de alocação do CARMA

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

*Figura 14. crastart.conf - Inicialização alternativa do CARMA \*CRASTART*

As seguintes variáveis podem ser usadas no arquivo de configuração:

*Tabela 10. Variáveis de crastart.conf*

&CRAUSER.	ID do usuário de logon do cliente.
&CRADATE.	Data atual no formato Dyyyddd (7 caracteres do calendário juliano).
&CRATIME.	Hora atual no formato Thmmss (hora - minuto - segundo).

Tabela 10. Variáveis de *crastart.conf* (continuação)

&CRAPRM3. através de &CRAPRM9.	Variáveis adicionais com valores designados pelo usuário. O uso destas variáveis requer a customização da inicialização do CARMA do REXX referenciado no <code>startup.script.name</code> em <code>CRASRV.properties</code> .  Quando você usa estas variáveis, é recomendável customizar uma cópia do REXX de inicialização padrão, <code>/usr/lpp/rdz/bin/carma.startup.rex</code> e apontar <code>startup.script.name</code> para esta cópia. Isto é para evitar que você perca o seu trabalho quando a manutenção atualizar o REXX padrão.
símbolo do sistema	Qualquer símbolo do sistema definido em <code>SYS1.PARMLIB(IEASYMxx)</code>
-<DD>	Um traço (-) seguido de um nome DD definido anteriormente atua como uma referência anterior <code>*.ddname</code> na JCL. O DD original deve ser alocado utilizando a instrução <code>-COMMAND</code> .

**Nota:** Não há nenhuma variável para o prefixo TSO, pois o TSO não está ativo quando o arquivo de configuração é interpretado.

## (Opcional) Inicialização Alternativa do Servidor CARMA Usando TSO/ISPF Client Gateway

As informações nesta seção descrevem como configurar um método alternativo para o Developer para System z iniciar um servidor CARMA. Essa etapa de customização poderá ser ignorada se você utilizar outro método de inicialização.

O Developer para System z suporta um método de inicialização alternativo do servidor CARMA que não requer que o módulo `CRASTART` esteja no LPA e que não envia uma tarefa do servidor utilizando um iniciador JES. O método utiliza o TSO/ISPF Client Gateway do ISPF e é semelhante à forma padrão de acesso ao serviço TSO Commands.

**Nota:** Detalhes do processo de inicialização do CARMA são mostrados em `rsecomm.log`. Consulte “(Opcional) Rastreo de RSE” na página 94 para obter informações adicionais sobre a configuração do nível de detalhes de `rsecomm.log`.

### Ajustar `CRASRV.properties`

O servidor RSE utiliza as configurações em `/etc/rdz/CRASRV.properties` para iniciar e se conectar a um servidor CARMA, conforme documentado em “Interface RSE para CARMA” na página 51. É possível editar o arquivo com o comando do TSO `OEDIT`. Observe que o RSE deve ser reiniciado para que as alterações sejam efetivadas.

Altere o valor da diretiva `clist.dsname` para `*ISPF`, como mostrado no seguinte exemplo. Consulte “Interface RSE para CARMA” na página 51 para obter informações adicionais sobre diferentes diretivas.

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

Figura 15. CRASRV.properties - Inicialização alternativa do CARMA \*ISPF

## Ajustar ISPF.conf

Mantenha uma impressão do CRASUBMT customizado (consulte “Inicialização do Servidor CARMA Usando Submissão em Lote” na página 53 ) pronta para facilitar a referência durante essa etapa de customização. A impressão será importante mesmo se você não tiver customizado o membro.

O TSO/ISPF Client Gateway do ISPF usa as definições no ISPF.conf para criar um ambiente válido para executar comandos TSO e ISPF em lote. O Developer para System z utiliza esse ambiente para executar o servidor CARMA.

O ISPF.conf está localizado em /etc/rdz/, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes. É possível editar o arquivo com o comando do TSO **OEDIT**.

**Nota:** As mudanças são efetivadas para todos os servidores CARMA iniciados após a atualização.

As etapas de customização a seguir são necessárias para ajustar o arquivo de configuração exibido na amostra de código abaixo.

- Forneça os nomes corretos para os conjuntos de dados do ISPF obrigatórios (entretanto, não aloque ISPPROF, ele é alocado dinamicamente).
- Anexe o proclib do Developer para System z, FEK.SFEKPROC, à instrução SYSPROC ou SYSEXEC, para que o CRASRVI exec possa ser localizado pelo sistema. Esse exec inicia o servidor CARMA (e, assim, substitui a SYSTSIN DD do CRASUBMT).
- Inclua a concatenação DD STEPLIB do procedimento CRASUBMT na instrução ispllib.
- Crie entradas obrigatórias para as DDs CRADEF, CRAMSG e CRSTRS do VSAM do CARMA. Use os nomes de conjuntos de dados fornecidos no procedimento CRASUBMT (customizado).
- Inclua qualquer instrução DD e a concatenação do conjunto de dados relacionado, disponível no procedimento CRASUBMT (customizado. Por exemplo, inclua a instrução CRARAM1 DD e o nome do conjunto de dados, caso utilize a amostra PDS RAM. Observe que você pode utilizar apenas os nomes de conjuntos de dados (alocados com DISP=SHR).
- Opcionalmente, remova o comentário e customize a diretiva allocexec para fazer alocações adicionais utilizando um exec.

**Nota:** Não inclua os DDs SYSTSIN, SYSTSOUT ou CARMALOG, nem qualquer outra instrução DD que use construções JES, como dados do fluxo de entrada e SYSOUT=. Essas entradas devem ser convertidas para utilizar conjuntos de dados.



```

sysproc=ISP.SISPCLIB,FEK.SFEKPROC
ispllib=FEK.SFEKLOAD
ispmllib=ISP.SISPMENU
isptllib=ISP.SISPTENU
ispplib=ISP.SISPPENU
ispslib=ISP.SISPSLIB
CRADEF =FEK.#CUST.CRADEF
CRAMSG =FEK.#CUST.CRAMSG
CRASTRS=FEK.#CUST.CRASTRS
*CRARAM1=FEK.#CUST.CRARAM1
allocjob=FEK.#CUST.CNTL(CRAISPRX)

```

Figura 16. ISPF.conf - Inicialização alternativa do CARMA \*ISPF

A DD CARMALOG refere-se a SYSOUT=\* por padrão, que não pode ser mapeada em ISPF.conf. Não é possível mapear o DD diretamente para um conjunto de dados, já que todos os usuários do Developer para System z estarão utilizando o mesmo arquivo ISPF.conf e, portanto, os mesmos conjuntos de dados.

No entanto, conforme descrito no Capítulo 16, “Customizando o Ambiente TSO”, na página 263, seção “Avançado – Usando um Exec de Alocação” na página 265, você pode utilizar um exec de alocação para criar e alocar um conjunto de dados baseado no ID de usuário ativo. Consulte o membro de amostra CRAISPRX no conjunto de dados FEK.#CUST.CNTL como um exemplo que aloca DD CARMALOG para o nome do conjunto de dados  
TSOPREFIX'. 'USERID'.CRA. 'TIMESTAMP'.CARMALOG'.

**Nota:**

- Ao usar a diretiva allocjob, tenha cuidado para não desfazer as definições DD feitas anteriormente em ISPF.conf.
- Deve-se esperar o encerramento de forma anormal 522 do módulo CRASERV se o parâmetro JWT do membro parmlib SMFPRMxx estiver configurado com um valor abaixo do ISPF\_timeout em ISPF.conf. Isso não causa impacto nas operações do CARMA, pois o servidor é reiniciado automaticamente, se necessário.

---

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

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

**Nota:** Os RAMs de amostra são fornecidos com o propósito de testar a configuração do ambiente CARMA e como exemplos para que você desenvolva seus próprios RAMs. NÃO utilize os RAMs de amostra fornecidos em um ambiente de produção.

Consulte *Rational Developer para System z Common Access Repository Manager Developer's Guide* (SC23-7660) para obter informações adicionais sobre RAMs de amostra e código de origem de amostra fornecidos.

Os membros de amostra referenciados abaixo estão localizados em FEK.#CUST.JCL, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes.



## Ativando o PDS RAM

O PDS RAM fornece uma lista de conjuntos de dados semelhante a **Arquivos MVS -> Meus Conjuntos de Dados** na visualização Sistemas Remotos. O PDS RAM usa o RAM ID 0 por padrão.

**Nota:** O PDS RAM espera que o CARMA seja iniciado dentro do ISPF (usando ISPSTART).

1. Customize e envie a JCL FEK.#CUST.JCL(CRA#VPDS). Consulte a documentação em CRA#VPDS para obter instruções de customização. CRA#VPDS cria e prepara o conjunto de dados VSAM de mensagens do PDS RAM.
2. Inclua a instrução DD CRARAM1 no método de inicialização selecionado do CARMA e forneça o nome do conjunto de dados do VSAM de mensagem do PDS RAM.

## Ativando o RAM do SCLM

O SCLM RAM fornece uma entrada básica no SCLM, o Software Configuration Manager do ISPF. O SCLM RAM usa o RAM ID 1 por padrão.

**Nota:** O SCLM RAM espera que o CARMA seja iniciado dentro do ISPF (usando ISPSTART).

1. Customize e envie a JCL FEK.#CUST.JCL(CRA#VSLM). Consulte a documentação em CRA#VSLM para obter instruções de customização. CRA#VSLM cria e prepara o conjunto de dados VSAM de mensagens do SCLM RAM.
2. Inclua a instrução DD CRARAM2 no método de inicialização selecionado do CARMA e forneça o nome do conjunto de dados do VSAM de mensagem do SCLM RAM.
3. Customize a JCL FEK.#CUST.JCL(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 FEK.#CUST.JCL(CRA#ASLM) uma vez antes de usar o CARMA com o SCLM RAM. Se isso não for feito, o resultado será um erro de alocação.

## Ativando o RAM da Estrutura

O RAM de base fornece uma estrutura de base que pode ser usada para desenvolver seus próprios RAMs. O RAM de base usa o RAM ID 3 por padrão.

1. Customize e envie a JCL FEK.#CUST.JCL(CRA#CRAM). Consulte a documentação em CRA#CRAM para obter instruções de customização. CRA#CRAM compila a RAM da estrutura.
2. Inclua a biblioteca de carregamento que contém o módulo RAM da estrutura compilada, CRARAMSA, na STEPLIB DD do método de inicialização CARMA selecionado (TASKLIB DD do método CRASTART).

---

## (Opcional) Ativando o CA Endeavor® SCM RAM

O IBM® Rational® Developer para System z Interface para CA Endeavor® Software Configuration Manager fornece aos clientes do Developer para System z acesso direto ao CA Endeavor® SCM. A partir daqui, o IBM® Rational® Developer para System z Interface para CA Endeavor® SCM é abreviado como CA Endeavor® SCM RAM (Repository Access Manager).

Em contradição com os RAMs de amostras documentados nesta publicação, o CA Endeavor® SCM RAM é um tipo de produção RAM. Você não deve ativar os dois tipos de RAM na mesma configuração.

Atenção: As tarefas de configuração fornecidas para CA Endeavor® SCM RAM substituem a configuração do CARMA ativa por uma que somente suspende o CA Endeavor® SCM RAM.

**Nota:** O método de inicialização do TSO/ISPF Client Gateway não pode ser usado junto com o CA Endeavor® SCM RAM.

## Requisitos e Lista de Verificação

Você precisa da assistência de um administrador de segurança e um administrador TCP/IP para completar esta tarefa de customização, que requer os seguintes recursos ou tarefas de customização especiais:

- Intervalo de portas TCP/IP para comunicação interna
- (Opcional) Regra de segurança para permitir que usuários enviem tarefas CRA\*
- (opcional) Atualização de LPA

Para iniciar usando o CA Endeavor® SCM RAM no seu site, você deve executar as seguintes tarefas. A menos que especificado o contrário, todas as tarefas são obrigatórias.

1. Aloque e prepare os conjuntos de dados VSAM que definem o CA Endeavor® SCM RAM no CARMA. Para obter detalhes, consulte “Defina o CA Endeavor® SCM RAM”.
2. Escolha o seu método de inicialização preferido, submissão em lote ou CRASTART, e faça a customização requerida dos arquivos de configuração relacionados. Para obter detalhes, consulte:
  - “Inicialização do CA Endeavor® SCM RAM Usando Submissão em Lote” na página 63
  - “Inicialização do CA Endeavor® SCM RAM Usando CRASTART” na página 65
3. Opcionalmente, customize o executável de alocação usado para a alocação dinâmica de conjuntos de dado específicos do usuário. Para obter detalhes, consulte “(Opcional) Customize CRANDVRA” na página 66.
4. Opcionalmente, customize os arquivos de configuração específicos do CA Endeavor® SCM RAM. Para obter detalhes, consulte “(Opcional) Customize o CA Endeavor® SCM RAM” na página 67.

## Defina o CA Endeavor® SCM RAM

Os seguintes componentes do CARMA devem ser customizados, independentemente do método de inicialização escolhido. Os membros de amostra referenciados abaixo estão localizados em FEK.#CUST.JCL, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes.

1. Customize e envie a JCL FEK.#CUST.JCL(CRA#VCAD). Consulte a documentação em CRA\$VDEF para obter instruções de customização. CRA#VCAD cria e prepara o conjunto de dados VSAM de configuração do CARMA, CRADEF.
2. Customize e envie a JCL FEK.#CUST.JCL(CRA\$VMSG). Consulte a documentação em CRA\$VMSG para obter instruções de customização. CRA\$VMSG cria e prepara o conjunto de dados VSAM de mensagens do CARMA, CRAMSG.

**Nota:** Essa é a mesma tarefa dos RAMs de amostra.

3. Customize e envie a JCL FEK.#CUST.JCL(CRA#VCAS). Consulte a documentação em CRA\$VSTR para obter instruções de customização. CRA#VCAS cria e prepara o conjunto de dados VSAM de informações customizadas do CARMA, CRASTRS.

**Nota:**

- O CA Endeavor® SCM RAM usa o RAM ID 0 por padrão.
- Consulte a tarefa de amostra FEK.#CUST.JCL(CRA#UADD) se precisar mesclar as definições de um RAM (customizado) em uma configuração VSAM existente. Essa tarefa deve ser customizada e enviada para cada VSAM do CARMA que for alterado. Consulte o *Rational Developer para System z Common Access Repository Manager Developer's Guide* (SC23-7660) para obter informações adicionais sobre a estrutura de registro usada pelos VSAMs diferentes do CARMA.
- Use a tarefa de amostra FEK.#CUST.JCL(CRA#UQRY) para extrair as definições ativas de um VSAM para um conjunto de dados sequencial.

## Inicialização do CA Endeavor® SCM RAM Usando Submissão em Lote

Não execute esta etapa se você usar o método CRASTART para iniciar o servidor CARMA com o CA Endeavor® SCM RAM.

O Developer para System z pode usar o método de inicialização do Servidor CARMA de submissão em lote para iniciar o CA Endeavor® SCM RAM. O método envia o servidor CARMA como uma tarefa em lote de longa execução no JES.

Consulte “Inicialização do Servidor CARMA Usando Submissão em Lote” na página 53 para obter informações adicionais sobre o método de inicialização de submissão em lote.

### Ajustar CRASRV.properties

O servidor RSE utiliza as configurações em /etc/rdz/CRASRV.properties para iniciar e se conectar a um servidor CARMA, conforme documentado em “Interface RSE para CARMA” na página 51. É possível editar o arquivo com o comando do TSO OEDIT. Observe que o RSE deve ser reiniciado para que as alterações sejam efetivadas.

Altere o valor da diretiva clist.dsname para o conjunto de dados e nome do membro do CLIST de inicialização do servidor CARMA CRASUBCA, como mostrado no seguinte exemplo. Consulte “Interface RSE para CARMA” na página 51 para obter informações adicionais sobre diretivas diferentes.

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

Figura 17. Figura x1. CRASRV.properties - inicialização do CA Endeavor® SCM RAM usando submissão em lote

### Ajustar CRASUBCA

Customize o CLIST CRASUBCA, como mostrado na amostra de código a seguir. Consulte a documentação dentro do CRASUBCA para obter instruções de customização. O CLIST CRASUBCA CLIST submete um servidor CARMA para o CA Endeavor® SCM.

O CRASUBCA está localizado no FEK.#CUST.CNTL, a menos que você tenha especificado um local diferente ao customizar e submeter a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes.

```
PROC 1 PORT TIMEOUT(420)
SUBMIT * END($$)
//CRA&PORT JOB CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1)
//RUN      EXEC PGM=IKJEFT01,DYNAMNBR=125,REGION=0M,TIME=NOLIMIT,
//          PARM='%CRANDVRA NDVRC1 PGM(CRASERV) PARM(&PORT &TIMEOUT)'
//STEPLIB DD DISP=SHR,DSN=FEK.SFEKLOAD
//          DD DISP=SHR,DSN=CA.NDVR.AUTHLIB
//          DD DISP=SHR,DSN=CA.NDVRU.AUTHLIB
//CRADEF DD DISP=SHR,DSN=FEK.#CUST.CRADEF
//CRMSG DD DISP=SHR,DSN=FEK.#CUST.CRMSG
//CRASTRS DD DISP=SHR,DSN=FEK.#CUST.CRASTRS
//*
//SYSPROC DD DISP=SHR,DSN=ISP.SISPCLIB
//          DD DISP=SHR,DSN=FEK.SFEKPROC
//ISPEXEC DD DISP=SHR,DSN=ISP.SISPEXEC
//ISPLIB DD DISP=SHR,DSN=ISP.SISPMENU
//ISPPLIB DD DISP=SHR,DSN=ISP.SISPPENU
//ISPSLIB DD DISP=SHR,DSN=ISP.SISPSENU
//ISPTLIB DD DISP=SHR,DSN=ISP.SISPTENU
//ISPTL0 DD DISP=(NEW,DELETE,DELETE),UNIT=SYSALLDA,
//          SPACE=(TRK,(1,1)),LRECL=80,RECFM=FB
//ISPTL1 DD DISP=(NEW,DELETE,DELETE),UNIT=SYSALLDA,
//          SPACE=(TRK,(1,1)),LRECL=80,RECFM=FB
//ISPPROF DD DISP=(NEW,DELETE,DELETE),UNIT=SYSALLDA,
//          SPACE=(TRK,(1,1,5)),LRECL=80,RECFM=FB
//*
//CARMALOG DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD DUMMY
//*
//CONLIB DD DISP=SHR,DSN=CA.NDVR.CONLIB
//JCLOUT DD SYSOUT=(A,INTRDR),DCB=(LRECL=80,RECFM=F,BLKSIZE=80)
//EXT1ELM DD DISP=(NEW,DELETE),UNIT=SYSALLDA,
//          RECFM=VB,LRECL=4096,BLKSIZE=27998,SPACE=(TRK,(5,5))
//EXT1DEP DD DISP=(NEW,DELETE),UNIT=SYSALLDA,
//          RECFM=VB,LRECL=4096,BLKSIZE=27998,SPACE=(TRK,(5,5))
//MSG3FILE DD DISP=(NEW,DELETE),UNIT=SYSALLDA,
//          RECFM=FB,LRECL=133,BLKSIZE=27930,SPACE=(TRK,(5,5))
//C1MSG51 DD DISP=(NEW,DELETE),UNIT=SYSALLDA,
//          RECFM=FB,LRECL=133,BLKSIZE=27930,SPACE=(TRK,(5,5))
//C1EXMSG5 DD DISP=(NEW,DELETE),UNIT=SYSALLDA,
//          RECFM=FB,LRECL=133,BLKSIZE=27930,SPACE=(TRK,(5,5))
//TYPEMAP DD DISP=SHR,DSN=FEK.#CUST.PARMLIB(CRATMAP)
//SHOWVIEW DD DISP=SHR,DSN=FEK.#CUST.PARMLIB(CRASHOW)
$$
EXIT CODE(0)
```

Figura 18. Figura x2. Inicialização do CRASUBCA - CA Endevor® SCM RAM usando submissão em lote

#### Nota:

- Você pode incluir suas próprias instruções DD e as concatenações do conjunto de dados para customizar o ambiente CARMA TSO, simulando um procedimento de logon do TSO.
- Opcionalmente, você pode alterar o valor de tempo limite do CARMA ao modificar a linha PROC 1 PORT TIMEOUT(420) no CLIST. 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).

- Detalhes do processo de inicialização do CARMA são mostrados em rsecomm.log. Consulte “(Opcional) Rastreo de RSE” na página 94 para obter informações adicionais sobre a configuração do nível de detalhes de rsecomm.log.
- As mudanças são efetivadas para todos os servidores CARMA iniciados após a atualização.

## Inicialização do CA Endeavor® SCM RAM Usando CRAFTSTART

Não execute esta etapa se você usar o método de submissão em lote para iniciar o servidor CARMA com o CA Endeavor® SCM RAM.

O Developer para System z pode usar o método CRAFTSTART de inicialização do servidor CARMA para iniciar o CA Endeavor® SCM RAM. O método usa o CRAFTSTART para iniciar o servidor CARMA como uma sub tarefa dentro do RSE.

Consulte “(Opcional) Inicialização Alternativa do Servidor CARMA Utilizando CRAFTSTART” na página 55 para obter informações adicionais sobre o método de inicialização CRAFTSTART.

**Nota:** Detalhes do processo de inicialização do CARMA são mostrados em rsecomm.log. Consulte “(Opcional) Rastreo de RSE” na página 94 para obter informações adicionais sobre a configuração do nível de detalhes de rsecomm.log.

### Ajustar CRASRV.properties

O servidor RSE utiliza as configurações em /etc/rdz/CRASRV.properties para iniciar e se conectar a um servidor CARMA, conforme documentado em “Interface RSE para CARMA” na página 51. É possível editar o arquivo com o comando do TSO **OEDIT**. Observe que o RSE deve ser reiniciado para que as alterações sejam efetivadas.

Altere o valor da diretiva clist.dsname para \*CRAFTSTART e forneça os valores corretos para as diretivas craftstart.\*, como mostrado no seguinte exemplo. Consulte “Interface RSE para CARMA” na página 51 para obter informações adicionais sobre diretivas diferentes.

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

Figura 19. Figura x3. CRASRV.properties - inicialização do CA Endeavor® SCM RAM usando CRAFTSTART

**Nota:** O encerramento de forma anormal 522 do sistema para módulo CRASRV ocorrerá se o parâmetro JWT no membro SMFPRMxx parmlib estiver configurado como um valor mais baixo do que o valor do tempo limite em CRASRV.properties. Isto não impacta nas operações do CARMA porque o servidor é reiniciado automaticamente, caso necessário.

## Ajuste crastart.endevor.conf

O CRASTART usa as definições no crastart.endevor.conf para criar um ambiente válido (TSO/ISPF) para invocar o CA Endevor® SCM. O Developer para System z usa este ambiente para executar o CA Endevor® SCM RAM.

O crastart.endevor.conf está localizado em /etc/rdz/, a menos que você tenha especificado um local diferente ao customizar e submeter a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes. É possível editar o arquivo com o comando do TSO OEDIT.

**Nota:** As mudanças são efetivadas para todos os servidores CARMA iniciados após a atualização.

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

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

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

TYPEMAP = FEK.#CUST.PARMLIB(CRATMAP)
SHOWVIEW= FEK.#CUST.PARMLIB(CRASHOW)
CONLIB   = CA.NDVR.CONLIB
-COMMAND=ALLOC FI(JCLOUT) SYSOUT(A) WRITER(INTRDR) RECFM(F) LRECL(80)
BLKSIZE(80)
-COMMAND=ALLOC FI(EXT1ELM) NEW DELETE DSORG(PS) RECFM(V,B) LRECL(4096)
BLKSIZE(27998) SPACE(5,5) TRACKS UNIT(SYSALLDA)
-COMMAND=ALLOC FI(EXT1DEP) NEW DELETE DSORG(PS) RECFM(V,B) LRECL(4096)
BLKSIZE(27998) SPACE(5,5) TRACKS UNIT(SYSALLDA)
-COMMAND=ALLOC FI(MSG3FILE) NEW DELETE DSORG(PS) RECFM(F,B) LRECL(133)
BLKSIZE(27930) SPACE(5,5) TRACKS UNIT(SYSALLDA)
-COMMAND=ALLOC FI(C1EXMSG5) NEW DELETE DSORG(PS) RECFM(F,B) LRECL(133)
BLKSIZE(27930) SPACE(5,5) TRACKS UNIT(SYSALLDA)
-COMMAND=ALLOC FI(C1MSG51) NEW DELETE DSORG(PS) RECFM(F,B) LRECL(133)
BLKSIZE(27930) SPACE(5,5) TRACKS UNIT(SYSALLDA)

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

Figura 20. crastart.conf - CA Endevor® inicialização do SCM RAM usando CRASTART

## (Opcional) Customize CRANDVRA

Os métodos de inicialização de submissão em lote e CRASTART invocam o executável REXX CRANDVRA para alocar os conjuntos de dados específicos de usuários pelo CA Endevor® SCM RAM.



DD	Nome do conjunto de dados	Tipo
DEPEND	&SYSPREF..&SYSUID.. &SYSNAME..CRA\$NDVR.DEPEND	Permanente
BROWSE	&SYSPREF..&SYSUID.. &SYSNAME..CRA\$NDVR.BROWSE	Temporário
C1PRINT	&SYSPREF..&SYSUID.. &SYSNAME..CRA\$NDVR.LISTING	Temporário

Você pode customizar uma cópia deste executável REXX de alocação se certos padrões, como o nome do conjunto de dados, não corresponderem aos padrões do seu site. O CRANDVRA está localizado em FEK.SFEKPROC, a menos que você tenha usado um qualificador de alto nível diferente durante a instalação do SMP/E do Developer para System z.

Consulte a documentação dentro do CRANDVRA para obter instruções de customização.

**Nota:** É recomendável copiar a alocação REXX de amostra para um novo conjunto de dados e customizar esta cópia para evitar sobrescrevê-la ao realizar manutenção. Ao fazê-lo, você deve atualizar a referência para SFEKPROC no SYSEXEC DD do método de inicialização do CARMA da sua escolha para que ele corresponda ao nome do seu novo conjunto de dados.

## (Opcional) Customize o CA Endevor® SCM RAM

Os seguintes componentes do CARMA podem ser customizados, independentemente do método de inicialização escolhido. Os membros da amostra referenciados abaixo estão localizados em FEK.#CUST.PARMLIB, a menos que você tenha especificado um local diferente ao customizar e submeter a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes.

1. (Opcional) Customize FEK.#CUST.PARMLIB(CRASHOW). Consulte a documentação dentro do CRASHOW para instruções de customização. O CRASHOW define os filtros padrão para os ambientes, sistemas e assim por diante do CA Endevor®.
2. (Opcional) Customize FEK.#CUST.PARMLIB(CRATMAP). Consulte a documentação dentro do CRATMAP para obter instruções de customização. O CRATMAP substitui o tipo de CA Endevor® SCM por mapeamentos de extensão do arquivo.

## (Opcional) Suportando Múltiplos RAMs

O CARMA permite que múltiplos RAMs sejam definidos e possam ser executados concomitantemente. Porém, uma vez que há somente um servidor CARMA ativo por usuário, mesmo quando há múltiplos RAMs, algumas mudanças de configuração podem ser necessárias para fazer com que esta configuração funcione.

Os RAMs são definidos por um desenvolvedor RAM no conjunto de dados VSAM de configuração do CARMA CRADEF. Durante a inicialização, o servidor CARMA CRASERV identificará todos os RAMs definidos e apresentará as informações para o cliente RAM. O usuário poderá então selecionar um ou mais RAMs, que serão carregados no servidor CARMA.

Uma vez que os RAMs estão ativos como plug-ins do servidor CARMA, você deve assegurar que todos os pré-requisitos (como as alocações do conjunto de dados) para cada um dos RAMs estejam disponíveis no espaço de endereço do servidor

CARMA. Isto pode requerer mudanças nas amostras de configuração do CARMA, como CRASUBMT ou crastart.conf, que são enviados com o Developer para System z.

## Exemplo

No exemplo a seguir, você inicia a partir de uma configuração existente com o CA Endevor<sup>®</sup> SCM RAM, usando o método de inicialização CRASTART, e inclui o PDS RAM da amostra.

Definições para o CA Endevor<sup>®</sup> SCM RAM:

- FEK.SFEKVSM2(CRA0VCAD) - definições CRADEF
- FEK.SFEKVSM2(CRA0VCAS) - definições CRASTRS
- /etc/rdz/crastart.endevor.conf - arquivo de configuração CRASTART

Definições para o PDS RAM:

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

O processo inicia com um desenvolvedor RAM reunindo os dados e informações necessários para programador de sistema completar a configuração.

1. Extraia os dados específicos para o PDS RAM dos membros do SFEKVSM2 (estes membros mantêm definições para todos os RAMs de amostra, não somente os PDS RAM).
2. Funda estes dados com os membros do CA Endevor<sup>®</sup> SCM RAM SFEKVSM2.
3. Crie uma lista de pré-requisitos específicos de PDS RAM:
  - DD CRARAM1, vinculado ao FEK.#CUST.CRARAM1
  - Ambiente TSO

Depois o programador de sistema usa estes dados para criar conjuntos de dados VSAM do CARMA atualizados e usa as informações sobre os pré-requisitos para criar um arquivo de configuração CRASTART que é capaz de suportar os dois RAMs.

1. Use os dados combinados como entrada para as tarefas CRA\$VDEF e CRA\$VSTR para criar a configuração do CARMA atualizada e os conjuntos de dados VSAM de informações customizadas CRADEF e CRASTRS.
2. Inclua uma definição CRARAM1 para crastart.endevor.conf:

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

3. Verifique a instrução do PROGRAMA no crastart.endevor.conf para assegurar que ele seja capaz de fornecer o ambiente necessário para os dois RAMs:

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

- IKJEFT01: TSO, usado para permitir certas chamadas autorizadas em um ambiente não autorizado e usado como ambiente para executar o executável de pré-alocação SCM RAM do CA Endevor<sup>®</sup>.
- Executável de pré-alocação SCM RAM do %CRANDVRA: CA Endevor<sup>®</sup> (localizado em FEK.SFEKPROC), que aloca conjuntos de dados de trabalho específicos de usuário temporários (e permanentes).



- NDVRC1: backend do CA Endeavor<sup>®</sup> que possui uma construção no mecanismo para executar comandos TSO and ISPF.
- PGM(CRASERV): comando para iniciar um servidor CARMA, no formato de comando ISPF
- PARM(&CRAPRM1. &CRAPRM2.): parâmetros para CRASERV, no formato de comando ISPF. &CRAPRM1 é a porta para ser usada e &CRAPRM2 é o valor de tempo limite.

O SCM RAM do CA Endeavor<sup>®</sup> é ativo em um ambiente ISPF, o que implica que o ambiente TSO requerido pelo PDS RAM também está disponível.

---

## (Opcional) IRXJCL versus CRAXJCL

Se o servidor CARMA for iniciado utilizando TSO (IKJEFTxx), você pode ter problemas se os RAMs chamarem serviços que, por sua vez, chamam a interface em lote REXX IRXJCL. O problema poderá ocorrer quando os processadores chamados pela RAM tiverem sido executados anteriormente sem o TSO ou apenas no TSO on-line, e alocarem dinamicamente DD SYSTSIN ou SYSTSPRT. Um programa de amostra CRAXJCL é fornecido para oferecer uma solução alternativa para esse problema.

Seu processador poderá falhar se ele tentar alocar SYSTSIN ou SYSTSPRT (necessárias para IRXJCL) porque o TSO em lote (necessário para CARMA) já possui esses nomes de DD alocados e abertos. O módulo de substituição CRAXJCL tenta alocar SYSTSIN e SYSTSPRT para DUMMY, mas ignora os erros que ocorrerem se as alocações falharem.

Isso significa que quando seus processadores forem executados em um ambiente CARMA iniciado pelo TSO, as alocações para SYSTSIN e SYSTSPRT serão as mesmas que aquelas usadas pelo CARMA. Quando os processadores forem executados fora do TSO/CARMA, as alocações de SYSTSIN e SYSTSPRT serão criadas pelo CRAXJCL. Portanto, seus processadores não devem depender do conteúdo do conjunto de dados alocado para SYSTSIN.

Assume-se que as chamadas para IRXJCL usam o campo PARM para transmitir o nome do REXX e os parâmetros de inicialização, conforme documentado em *TSO/E REXX Reference* (SA22-7790). Isso significa que SYSTSIN pode ser utilizado com segurança pelo CARMA. Qualquer saída enviada para SYSTSPRT pelo IRXJCL terminará no log do CARMA.

Os processadores que chamam o módulo de substituição CRAXJCL não devem tentar alocar DD SYSTSIN ou SYSTSPRT antes de chamar CRAXJCL.

## Criar CRAXJCL

O módulo de substituição CRAXJCL é fornecido no formato de origem porque você precisará customizá-lo para especificar as alocações específicas que deseja utilizar para SYSTSPRT. SYSTSIN geralmente deve ser alocado para um conjunto de dados fictício.

O código de origem do assembler de amostra e uma tarefa de amostra de compilação/ligação são fornecidos como FEK.#CUST.ASM(CRAXJCL) e FEK.#CUST.JCL(CRA#CIRX) respectivamente, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP) . Consulte “Configuração da Customização” na página 15 para obter mais detalhes.

Customize o código de origem do assembler CRAXJCL de acordo com suas necessidades, utilizando a documentação no membro. Em seguida, customize e envie a JCL CRA#CIRX para criar o módulo de carregamento CRAXJCL. Consulte a documentação em CRA#CIRX para obter instruções de customização.

---

## Capítulo 4. (Opcional) Application Deployment Manager

O Developer para System z utiliza determinadas funções do Application Deployment Manager como uma abordagem de implementação comum para vários componentes. As etapas de customização listadas neste capítulo serão necessárias se seus desenvolvedores utilizarem qualquer uma das funções a seguir:

- Enterprise Service Tools (EST)
- BMS Screen Designer
- MFS Screen Designer
- Geração de Código CICSTS

**Nota:** O Enterprise Service Tools (EST) inclui diversas ferramentas, como o Modelador de Fluxo de Serviço (SFM) e os Serviços XML para a Empresa (XSE).

Customizar o Application Deployment Manager inclui o servidor CICS Resource Definition (CRD), que é executado como um aplicativo do CICS no z/OS para suportar as seguintes funções:

- Consultas de recursos do CICS
- Pedidos de instalação e de desinstalação da definição de recursos do CICS nos dois ambientes CICSplex SM e não-CICSplex SM
- Pedidos de phase-in do conjunto de mapas e programa
- Pedidos de varredura de pipeline
- Pedidos de exportação, importação e atualização de manifesto

Administradores do CICS podem localizar mais informações sobre o servidor CRD no Capítulo 15, “Considerações sobre o CICSTS”, na página 251.

---

### Requisitos e Lista de Verificação

Você precisará da assistência de um administrador do CICS, um administrador de TCP/IP e um administrador de segurança para concluir essa tarefa de customização, que exige os seguintes recursos ou tarefas de customização especiais:

- Porta TCP/IP para comunicação externa
- Atualização de JCL da região CICS
- Atualização de CSD da região CICS
- Definição de grupo para a região CICS
- Regra de segurança para permitir que administradores atualizem um VSAM do Application Deployment Manager
- Configuração de segurança do CICSTS
- (Opcional) Definição de nomes da transação CICS
- (Opcional) Regra de segurança para permitir que usuários atualizem um VSAM do Application Deployment Manager

Para iniciar o uso do Application Deployment Manager em seu site, é necessário executar as tarefas a seguir. A menos que especificado o contrário, todas as tarefas são obrigatórias.

1. Crie o repositório CRD. Para obter detalhes, consulte “Repositório do CRD”.
2. Escolha a interface do CICS interface (RESTful ou Serviço da Web) a ser usada. (As interfaces podem coexistir). Para obter detalhes, consulte “RESTful versus Serviços da Web” na página 73.
3. Se desejar, faça as customizações específicas de RESTful. Para obter detalhes, consulte “Servidor CRD Usando a Interface RESTful” na página 73.
  - Defina o servidor CRD como a região de conexão primária do CICS
  - Defina, opcionalmente, o servidor CRD para as regiões de conexão não-primárias do CICS.
  - Opcionalmente, customize os IDs de transação do servidor CRD.
4. Se desejar, faça as customizações específicas do Serviço da Web. Para obter detalhes, consulte “Servidor CRD Usando a Interface do Serviço da Web” na página 75.
  - Inclua o manipulador de mensagens do pipeline (possivelmente customizados) na concatenação RPL do CICS.
  - Defina o servidor CRD para a região de conexão primária do CICS.
  - Defina, opcionalmente, o servidor CRD para as regiões de conexão não-primárias do CICS.
5. Crie, opcionalmente, o repositório de manifesto. Para obter detalhes, consulte “(Opcional) Repositório de Manifesto” na página 77.

---

## Repositório do CRD

Customize e envie a tarefa ADNVCRD para alocar e inicializar o conjunto de dados VSAM do repositório CRD. Consulte a documentação no membro para obter instruções de customização.

ADNVCRD está localizado em FEK.#CUST.JCL, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes.

É aconselhável criar um repositório separado para cada região de conexão primária do CICS. O compartilhamento do repositório implica que todas as regiões CICS relacionadas utilizarão os mesmos valores armazenados no repositório.

### Nota:

- Um repositório do servidor CRD existente deve ser aumentado para ativar o suporte URIMAP incluído no Administrative Utility no Developer para System z versão 7.6.1. Consulte “Notas de Migração do Utilitário Administrativo” na página 259 para obter mais detalhes.
- A menos que seja notificado o contrário, o repositório do servidor CRD (contendo os valores customizados) pode ser reusado nos releases do Developer para System z.

Os usuários precisam de acesso READ ao repositório CRD e os administradores do CICS precisam de acesso UPDATE.

## CICS Administrative Utility

O Developer para System z fornece o administrative utility para permitir que administradores do CICS forneçam os valores padrão para as definições de recurso do CICS. Esses padrões podem ser somente de leitura ou podem ser editados pelo desenvolvedor de aplicativos.

O administrative utility é chamado pela tarefa de amostra ADNJSPAU. O uso desse utilitário requer acesso UPDATE ao repositório do CRD.

O ADNJSPAU está localizado em FEK.#CUST.JCL, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes.

Informações adicionais estão disponíveis em Capítulo 15, “Considerações sobre o CICSTS”, na página 251.

---

## RESTful versus Serviços da Web

O CICS Transaction Server fornece na versão 4.1 e superior suporte para uma interface HTTP projetada usando os princípios do Representational State Transfer (RESTful). Essa interface RESTful é agora a interface CICSTS estratégica para uso por aplicativos clientes. A interface de Serviço da Web mais antiga foi estabilizada e os aprimoramentos serão apenas para a interface RESTful.

O Application Deployment Manager segue essa instrução de direção e exige o servidor RESTful CRD para todos os serviços que são novos no Developer para System versão 7.6 ou superior.

As interfaces RESTful e de Serviço da Web podem ser ativadas simultaneamente em uma única região do CICS, se desejado. Nesse caso, haverá dois servidores CRD ativos na região. Os dois servidores compartilharão o mesmo repositório CRD. Observe que o CICS emitirá alguns avisos sobre definições duplicadas quando a segunda interface for definida para a região.

---

## Servidor CRD Usando a Interface RESTful

As informações desta seção descrevem como definir o servidor CRD que usa a interface RESTful para se comunicar com o cliente do Developer para System z.

As interfaces RESTful e de Serviço da Web podem ser ativadas simultaneamente em uma única região do CICS, se desejado. Nesse caso, haverá dois servidores CRD ativos na região. Os dois servidores compartilharão o mesmo repositório CRD. Observe que o CICS emitirá alguns avisos sobre definições duplicadas quando a segunda interface for definida para a região.

## Região de Conexão Primária do CICS

O servidor CRD deve ser definido para a região de conexão primária. Essa é a WOR (Web Owning Region) que processará os pedidos de Serviço da Web do Developer para System z.

- Coloque os módulos de carregamento FEK.SFEKLOAD(ADNCRD\*, ADNANAL e ADNREST) 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 ADNCSDRS para atualizar a CSD (CICS System Definition) para a região de conexão primária do CICS. Consulte a documentação no membro para obter instruções de customização.

ADNCSDRS está localizado em FEK.#CUST.JCL, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes.

- Use o comando CEDA apropriado para instalar o grupo do Application Deployment Manager para essa região, por exemplo:  
CEDA INSTALL GROUP(ADNPCRGP)

## Regiões de Conexão Não-primária do CICS

O servidor CRD também pode ser usado com uma ou mais regiões de conexão não-primária adicionais, que geralmente são Application Owning Regions (AOR).

**Nota:** Não é necessário executar estas etapas se o CICSplex SM Business Application Services (BAS) for usado para gerenciar as definições de recurso do CICS.

- Coloque o módulo de carregamento do Application Deployment Manager FEK.SFEKLOAD(ADNCRD\*) 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 ADNCSDAR para atualizar a CSD dessas regiões de conexão não-primária. Consulte a documentação no membro para obter instruções de customização.

ADNCSDAR está localizado em FEK.#CUST.JCL, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes.

- Use o comando CEDA apropriado para instalar o grupo do Application Deployment Manager para essas regiões, por exemplo:  
CEDA INSTALL GROUP(ADNARRGP)

## (Opcional) Customizar IDs de Transação do Servidor CRD

O Developer para System z fornece várias transações que são usadas pelo servidor CRD ao definir e consultar os recursos do CICS.

*Tabela 11. IDs de Transação do Servidor CRD Padrão*

Transação	Descrição
ADMS	Para pedidos da ferramenta Processamento de Manifesto para alterar os recursos do CICS. Geralmente, isso é destinado aos administradores do CICS.
ADMI	Para pedidos que definem, instalam ou desinstalam recursos do CICS.
ADMR	Para todos os outros pedidos que recuperam informações ambientais ou sobre recursos do CICS.

Você pode alterar os IDs da transação para que correspondam aos padrões do site seguinte estas etapas:

1. Customize e envie ADNTXNC para criar o módulo de carregamento ADNRCUST. Consulte a documentação no membro para obter instruções de customização.
2. Coloque o módulo de carregamento ADNRCUST resultante na concatenação RPL do CICS (instrução DD DFHRPL) das regiões do CICS em que o servidor CRD está definido.

3. Customize e envie ADNCSDTX para definir ADNRCUST como o programa para as regiões do CICS em que o servidor CRD está definido. Consulte a documentação no membro para obter instruções de customização.

**Nota:** O servidor CRD RESTful sempre tentará carregar o módulo de carregamento ADNRCUST. Portanto, você pode obter uma pequena melhora no desempenho criando e definindo o módulo de carregamento ADNRCUST, mesmo se não alterar os IDs da transação.

---

## Servidor CRD Usando a Interface do Serviço da Web

As informações desta seção descrevem como definir o servidor CRD que usa a interface do Serviço da Web para se comunicar com o cliente do Developer para System z.

As interfaces RESTful e de Serviço da Web podem ser ativadas simultaneamente em uma única região do CICS, se desejado. Nesse caso, haverá dois servidores CRD ativos na região. Os dois servidores compartilharão o mesmo repositório CRD. Observe que o CICS emitirá alguns avisos sobre definições duplicadas quando a segunda interface for definida para a região.

## Manipulador de Mensagens do Pipeline

O manipulador de mensagens do pipeline (ADNTMSGH) é usado para segurança através do processamento do ID do usuário e da senha no cabeçalho SOAP. ADNTMSGH é referido pelo arquivo de configuração de pipeline de amostra e, portanto, deve ser colocado na concatenação RPL do CICS. Consulte o Capítulo 15, “Considerações sobre o CICS”, na página 251 para saber mais sobre o manipulador de mensagens de pipeline e a configuração de segurança necessária.

O Developer para System z fornece várias transações que são usadas pelo servidor CRD durante a definição e a consulta de recursos do CICS. Esses IDs de transação são configurados pelo ADNTMSGH, dependendo da operação solicitada. O código de origem COBOL de amostra é fornecido para permitir customizações específicas do site para ADNTMSGH:

*Tabela 12. IDs de Transação do Servidor CRD Padrão*

Transação	Descrição
ADMS	Para pedidos da ferramenta Processamento de Manifesto para alterar recursos 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 recuperam as informações de ambiente e de recurso do CICS.

Utilizando o padrão:

- Coloque o módulo de carregamento FEK.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:



Os membros de amostra ADNMSGH\* estão localizados em FEK.#CUST.JCL e em FEK.#CUST.COBOL, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes.

- Customize o código de origem do Manipulador de Mensagens do Pipeline (COBOL) da amostra, FEK.#CUST.COBOL (ADNMSGHS), para que ele corresponda aos padrões do seu site’.
- Customize e submeta a tarefa FEK.#CUST.JCL (ADNMSGHC) para compilar a origem customizada ADNMSGHS. Consulte a documentação dentro do ADNMSGHC para obter instruções de customização. Observe que o módulo de carregamento resultante deve receber o nome de ADNTMSGH.
- 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 seja localizado antes de qualquer referência à FEK.SFEKLOAD, caso contrário, o padrão será usado.

## 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 do Developer para System z.

- Coloque os módulos de carregamento FEK.SFEKLOAD (ADNCRD\*, ADNANAL e ADNREST) na concatenação RPL do CICS (instrução DD DFHRPL) da região de conexão primária do CICS. É recomendável fazer isso incluindo o conjunto de dados de instalação na concatenação para que a manutenção aplicada esteja automaticamente disponível ao CICS. Observe que o módulo de carregamento do manipulador de mensagens do pipeline, ADNTMSGH, também deve ser colocado na concatenação RPL, conforme descrito em “Manipulador de Mensagens do Pipeline” na página 75.
- Customize e envie a tarefa ADNCSDWS para atualizar a CSD (CICS System Definition) para a região de conexão primária do CICS. Consulte a documentação no membro para obter instruções de customização. Observe que os IDs de transação usados nesta tarefa devem corresponder àqueles usados pelo Manipulador de Mensagens do Pipeline (que podem ter sido customizados).  
ADNCSDWS está localizado em FEK.#CUST.JCL, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes.
- Use o comando CEDA apropriado para instalar o grupo do Application Deployment Manager para essa região, por exemplo:  
CEDA INSTALL GROUP (ADNPCRGP)

## regiões de conexão não-primária do CICS

O servidor CRD também pode ser usado com uma ou mais regiões de conexão não-primária adicionais, que geralmente são Application Owning Regions (AOR).

**Nota:** Não será necessário executar estas etapas se o CICSplex SM Business Application Services (BAS) for usado para gerenciar as definições de recursos do CICS.

- Coloque os módulos de carregamento do Application Deployment Manager FEK.SFEKLOAD (ADNCRD\*) na concatenação RPL do CICS (instrução DD DFHRPL) dessas regiões de conexão não-primária. É recomendável fazer isso



incluindo o conjunto de dados de instalação na concatenação para que a manutenção aplicada esteja automaticamente disponível ao CICS.

- Customize e envie a tarefa ADNCSDAR para atualizar a CSD dessas regiões de conexão não-primária. Consulte a documentação no membro para obter instruções de customização.

ADNCSDAR está localizado em FEK.#CUST.JCL, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes.

- Use o comando CEDA apropriado para instalar o grupo do Application Deployment Manager para essas regiões, por exemplo:  
CEDA INSTALL GROUP(ADNARRGP)

---

## (Opcional) Repositório de Manifesto

O Developer para System z permite que clientes procurem e, opcionalmente, alterem manifestos descrevendo recursos do CICS selecionados. Dependendo das permissões configuradas pelo administrador do CICS, as alterações podem ser feitas diretamente ou exportadas para o repositório de manifesto para processamento adicional por um administrador do CICS.

### Nota:

- Esta etapa será necessária apenas para clientes que exportam manifestos do Developer para System z para serem processados pela ferramenta de Processamento de Manifesto.
- A ferramenta Processamento de Manifesto é um plug-in para o IBM CICS Explorer.

Customize e envie a tarefa ADNMFST para alocar e inicializar o conjunto de dados VSAM do repositório de manifesto e para defini-la na região de conexão primária do CICS. Consulte a documentação no membro para obter instruções de customização. Um repositório de manifesto separado deve ser criado para cada região de conexão primária do CICS. Todos os usuários precisam de acesso UPDATE ao repositório de manifesto.

ADNMFST está localizado em FEK.#CUST.JCL, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes.



---

## Capítulo 5. (Opcional) SCLM Developer Toolkit

O SCLM Developer Toolkit fornece as ferramentas necessárias para estender os recursos do SCLM para o cliente. O SCLM por si só é um gerenciador de código de origem baseado em host enviado como parte do ISPF.

O SCLM Developer Toolkit possui um plug-in baseado em Eclipse que faz a interface com o SCLM e fornece acesso a todos os processos do SCLM para o desenvolvimento de código legado, bem como suporte para o desenvolvimento completo do Java e J2EE na estação de trabalho com sincronização para SCLM no mainframe, incluindo construção, montagem e implementação do código J2EE a partir do mainframe.

---

### Requisitos e Lista de Verificação

Você precisará da assistência de um administrador de SCLM e, opcionalmente, de um administrador de segurança para concluir essa tarefa de customização, que exige os seguintes recursos e/ou tarefas de customização especiais:

- Atualizações de APF e LINKLIST
- Definir tradutores de idiomas SCLM para suporte JAVA/J2EE
- Definir tipos de SCLM para suporte JAVA/J2EE
- (Opcional) Regra de segurança para permitir que os usuários atualizem para um SCLM VSAM
- (opcional) Instalação de Ant

Para iniciar o uso do SCLM Developer Toolkit em seu site, é necessário executar as tarefas a seguir. A menos que especificado o contrário, todas as tarefas são obrigatórias.

1. Verifique e ajuste os pré-requisitos e as atualizações de PARMLIB. Para obter detalhes, consulte “Pré-requisitos” na página 80.
2. Customize os arquivos de configuração do Developer para System z. Para obter detalhes, consulte:
  - “Atualizações do ISPF.conf para SCLMDT” na página 80
  - “Atualizações do rsed.envvars para SCLMDT” na página 81
3. Defina, opcionalmente, o suporte à conversão de nomes longos/curtos. Para obter detalhes, consulte “(Opcional) Tradução de Nome Longo/Abreviado” na página 82.
4. Instale e customize, opcionalmente, o Ant para que utilize o suporte à construção de JAVA/J2EE. Para obter detalhes, consulte “(Opcional) Instalar e Customizar Ant” na página 84.
5. Atualize o SCLM para definir partes específicas do SCLMDT. Para obter detalhes, consulte “Atualizações SCLM para SCLMDT” na página 85.
6. Opcionalmente, configure a automação para limpar, periodicamente, a área de trabalho SCLMDT. Para obter detalhes, consulte “Remover Arquivos Antigos de WORKAREA” na página 86.

---

## Pré-requisitos

Consulte o Apêndice E, “Requisitos”, na página 329 para obter uma lista de manutenções necessárias do SCLM.

Esse apêndice documenta também as especificações Ant necessárias para construções JAVA/J2EE no SCLM Developer Toolkit.

**Atenção:** O SCLM Developer Toolkit requer o uso do ISPF's TSO/ISPF Client Gateway, o que implica que o z/OS 1.8 ou superior é necessário.

Conforme descrito em “Alterações PARMLIB” na página 16, o SCLM Developer Toolkit requer customização adicional de configurações do sistema. Essas alterações incluem:

- (BPXPRMxx) Aumentar o número de processos por ID do usuário do z/OS UNIX.
- (PROGxx) Autorizar por APF SYS1.LINKLIB e o tempo de execução REXX, REXX.V1R4M0.SEAGLPA ou REXX.V1R4M0.SEAGALT.
- (PROGxx/LPALSTxx) Coloque ISP.SISPLPA, ISP.SISPLOAD, SYS1.LINKLIB e o tempo de execução do REXX em LINKLIST/LPALIB.

Além disso, o SCLM Developer Toolkit utiliza o SDSF ou o comando do TSO **OUTPUT** para recuperar o status de conclusão da tarefa e a saída de tarefas. Ambos os métodos requerem atenção adicional:

- O SDSF deve ser ordenado, instalado e configurado separadamente. Também requer o uso do JES2.
- As configurações padrão do comando **OUTPUT** do TSO permitem que um usuário recupere a saída de tarefa que começa apenas com seu ID de usuário. Se você deseja usar o recurso **OUTPUT** completamente, poderá ser necessário modificar a saída do TSO/E de amostra IKJEFF53 para que um usuário possa recuperar a saída de tarefas que ele possui, mas que não começa com esse ID do usuário. Para obter informações adicionais sobre essa saída, consulte-o *TSO/E Customization* (SA22-7783).

Os usuários precisam de permissão READ, WRITE e EXECUTE para os diretórios z/OS UNIX /tmp/ e /var/rdz/WORKAREA/. O diretório WORKAREA/ está localizado em /var/rdz/, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes.

---

## Atualizações do ISPF.conf para SCLMDT

O SCLM Developer Toolkit usa as estruturas padrão do ISPF/SCLM, portanto, assegure-se de que a biblioteca de estruturas ISP.SISPSLIB esteja alocada na concatenação ISPSLIB no ISPF.conf. O uso do conjunto de dados ISP.SISPSENU é opcional.

O ISPF.conf está localizado em /etc/rdz/, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes. É possível editar o arquivo com o comando do TSO **OEDIT**.

**Nota:** As mudanças são efetivadas para todos os clientes que se conectam ao host após a atualização.

O código de amostra a seguir mostra o arquivo ISPF.conf, que deve ser customizado para corresponder ao seu ambiente do sistema. As linhas de comentário iniciam com um asterisco (\*). Inclua conjuntos de dados na concatenação na mesma linha e separe os nomes com uma vírgula (.). Consulte “ISPF.conf, Arquivo de Configuração do TSO/ISPF Client Gateway do ISPF” na página 47 para obter mais detalhes sobre como customizar o ISPF.conf.

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

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

*Figura 21. Atualizações do ISPF.conf para SCLMDT*

**Nota:**

- Você pode incluir suas próprias instruções equivalentes ao DD e as concatenações do conjunto de dados para customizar o ambiente TSO, assemelhando-se a um procedimento de logon do TSO. Consulte o Capítulo 16, “Customizando o Ambiente TSO”, na página 263 para obter mais detalhes.
- Quando você estiver fazendo construções em lote, assegure-se de que a versão customizada da estrutura FLMLIBS seja concatenada antes da biblioteca de estruturas ISPF/SCLM.

```
ispslib=h1q.USERSKEL,ISP.SISPSLIB
```

---

## Atualizações do rsed.envvars para SCLMDT

O SCLM Developer Toolkit utiliza algumas diretivas configuradas em rsed.envvars para localizar conjuntos de dados e diretórios.

O rsed.envvars está localizado em /etc/rdz/, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes. É possível editar o arquivo com o comando do TSO OEDIT.

**Nota:** A tarefa iniciada RSED deve ser reiniciada para assimilar as mudanças feitas.

A amostra de código a seguir mostra as diretivas SCLMDT em rsed.envvars, que deve ser customizado para corresponder ao seu ambiente do sistema. Consulte “Arquivo de Configuração do RSE rsed.envvars” na página 31 para obter mais detalhes sobre a customização de rsed.envvars.

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

Figura 22. Atualizações do *rsed.envvars* para SCLMDT

---

## (Opcional) Tradução de Nome Longo/Abreviado

O SCLM Developer Toolkit fornece a capacidade para armazenar arquivos com nomes longos (que são nomes de arquivos com mais de 8 caracteres ou compostos por letras maiúsculas e minúsculas) no SCLM. Isso pode ser feito usando um arquivo VSAM que contém o mapeamento do nome do arquivo longo para o nome do membro de 8 caracteres usado no SCLM.

### Nota:

- Para versões anteriores ao z/OS 1.8, esse recurso é fornecido através de uma PTF ISPF/SCLM de base que aborda APAR OA11426.
- A conversão de nomes longos/abreviados também é usada por outros produtos relacionados ao SCLM, como o IBM SCLM Administrator Toolkit.

## Criar LSTRANS.FILE, o VSAM de Conversão de nome Longo/Curto

Customize e envie o membro de amostra FLM02LST na biblioteca de amostra do ISPF ISP.SISPSAMP para criar o VSAM de conversão de nomes longos/abreviados. As etapas de configuração nesta publicação esperam que o VSAM seja denominado FEK.#CUST.LSTRANS.FILE, conforme mostrado na JCL de configuração de amostra a seguir.

```

//FLM02LST JOB <parâmetros da tarefa>
//*
/* CUIDADO: Isso não é um procedimento JCL nem uma tarefa completa.
/* Antes de usar 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 o VSAM.
/* 3. Altere todas as referências de FEK.#CUST.LSTRANS.FILE para
/*     corresponder à sua convenção de nomenclatura para o VSAM
de conversão do SCLM.
/*
//CREATE EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
DELETE FEK.#CUST.LSTRANS.FILE
SET MAXCC=0
DEFINE CLUSTER(NAME(FEK.#CUST.LSTRANS.FILE) -
               VOLUMES(***** ) -
               RECORDSIZE(58 2048) -
               SHAREOPTIONS(3 3) -
               CYLINDERS(1 1) -
               KEYS(8 0) -
               INDEXED) -
DATA (NAME(FEK.#CUST.LSTRANS.FILE.DATA)) -
INDEX (NAME(FEK.#CUST.LSTRANS.FILE.INDEX))

/* DEFINE ALTERNATE INDEX WITH NONUNIQUE KEYS -> ESDS */

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

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

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

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

```

Figura 23. FLM02LST - JCL de Configuração da Conversão de Nomes Longos/Curtos

**Nota:** Os usuários precisam da autoridade UPDATE para esse conjunto de dados VSAM, conforme descrito em Capítulo 10, “Considerações sobre Segurança”, na página 157.



## Atualizações de rsed.envvars para Conversão de Nomes Longos/Curtos

Antes de usar a conversão de nomes longos/curtos, remova o comentário e configure a variável de ambiente `rsed.envvars _SCLMDT_TRANTABLE` para que corresponda ao nome do VSAM de conversão de nomes longos/curtos.

O `rsed.envvars` está localizado em `/etc/rdz/`, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes. É possível editar o arquivo com o comando do TSO **OEDIT**.

**Nota:** A tarefa iniciada RSED deve ser reiniciada para assimilar as mudanças feitas.

---

### (Opcional) Instalar e Customizar Ant

Esta etapa será necessária apenas se você planejar utilizar o suporte à construção JAVA/J2EE no SCLM.

Apache Ant é uma ferramenta de construção Java de software livre e pode ser transferida por download de <http://ant.apache.org/>. O Ant consiste em arquivos de texto e scripts que são distribuídos no formato ASCII e, portanto, requerem que uma conversão ASCII/EBCDIC seja executada no z/OS UNIX.

Execute as seguintes etapas para implementar Ant no z/OS e para defini-lo para o Developer para System z:

- Faça o download, em formato binário, do arquivo Ant compactado mais recente no sistema de arquivo z/OS UNIX. É recomendável fazer download da versão .zip do ANT devido aos problemas que podem ser encontrados no z/OS durante a extração de versões de formato com sufixo tar.gz ou tar.bz2.
- Abra uma sessão de linha de comandos z/OS UNIX para continuar a instalação, por exemplo, com o comando **TSO OMVS**.
- Crie um diretório inicial para a instalação de Ant com o comando **mkdir -p /home-dir** e torne-o seu diretório atual com o comando **cd /home-dir**.
- Use o comando de extração JAR **jar -xf apache-ant-1.7.1.zip** para extrair o arquivo para o diretório atual. Um diretório bin Java deverá estar em seu PATH z/OS UNIX local para usar o comando **jar**. Caso contrário, qualifique o comando com o local bin Java (por exemplo, **/usr/lpp/java/J5.0/bin/jar -xf apache-ant-1.7.1.zip**).
- Converta todos os arquivos de texto do Ant para EBCDIC (opcionalmente customizando e) executando o script de amostra `/usr/lpp/rdz/samples/BWBTRANT`.

**Nota:** Execute esse script apenas uma vez. Várias execuções corromperão a instalação do Ant.

- Para verificar se a tradução foi bem-sucedida, localize e procure no arquivo de texto no diretório ANT, como `apache-ant-1.7.1/README`. Se o arquivo estiver legível, a conversão foi bem-sucedida.
- Use o comando **chmod -R 755 \*** para permitir que todos os usuários READ e EXECUTE arquivos no diretório ANT.
- Antes de usar o Ant, configure as variáveis de ambiente `rsed.envvars JAVA_HOME` e `ANT_HOME`.

- O JAVA\_HOME é necessário para apontar o diretório inicial do Java, por exemplo:  
JAVA\_HOME=/usr/lpp/java/IBM/J5.0
- ANT\_HOME deve apontar para o diretório home do Ant, por exemplo:  
ANT\_HOME=/usr/lpp/Apache/Ant/apache-ant-1.7.1

Por exemplo:

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

Para testar se a inicialização do Ant foi bem-sucedida:

- Inclua os diretórios bin do Ant e do Java na variável de ambiente PATH.

Exemplo:

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

- Execute **ant -version** para exibir a versão, se a instalação tiver sido bem-sucedida.

Exemplo:

```
ant -version
```

**Nota:** Configurar a instrução PATH assim é necessário apenas para teste, não para uso operacional.

## Atualizações SCLM para SCLMDT

O próprio SCLM também requer customização para funcionar com o SCLM Developer Toolkit. Consulte o *IBM Rational Developer para System z SCLM Developer Toolkit Administrator's Guide* (SC23-9801) para obter informações adicionais sobre as tarefas de customização necessárias:

- Definir tradutores de idiomas para suporte JAVA/J2EE
- Definir tipos de SCLM para suporte JAVA/J2EE

Para concluir a customização e as tarefas de definição de projetos, o administrador de SCLM precisa conhecer vários valores customizáveis do Developer para System z, conforme descrito em Tabela 13.

Tabela 13. Lista de Verificação do Administrador de SCLM

Descrição	<ul style="list-style-type: none"> <li>• Valor Padrão</li> <li>• Onde encontrar a resposta</li> </ul>	Valor
Biblioteca de amostra do Developer para System z	<ul style="list-style-type: none"> <li>• FEK.SFEKSAMV</li> <li>• Instalação SMP/E</li> </ul>	
Diretório de amostra do Developer para System z	<ul style="list-style-type: none"> <li>• /usr/lpp/rdz/samples</li> <li>• Instalação SMP/E</li> </ul>	

Tabela 13. Lista de Verificação do Administrador de SCLM (continuação)

Descrição	<ul style="list-style-type: none"> <li>• Valor Padrão</li> <li>• Onde encontrar a resposta</li> </ul>	Valor
Diretório bin Java	<ul style="list-style-type: none"> <li>• /usr/lpp/java/J5.0/bin</li> <li>• rsed.envvars - \$JAVA_HOME/bin</li> </ul>	
Diretório bin Ant	<ul style="list-style-type: none"> <li>• /usr/lpp/Apache/Ant/apache-ant-1.7.1/bin</li> <li>• rsed.envvars - \$ANT_HOME/bin</li> </ul>	
Diretório home WORKAREA	<ul style="list-style-type: none"> <li>• /var/rdz</li> <li>• rsed.envvars - \$_CMDSERV_CONF_HOME</li> </ul>	
Diretório home de configuração de projeto do SCLMDT	<ul style="list-style-type: none"> <li>• /var/rdz/sclmdt</li> <li>• rsed.envvars - \$_SCLMDT_CONF_HOME</li> </ul>	
VSAM de tradução de nomes longos/abreviados	<ul style="list-style-type: none"> <li>• FEK.#CUST.LSTRANS.FILE</li> <li>• rsed.envvars - \$_SCLMDT_TRANTABLE</li> </ul>	

## Remover Arquivos Antigos de WORKAREA

SCLM Developer Toolkit e TSO/ISPF Client Gateway do ISPF compartilham o mesmo WORKAREA, que pode precisar de limpeza periódica. Consulte “(Opcional) Limpeza de WORKAREA” na página 105 para obter informações adicionais sobre isso.

---

## Capítulo 6. (Opcional) Outras Tarefas de Customização

Esta seção combina uma variedade de tarefas opcionais de customização. Siga as instruções na seção apropriada para configurar o serviço desejado.

- “(Opcional) Procedimento Armazenado do DB2”
- “(Opcional) Suporte de Enterprise Service Tools (EST)” na página 89
- “(Opcional) Suporte de Idioma Bidirecional do CICS” na página 90
- “(Opcional) Mensagens de Erro IRZ de Diagnóstico” na página 91
- “(Opcional) Criptografia SSL do RSE” na página 91
- “(Opcional) Rastreamento de RSE” na página 94
- “(Opcional) Grupos de Propriedade Baseados em Host” na página 96
- “(Opcional) Projetos Baseados no Host” na página 97
- “(Opcional) Integração do File Manager” na página 98
- “(Opcional) Caracteres Não-editáveis” na página 99
- “(Opcional) Usando REXEC (ou SSH)” na página 100
- “(Opcional) Transação APPC para o Serviço de Comandos TSO” na página 102
- “(Opcional) Limpeza de WORKAREA” na página 105

---

### (Opcional) Procedimento Armazenado do DB2

Você precisará da assistência de um administrador de WLM e de um administrador de DB2 para concluir essa tarefa de customização, que exige os seguintes recursos ou tarefas de customização especiais:

- Atualização do WLM
- Novo membro PROCLIB
- Atualização do DB2

O Developer para System z fornece um procedimento armazenado do DB2 de amostra (Construtor de Procedimentos Armazenados PL/I e COBOL) com o propósito de construir Procedimentos Armazenados COBOL e PL/I a partir do cliente do Developer para System z.

**Nota:** Os membros de amostra ELAXM\* estão localizados em FEK.#CUST.JCL e FEK.#CUST.PROCLIB, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP (FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes.

### Alterações do Workload Manager (WLM)

Use 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 isso.

**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.

## Alterações do PROCLIB

Customize a tarefa de Procedimento Armazenado de amostra FEK.#CUST.PROCLIB(ELAXMSAM), conforme descrito no membro, e copie-a para SYS1.PROCLIB. Conforme mostrado na amostra de código a seguir, você deve fornecer o seguinte:

- O nome do ambiente de aplicativos definido no WLM para este Procedimento Armazenado
- O nome do subsistema do DB2
- O qualificador de alto nível de vários conjuntos de dados

```
//ELAXMSAM PROC RGN=0M,  
//          NUMTCB=1,  
//          APPLENV=#w1mwd4z,  
//          DB2SSN=#ssn,  
//          DB2PRFX='DSN810',  
//          COBPRFX='IGY.V3R4M0',  
//          PLIPRFX='IBMZ.V3R6M0',  
//          LIBPRFX='CEE',  
//          LODPRFX='FEK'  
//*  
//DSNX9WLM EXEC PGM=DSNX9WLM,REGION=&RGN,TIME=NOLIMIT,DYNAMNBR=10,  
//          PARM='&DB2SSN,&NUMTCB,&APPLENV'  
//STEPLIB DD DISP=SHR,DSN=&DB2PRFX..SDSNEXIT  
//          DD DISP=SHR,DSN=&DB2PRFX..SDSNLOAD  
//          DD DISP=SHR,DSN=&LIBPRFX..SCEERUN  
//          DD DISP=SHR,DSN=&COBPRFX..SIGYCOMP  
//          DD DISP=SHR,DSN=&PLIPRFX..SIBMZCMP  
//SYSEXEC DD DISP=SHR,DSN=&LODPRFX..SFEKPROC  
//SYSTSPRT DD SYSOUT=*  
//CEEDUMP DD SYSOUT=*  
//SYSABEND DD DUMMY  
//SYSUT1 DD UNIT=SYSALLDA,SPACE=(CYL,(1,1))  
//SYSUT2 DD UNIT=SYSALLDA,SPACE=(CYL,(1,1))  
//SYSUT3 DD UNIT=SYSALLDA,SPACE=(CYL,(1,1))  
//SYSUT4 DD UNIT=SYSALLDA,SPACE=(CYL,(1,1))  
//SYSUT5 DD UNIT=SYSALLDA,SPACE=(CYL,(1,1))  
//SYSUT6 DD UNIT=SYSALLDA,SPACE=(CYL,(1,1))  
//SYSUT7 DD UNIT=SYSALLDA,SPACE=(CYL,(1,1))  
//*
```

Figura 24. ELAXMSAM - Tarefa de Procedimento Armazenado do DB2

### Nota:

- O procedimento armazenado do DB2 utiliza REXX exec ELAXMREX, localizado em FEK.SFEKPROC. Não altere esta localização se desejar que seja possível que a manutenção SMP/E seja ativada automaticamente.
- Consulte o Capítulo 17, “Executando Várias Instâncias”, na página 271 se quiser renomear os membros ELAXMSAM ou ELAXMREX.

## Mudanças no DB2

Customize e envie o membro de amostra ELAXMJCL no conjunto de dados FEK.#CUST.JCL para definir o Procedimento Armazenado para o DB2. Consulte a documentação no membro para obter instruções de customização.

```

//ELAXMJCL JOB <parâmetros da tarefa>
//JOBPROC JCLLIB ORDER=(#hlq.SDSNPROC)
//JOBLIB DD DISP=SHR,DSN=#hlq.SDSNEXIT
// DD DISP=SHR,DSN=#hlq.SDSNLOAD
//*
//RUNTIAD EXEC PGM=IKJEFT01,DYNAMNBR=20
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
DSN S(#ssn) R(1) T(1)
RUN PROGRAM(DSNTIAD) PLAN(DSNTIAD) -
LIB('#hlq.RUNLIB.LOAD')
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
CREATE PROCEDURE SYSPROC.ELAXMREX
( IN FUNCTION_REQUEST VARCHAR(20) CCSID EBCDIC
, IN SQL_ROUTINE_NAME VARCHAR(27) CCSID EBCDIC
, IN SQL_ROUTINE_SOURCE VARCHAR(32672) CCSID EBCDIC
, IN BIND_OPTIONS VARCHAR(1024) CCSID EBCDIC
, IN COMPILE_OPTIONS VARCHAR(255) CCSID EBCDIC
, IN PRECOMPILE_OPTIONS VARCHAR(255) CCSID EBCDIC
, IN PRELINK_OPTIONS VARCHAR(32672) CCSID EBCDIC
, IN LINK_OPTIONS VARCHAR(255) CCSID EBCDIC
, IN ALTER_STATEMENT VARCHAR(32672) CCSID EBCDIC
, IN SOURCE_DATASETNAME VARCHAR(80) CCSID EBCDIC
, IN BUILDOWNER VARCHAR(8) CCSID EBCDIC
, IN BUILDUTILITY VARCHAR(18) CCSID EBCDIC
, OUT RETURN_VALUE VARCHAR(255) CCSID EBCDIC )
PARAMETER STYLE GENERAL RESULT SETS 1
LANGUAGE REXX EXTERNAL NAME ELAXMREX
COLLID DSNREXCS WLM ENVIRONMENT ELAXMSAM
PROGRAM TYPE MAIN MODIFIES SQL DATA
STAY RESIDENT NO COMMIT ON RETURN NO
ASUTIME NO LIMIT SECURITY USER;

COMMENT ON PROCEDURE SYSPROC.ELAXMREX IS
'PLI & COBOL PROCEDURE PROCESSOR (ELAXMREX), INTERFACE LEVEL 0.01';

GRANT EXECUTE ON PROCEDURE SYSPROC.ELAXMREX TO PUBLIC;
//*
```

Figura 25. ELAXMJCL – Definição do procedimento armazenado do DB2

**Nota:** 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) Suporte de Enterprise Service Tools (EST)

Esta tarefa de customização não requer assistência, recursos especiais ou tarefas de customização especiais.

---

O cliente do Developer para System z possui um componente de geração de códigos chamado Enterprise Service Tools (EST). Dependendo do tipo de código que está sendo gerado, esse código baseia-se nas funções fornecidas pela instalação do host do Developer para System z. Como tornar essas funções disponíveis está descrito nas seguintes seções:

- Capítulo 4, “(Opcional) Application Deployment Manager”, na página 71
- “(Opcional) Suporte de Idioma Bidirecional do CICS” na página 90
- “(Opcional) Mensagens de Erro IRZ de Diagnóstico” na página 91

**Nota:** O Enterprise Service Tools (EST) inclui diversas ferramentas, como o Modelador de Fluxo de Serviço (SFM) e os Serviços XML para a Empresa (XSE).

---

## (Opcional) Suporte de Idioma Bidirecional do CICS

Você precisará da assistência de um administrador do CICS para concluir essa tarefa de customização, que exige os seguintes recursos ou tarefas de customização especiais:

- Atualização de JCL da região do CICS
  - Definir um programa para o CICS
- 

O componente EST (Enterprise Service Tools) do Developer para 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.

Além disso, o código gerado por EST pode suportar transformação bidi em ambientes diferentes do CICS SFR (Service Flow Runtime). Um exemplo são os aplicativos em lote. Você pode criar os geradores de EST para incluir chamadas nas rotinas de conversão bidirecional, especificando as opções de transformação bidirecional apropriadas nos assistentes de geração EST e vinculando os programas gerados com a biblioteca de conversão bidirecional apropriada, FEK.SFEKLOAD.

Execute as seguintes tarefas para ativar o suporte à linguagem bidirecional do CICS:

1. Coloque os módulos de carregamento FEJBDCMP e FEJBDTRX do FEK.SFEKLOAD na concatenação RPL do CICS (instrução DD DFHRPL). É recomendável fazer isso incluindo o conjunto de dados de instalação na concatenação para que a manutenção aplicada esteja automaticamente disponível ao CICS.

**Nota:** Se você não concatenar o conjunto de dados de instalação, mas copiar os módulos para um conjunto de dados novo ou existente, lembre-se de que esse módulos são DLLs e DEVEM residir em uma biblioteca PDSE.

2. Definir FEJBDCMP e FEJBDTRX como programas para o CICS usando o comando CEDA apropriado, por exemplo:

```
CEDA DEF PROG(FEJBDCMP) LANG(LE) G(xxx)
CEDA DEF PROG(FEJBDTRX) LANG(LE) G(xxx)
```



---

## (Opcional) Mensagens de Erro IRZ de Diagnóstico

Essa tarefa de customização não exige assistência, mas exige os seguintes recursos ou tarefas de customização especiais:

- Atualização de LINKLIST
  - Atualização de JCL da região CICS
- 

O cliente do Developer para System z possui um componente de geração de códigos chamado Enterprise Service Tools (EST). Para que o código gerado por EST emita mensagens de erro de diagnóstico, todos os módulos IRZ\* e IIRZ\* da biblioteca de carregamento FEK.SFEKLOAD devem ser disponibilizados por meio do código gerado. EST pode gerar códigos para os seguintes ambientes:

- CICS
- IMS
- lote MVS

Quando o código gerado for executado em uma transação do CICS, então inclua todos os módulos IRZ\* e IIRZ\* no FEK.SFEKLOAD para o DFHRPL DD da região CICS. É recomendável fazer isso incluindo o conjunto de dados de instalação na concatenação para que a manutenção aplicada esteja automaticamente disponível ao CICS.

Para todas as outras situações, torne todos os módulos IRZ\* e IIRZ\* no FEK.SFEKLOAD disponíveis por meio de STEPLIB ou LINKLIST. É recomendável fazer isso incluindo o conjunto de dados de instalação na concatenação para que a manutenção aplicada esteja automaticamente disponível ao CICS.

Se você decidir usar o STEPLIB, deverá definir os módulos não disponíveis por meio do LINKLIST na diretiva STEPLIB da tarefa que executa o código.

Se os módulos de carregamento não estiverem disponíveis e um erro for encontrado pelo código gerado, a seguinte mensagem será emitida:

IRZ9999S Falha ao recuperar o texto de uma mensagem de tempo de execução do Ambiente de Linguagem.

Verifique se o módulo de mensagem de tempo de execução do Ambiente de Linguagem do recurso IRZ está instalado em DFHRPL ou em STEPLIB.

---

## (Opcional) Criptografia SSL do RSE

Você precisará da assistência de um administrador de segurança para concluir essa tarefa de customização, que exige os seguintes recursos ou tarefas de customização especiais:

- Atualização de LINKLIST
  - Regra de segurança para incluir conjuntos de dados controlados pelo programa
  - (Opcional) Regra de segurança para incluir certificado para SSL
- 

A comunicação externa (cliente-host) pode ser criptografada utilizando-se SSL (Secure Socket Layer). Esse recurso é desativado por padrão e é controlado pelas configurações no `ssl.properties`.

O `ssl.properties` está localizado em `/etc/rdz/`, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15

para obter mais detalhes. É possível editar o arquivo com o comando do TSO **OEDIT**. Observe que o RSE deve ser reiniciado para que as alterações sejam efetivadas.

O cliente se comunica com o daemon RSE durante a configuração de conexão e com o servidor RSE durante a sessão real. Os dois fluxos de dados são criptografados quando o SSL é ativado.

O daemon RSE e o servidor RSE suportam diferentes mecanismos para armazenar certificados devido a diferenças de arquitetura entre eles. Isso implica que as definições SSL são necessárias para o daemon RSE e para o servidor RSE. Um certificado compartilhado poderá ser usado se o daemon RSE e o servidor RSE usarem o mesmo método de gerenciamento de certificado.

*Tabela 14. Mecanismos de armazenamento de certificado SSL*

Armazenamento de certificado	Criado e gerenciado por	Daemon RSE	servidor RSE
conjunto de chaves	Produto de segurança compatível com SAF	suportados	suportados
banco de dados de chaves	gskkyman do z/OS UNIX	suportados	/
keystore	keytool Java	/	suportados

**Nota:**

- Conjuntos de chaves compatíveis com SAF é o método preferido para gerenciar certificados.
- Os conjuntos de chaves compatíveis com SAF podem armazenar a chave privada do certificado no banco de dados de segurança ou usando ICSF, a interface para o hardware de criptografia do System z. O acesso ao ICSF é protegido por perfis da classe de segurança CSFSERV.

O daemon RSE utiliza funções SSL do Sistema para gerenciar o SSL. Isso implica que SYS1.SIEALNKE deve ser controlado pelo programa pelo software de segurança e estar disponível para o RSE através de LINKLIST ou da diretiva STEPLIB em rsed.envvars.

A amostra de código a seguir mostra o arquivo `ssl.properties` de amostra, que deve ser customizado para corresponder a seu ambiente do sistema. Linhas de comentários são iniciadas com um sinal de sustenido (#) ao utilizar uma página de códigos US. As linhas de dados só podem ter uma diretiva e seu valor designado; não são permitidos comentários na mesma linha. Continuações de linha não são suportadas.

```
# ssl.properties – arquivo de configuração SSL
enable_ssl=false

# Daemon Properties

#daemon_keydb_file=
#daemon_keydb_password=
#daemon_key_label=

# Server Properties

#server_keystore_file=
#server_keystore_password=
#server_keystore_label=
#server_keystore_type=JCECERCFKS
```

*Figura 26. ssl.properties – Arquivo de configuração SSL*

As propriedades do daemon e do servidor precisarão ser configuradas somente se você ativar o SSL. Consulte o Apêndice A, “Configurando o SSL e a Autenticação X.509”, na página 293 para obter informações adicionais sobre a configuração do SSL.

#### **enable\_ssl**

Ative ou desative a comunicação SSL. O padrão é false. As únicas opções válidas são true e false.

#### **daemon\_keydb\_file**

Nome do anel de chaves RACF (ou produto de segurança semelhante). Forneça o nome do banco de dados de chaves se você utilizou **gskkyman** para criar um banco de dados de chaves em vez de utilizar um anel de chaves. Remova o comentário e customize essa diretiva se o SSL estiver ativado.

#### **daemon\_keydb\_password**

Deixe a linha comentada ou em branco se você usar um conjunto de chaves, caso contrário, forneça a senha do banco de dados de chaves. Remova o comentário e customize essa diretiva se o SSL estiver ativado e você estiver usando um banco de dados de chaves **gskkyman**.

#### **daemon\_key\_label**

O rótulo do certificado usado no anel de chaves ou banco de dados de chaves, se não estiver definido como padrão. O comentário deve ser removido, se o padrão for usado. Remova o comentário e customize essa diretiva se o SSL estiver ativado e você não estiver usando o certificado de segurança padrão.

#### **server\_keystore\_file**

Nome do keystore criado pelo comando Java’s **keytool**, ou o nome do conjunto de chaves RACF (ou produto de segurança semelhante) se **server\_keystore\_type=JCECERCFKS**. Remova o comentário e customize essa diretiva se o SSL estiver ativado.

#### **server\_keystore\_password**

Deixe a linha comentada ou em branco se você utilizar um conjunto de chaves, caso contrário, forneça a senha do keystore. Remova o comentário e customize essa diretiva se o SSL estiver ativado e você estiver utilizando um keystore **keytool**.

#### **server\_keystore\_label**

O rótulo do certificado usado no conjunto de chaves ou keystore. O padrão

é o primeiro certificado válido encontrado. Remova o comentário e customize essa diretiva se o SSL estiver ativado e você não estiver usando o certificado de segurança padrão.

#### **server\_keystore\_type**

Tipo de keystore. O padrão é JKS. Valores válidos são:

*Tabela 15. Tipos de keystore válidos*

Palavra-chave	Tipo de keystore
JKS	Keystore Java
JCERACFKS	Conjunto de chaves compatível com SAF, em que a chave privada do certificado é armazenada no banco de dados de segurança.
JCECCARACFKS	O conjunto de chaves compatível com SAF, em que a chave privada do certificado é armazenada usando ICSF, a interface para o hardware criptográfico do System z.

**Nota:** No momento da publicação, o IBM z/OS Java exige uma atualização do arquivo `/usr/lpp/java/J5.0/lib/security/java.security` para suportar JCECCARACFKS. A seguinte linha deve ser incluída:

```
security.provider.1=com.ibm.crypto.hwrfCCA.provider.IBMJCECCA
```

O arquivo resultante será semelhante a este:

```
security.provider.1=com.ibm.crypto.hwrfCCA.provider.IBMJCECCA
security.provider.2=com.ibm.jsse2.IBMJSSEProvider2
security.provider.3=com.ibm.crypto.provider.IBMJCE
security.provider.4=com.ibm.security.jgss.IBMJGSSProvider
security.provider.5=com.ibm.security.cert.IBMCertPath
security.provider.6=com.ibm.security.sasl.IBMSASL
```

## **(Opcional) Rastreo de RSE**

Esta tarefa de customização não requer assistência, recursos especiais ou tarefas de customização especiais.

O Developer para System z suporta diferentes níveis de rastreo do fluxo do programa interno para propósitos de resolução de problemas. O RSE e alguns dos serviços chamados pelo RSE usam as configurações no `rsecomm.properties` para conhecer o nível desejado de detalhes nos logs de saída.

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

O `rsecomm.properties` está localizado em `/etc/rdz/`, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes. É possível editar o arquivo com o comando do TSO **OEDIT**.

A amostra de código a seguir mostra o arquivo `rsecomm.properties`, que pode ser customizado para corresponder às suas necessidades de rastreo. Linhas de comentários são iniciadas com um sinal de sustenido (#) ao utilizar uma página de

códigos US. As linhas de dados só podem ter uma diretiva e seu valor designado; não são permitidos comentários na mesma linha. Continuações de linha não são suportadas.

```
# 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
debug_level=1

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

Figura 27. *rsecomm.properties* – Arquivo de configuração da criação de log

#### **server.version**

Versão do servidor de criação de log. O padrão é 5.0.0. Não modifique.

#### **debug\_level**

Nível de detalhe para logs de saída. O padrão é 1 (registrar mensagens de erro e de aviso). Observe que `debug_level` controla o nível de detalhe de vários serviços (e, portanto, vários arquivos de saída). O aumento do nível de detalhe causará diminuição no desempenho e deverá ser realizado somente sob a orientação do IBM Support Center. Consulte “Rastreo do RSE” na página 143 para obter informações adicionais sobre quais logs são controlados por essa diretiva.

Os valores válidos são os seguintes:

0	Registrar apenas mensagens de erro.
1	Registrar mensagens de erro e de aviso.
2	Mensagens de erro de log, de aviso e informativa.

**Nota:** `debug_level` pode ser alterado dinamicamente com o comando do operador **modify rsecommlog**, conforme descrito no Capítulo 8, “Comandos do Operador”, na página 123.

#### **log\_location**

Meio de saída para a criação de log relacionada ao RSE. O padrão é `Log_To_File`. Não altere ao usar o método de conexão do daemon RSE (padrão), a menos que direcionado pelo centro de suporte IBM.

Os valores válidos são os seguintes:

<code>Log_To_File</code>	<p>Enviar mensagens de log para um arquivo separado no diretório de saída do log.</p> <ul style="list-style-type: none"> <li>• Daemon RSE: <code>rsedaemon.log</code> em <code>daemonlog</code></li> <li>• Conjuntos de encadeamento RSE: <code>rserver.log</code> em <code>daemonlog</code></li> <li>• Usuário: <code>rsecomm.log</code> em <code>userlog/.eclipse/RSE/\$LOGNAME</code></li> </ul>
--------------------------	---

Log_To_StdOut	<p>Enviar mensagens de log para stdout.</p> <ul style="list-style-type: none"> <li>• Daemon RSE: roteado novamente para DD STDOUT na tarefa iniciada RSED</li> <li>• Conjuntos de encadeamentos do RSE: indefinido</li> <li>• Usuário: roteado novamente para stdout.log em userlog/.eclipse/RSE/\$LOGNAME</li> </ul>
---------------	---

daemonlog é o valor da diretiva daemon.log em rsed.envvars. Se a diretiva daemon.log for comentada ou não estiver presente, o caminho inicial do ID de usuário designado à tarefa iniciada RSED será usado. O caminho inicial é definido no segmento de segurança OMVS do ID de usuário.

Os logs específicos do usuário vão para userlog/.eclipse/RSE/\$LOGNAME, em que userlog é o valor da diretiva user.log em rsed.envvars, e \$LOGNAME é o ID do usuário de logon (em maiúsculas). Se a diretiva user.log for comentada ou não estiver presente, o caminho inicial do usuário será usado. O caminho inicial é definido no segmento de segurança OMVS do ID de usuário.

---

## (Opcional) Grupos de Propriedade Baseados em Host

Esta tarefa de customização não requer assistência, recursos especiais ou tarefas de customização especiais.

---

Os clientes do Developer para System z podem definir grupos de propriedades que contêm valores padrão para várias propriedades (por exemplo, as opções do compilador COBOL a serem usadas durante a compilação do código de origem COBOL). O Developer para System z possui alguns valores padrão integrados, mas permite também a definição de padrões customizados, específicos do sistema.

O local dos arquivos de configuração do valor padrão e do grupo de propriedades customizadas é definido em propertiescfg.properties, que está localizado em /etc/rdz/, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes. É possível editar o arquivo com o comando do TSO **OEDIT**. Observe que o RSE deve ser reiniciado para que as alterações sejam efetivadas.

A amostra de código a seguir mostra o arquivo propertiescfg.properties, que deve ser customizado para corresponder ao seu ambiente do sistema. Linhas de comentários são iniciadas com um sinal de sustenido (#) ao utilizar uma página de códigos US. As linhas de dados só podem ter uma diretiva e seu valor designado. Não são permitidos comentários na mesma linha. Continuações de linha não são suportadas.

```
#
# grupos de propriedades baseados no host - arquivo de configuração raiz
#
ENABLED=FALSE
RDZ-VERSION=7.5.0.0
PROPERTY-GROUP=/var/rdz/properties
DEFAULT-VALUES=/var/rdz/properties
```

*Figura 28. propertiescfg.properties - Arquivo de configuração de grupos de propriedades baseado no host*

### ATIVADO

Indica se o Developer para System z utilizará os arquivos de configuração do valor padrão e do grupo de propriedades. O padrão é FALSE. As únicas opções válidas são TRUE e FALSE.

### RDZ-VERSION

Nível mínimo de cliente do Developer para System z para usar grupos de propriedades baseadas no host. O padrão é 7.5.0.0. Não modifique.

### PROPERTY-GROUP

O local do arquivo de configuração do grupo de propriedades. O padrão é /var/rdz/properties.

### DEFAULT-VALUES

O local do arquivo de configuração do valor padrão. O padrão é /var/rdz/properties.

Consulte o Centro de Informações do Developer para System z (<http://publib.boulder.ibm.com/infocenter/ratdevz/v7r6/index.jsp>) para obter informações adicionais sobre a criação do arquivo de configuração do grupo de propriedades (propertygroups.xml) e o arquivo de configuração do valor-padrão (defaultvalues.xml).

---

## (Opcional) Projetos Baseados no Host

Esta tarefa de customização não requer assistência, recursos especiais ou tarefas de customização especiais.

---

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 que suas estruturas, membros e propriedades não podem ser modificados pelo cliente e são acessíveis apenas quando conectados ao host.

O local das definições de projeto é definido em projectcfg.properties, que está localizado em /etc/rdz/, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte "Configuração da Customização" na página 15 para obter mais detalhes. É possível editar o arquivo com o comando do TSO **OEDIT**. Observe que o RSE deve ser reiniciado para que as alterações sejam efetivadas.

A amostra de código a seguir mostra o arquivo projectcfg.properties, que deve ser customizado para corresponder ao seu ambiente do sistema. Linhas de comentários são iniciadas com um sinal de sustenido (#) ao utilizar uma página de códigos US. As linhas de dados só podem ter uma diretiva e seu valor designado. Não são permitidos comentários na mesma linha. Continuações de linha não são suportadas.



```
#
# 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/rdz/projects
```

Figura 29. *projectcfg.properties* – Arquivo de configuração de projetos baseados no host

#### WSED-VERSION

Nível mínimo de cliente do Developer para System z para usar projetos baseados no host. O padrão é 7.0.0.0. Não modifique.

#### PROJECT-HOME

O diretório base para as definições de projeto. O padrão é /var/rdz/projects.

**Nota:** Par ativar projetos baseados no host, um arquivo *project.instance* deve existir em /var/rdz/projects/USERID, em que /var/rdz/projects é o local dos arquivos de definição de projeto e USERID é o ID do usuário (em maiúsculas) com o qual o desenvolvedor efetua login.

Consulte o Centro de Informações do Developer para System z (<http://publib.boulder.ibm.com/infocenter/ratdevz/v7r6/index.jsp>) para obter informações adicionais sobre projetos baseados no host.

---

## (Opcional) Integração do File Manager

Você precisará da assistência de um administrador de segurança para concluir essa tarefa de customização, que exige os seguintes recursos ou tarefas de customização especiais:

- Regra de segurança para incluir conjuntos de dados controlados pelo programa
- 

O Developer para 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 utilitários já familiares de procura, edição, cópia e impressão localizados no ISPF, aprimorados para atender às necessidades dos desenvolvedores de aplicativos. Na versão atual do Developer para System z, apenas a procura e a edição dos conjuntos de dados MVS (incluindo todos os tipos de VSAM), criando e editando modelos de conjuntos de dados MVS (incluindo modelos dinâmicos) e os utilitários de cópia avançados são suportados.

Observe que o produto IBM File Manager para z/OS deve ser solicitado, instalado e configurado separadamente. Consulte o *Rational Developer para System z Prerequisites* (SC23-7659) para saber qual o nível do File Manager é necessário em sua versão do Developer para System z. A instalação e customização deste produto não está descrita neste manual.

Observe que o Developer para System z e o File Manager não suportam mais a interface em lote para acessar serviços do File Manager. O uso do listener do File Manager é necessário agora.

**Nota:** Além das tarefas de configuração normais do listener, descritas na documentação do File Manager, o Developer para System z exige que os conjuntos de dados STEPLIB do servidor sejam controlados pelo programa.

As definições da Integração do File Manager necessárias para o Developer para System z estão armazenadas em `FMIEXT.properties`, que está localizado em `/etc/rdz/`, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa `FEK.SFEKSAMP(FEKSETUP)`. Consulte “Configuração da Customização” na página 15 para obter mais detalhes. É possível editar o arquivo com o comando do TSO **OEDIT**. Observe que o RSE deve ser reiniciado para que as alterações sejam efetivadas.

A amostra de código a seguir mostra o arquivo `FMIEXT.properties`, que deve ser customizado para corresponder ao seu ambiente do sistema. Linhas de comentários são iniciadas com um sinal de sustenido (#) ao utilizar uma página de códigos US. As linhas de dados só podem ter uma diretiva e seu valor designado. Não são permitidos comentários na mesma linha. Continuações de linha não são suportadas.

```
# Propriedades de Extensão do FMI (File Manager Integration)
#
enabled=false
fmlistenport=1960
```

*Figura 30. FMIEXT.properties – Arquivo de configuração do File Manager*

**ativado**

Indica se o listener do File Manager está disponível no mesmo sistema host ou não. O valor padrão é `false`. Os únicos valores permitidos são `true` e `false`.

**fmlistenport**

Porta usada pelo listener do File Manager. O padrão é 1960. A comunicação nesta porta está confinada para a máquina host.

---

## (Opcional) Caracteres Não-editáveis

Esta tarefa de customização não requer assistência, recursos especiais ou tarefas de customização especiais.

---

Alguns caracteres não são convertidos adequadamente entre as páginas de códigos do host (baseadas em EBCDIC) e as páginas de códigos do cliente (baseadas em ASCII). O editor do cliente Developer para System z utiliza as definições no arquivo `uchars.settings` para identificar esses caracteres não editáveis. Ao abrir um conjunto de dados com um caractere identificado em `uchars.settings`, o editor impingirá o modo somente leitura para evitar que o conjunto de dados seja corrompido quando ele for salvo.

O `uchars.settings` está localizado em `/etc/rdz/`, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa `FEK.SFEKSAMP(FEKSETUP)`. Consulte “Configuração da Customização” na página 15 para obter mais detalhes. É possível editar o arquivo com o comando do TSO **OEDIT**. Observe que o RSE deve ser reiniciado para que as alterações sejam efetivadas. Observe também que é recomendado não alterar esse arquivo, a menos que você seja orientado pelo centro de suporte da IBM.

```
# uchars.settings - pontos de código não editáveis
#
*          *          0D 15 25;

# DBCS (japonês, coreano e chinês)
IBM-930   *          0D 15 1E 1F 25;
IBM-933   *          0D 15 1E 1F 25;
IBM-935   *          0D 15 1E 1F 25;
IBM-937   *          0D 15 1E 1F 25;
IBM-939   *          0D 15 1E 1F 25;
IBM-1390  *          0D 15 1E 1F 25;
IBM-1399  *          0D 15 1E 1F 25;
IBM-1364  *          0D 15 1E 1F 25;
IBM-1371  *          0D 15 1E 1F 25;
IBM-1388  *          0D 15 1E 1F 25;

# UNICODE
UTF-8     *          0D 0A;
UTF-16BE  *          0D 0A;
UTF-16LE  *          0D 0A;
UTF-16    *          0D 0A;
```

*Figura 31. uchars.settings - Arquivo de configuração de caracteres não editáveis*

O arquivo consiste de várias entradas no seguinte formato:

```
HOST-CODEPAGE LOCAL-CODEPAGE HEX-CODEPOINTS ;
```

Em que HEX-CODEPOINTS é uma lista delimitada por espaços em branco de pontos de código hexadecimais de 2 dígitos que identificam os caracteres não editáveis. A lista deve terminar com um ponto e vírgula (;).

As regras de sintaxe a seguir são aplicadas:

- Linhas de comentários são iniciadas com um sinal de sustenido (#) ao utilizar uma página de códigos US.
- As linhas de dados só podem ter dados; não são permitidos comentários na mesma linha.
- Um asterisco (\*) pode ser usado para HOST-CODEPAGE e/ou LOCAL-CODEPAGE. Ele atua como um curinga e representa todas as páginas de código.
- Entradas específicas têm precedência sobre as entradas genéricas (curinga).
- "host-cp \*" tem precedência se "host-cp \*" e "\*" local-cp" forem especificados e não forem substituídos por "host-cp local-cp".
- Se o mesmo par de páginas de códigos for especificado mais de uma vez, a última entrada será usada.

---

## (Opcional) Usando REXEC (ou SSH)

Esta tarefa de customização não requer assistência, recursos especiais ou tarefas de customização especiais.

---

REXEC (Execução Remota) é um serviço TCP/IP para permitir que clientes executem um comando no host. O SSH (Secure Shell) é um serviço semelhante, mas aqui todas as comunicações são criptografadas utilizando o SSL (Secure Socket Layer). O Developer para System z usa qualquer serviço para executar ações remotas (baseadas em host) em subprojetos z/OS UNIX.

O Developer para System z também pode ser configurado para usar o REXEC (ou SSH) para iniciar um servidor RSE no host. Observe, no entanto, que cada conexão iniciada dessa forma resultará em um servidor RSE separado, cada um utilizando uma quantidade razoável de recursos do sistema. Portanto, esse método de conexão alternativo é viável apenas para um pequeno número de conexões.

Além disso, como o método de conexão alternativa do REXEC (ou SSH) ignora o daemon RSE, ele não terá acesso a todos os serviços do host descritos nesta publicação, como processamento e auditoria de servidor único. Entre em contato com o suporte da IBM para saber se um serviço de host específico é suportado pelo método de conexão alternativo REXEC.

**Nota:** O Developer para System z utiliza a versão de REXEC do z/OS UNIX, e não a versão do TSO.

## Ações Remotas (Baseadas em Host) para Subprojetos z/OS UNIX

As ações remotas (baseadas em host) para subprojetos z/OS UNIX exigem que REXEC ou SSH estejam ativos no host. Se o REXEC/SSH não estiver configurado para utilizar a porta padrão, o cliente Developer para System z deve definir a porta correta a ser usada por subprojetos do z/OS UNIX. Isso pode ser feito selecionando a página de preferências **Janela > Preferências... > z/OS Soluções > Subprojetos do USS > Opções de Ação Remota**. Consulte o “Configuração de REXEC (ou SSH)” para saber qual porta é usada.

## Método de Conexão Alternativo do RSE

Os clientes do Developer para System z precisam conhecer os dois valores para iniciar a conexão do RSE por meio do REXEC (ou SSH), como a seguir:

- O diretório no qual o script de inicialização `server.zseries` está localizado.

O `server.zseries` está localizado em `/etc/rdz/`, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes.

**Nota:** Para desencorajar o uso do REXEC (ou SSH) como método de logon alternativo, o `server.zseries` não é mais copiado automaticamente para `/etc/rdz`. Se desejar usar esta função, você deve copiá-la manualmente a partir do `/usr/lpp/rdz/bin`, como mostrado no seguinte comando de amostra:

```
cp /usr/lpp/rdz/bin/server.zseries /etc/rdz
```

- A porta que está sendo usada.  
Consulte o “Configuração de REXEC (ou SSH)” para saber qual porta é usada.

## Configuração de REXEC (ou SSH)

O *Communications Server IP Configuration Guide* (SC31-8775) descreve as etapas necessárias para configurar o REXEC (ou o SSH). Consulte o Apêndice C, “Configurando o INETD”, na página 315 para conhecer algumas considerações de configuração específicas do Developer para System z (não existem etapas de configuração específicas do Developer para System z).

Uma porta comum usada pelo REXEC é 512. Para verificar isso, você pode verificar os serviços `/etc/inetd.conf` e `/etc/services` para localizar o número de porta usado.

- 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`
- Localize a porta (2ª palavra, `512`, neste exemplo) conectada a este 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:** `/etc/inetd.conf` e `/etc/services` podem ter nomes diferentes. Consulte o Apêndice C, “Configurando o INETD”, na página 315 para obter informações adicionais.

---

## (Opcional) Transação APPC para o Serviço de Comandos TSO

Você precisará da assistência de um administrador de APPC e de um administrador de WLM para concluir essa tarefa de customização, que exige os seguintes recursos ou tarefas de customização especiais:

- Atualização de LINKLIST
  - Transação APPC
  - Atualização do WLM
- 

O serviço TSO Commands pode ser implementado como um programa de transações APPC, FEKFRSRV. Essa transação age como um servidor host para executar comandos do TSO e do ISPF que são emitidos a partir da estação de trabalho. A APPC não é necessária na estação de trabalho, pois o cliente se comunica com o FEKFRSRV por meio do RSE. Cada cliente pode ter uma conexão ativa com vários hosts ao mesmo tempo.

**Nota:**

- Por padrão, o Developer para System z utiliza o Gateway do Cliente TSO/ISPF do ISPF para acessar o serviço de Comandos do TSO.
- Se você não estiver familiarizado com APPC, consulte o Apêndice D, “Configurando o APPC”, na página 323 antes de continuar com esta seção.
- O RSE usa a API do soquete TCP/IP REXX para se comunicar com FEKFRSRV. Isso significa que a biblioteca de carregamento TCP/IP, TCPIP.SEZALOAD padrão, deve estar disponível por meio da diretiva LINKLIST ou STEPLIB em `rsed.envvars`.
- O RSE deve ser reiniciado para que as alterações descritas sejam efetivadas.

## Preparação

- As tarefas a seguir são um pré-requisito e devem ser concluídas antes da configuração do Servidor de Comandos do TSO. As publicações mencionadas descrevem essas tarefas.
  1. Instale, configure e inicie o VTAM no sistema z/OS. Consulte *Communications Server IP SNA Network Implementation Guide* (SC31-8777) para obter informações adicionais.
  2. Instale, configure e inicie o TCP/IP no sistema z/OS. Consulte o Apêndice B, “Configurando o TCP/IP”, na página 307 para obter informações adicionais.

3. Configure e inicie o APPC e o subsistema ASCH (planejador de transações APPC). Consulte o Apêndice D, “Configurando o APPC”, na página 323 para obter informações adicionais.
- A amostra REXX a seguir pode ser usada para gerenciar o APPC através de painéis ISPF.

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

Figura 32. REXX para Painéis ISPF do APPC

**Nota:** Lembre-se de que é possível 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 16. Lista de verificação de transações APPC

Experiência	Informações necessárias: • Valor Padrão • Onde encontrar a resposta	Valor
APPC	Nome do conjunto de dados de TPDATA • Padrão: SYS1.APPCTP • O valor está listado em SYS1.PARMLIB(APPCPMxx)	
APPC	Nome da transação a ser usada (talvez não exista) • Padrão: FEKFRSRV • Transações existentes podem ser consultadas selecionando-se "Administração de Perfis TP" no menu ISPF APPC	
APPC	Classe de transação APPC a ser usada • Padrão: A • As classes APPC são definidas em SYS1.PARMLIB(ASCHPMxx)	
WLM/SRM	Grupo e domínio de desempenho do TSO • Sem padrão IBM (dependente do site)	
RACF	Cada usuário do Developer para System z possui acesso a um segmento OMVS (isto é necessário) • Sem padrão IBM (dependente do site) • O comando RACF do TSO LU <b>userid OMVS</b> exibirá um segmento OMVS pessoal existente	

Tabela 16. Lista de verificação de transações APPC (continuação)

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

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

## Implementação

**Nota:** Os membros de amostra FEKAPPC\* estão localizados em FEK.#CUST.JCL , a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP) . Consulte “Configuração da Customização” na página 15 para obter mais detalhes.

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 usada pelo programa de transações FEKFRSRV. Essa classe é usada na JCL de amostra FEK.#CUST.JCL(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 TSO Commands precisa que as especificações padrão sejam especificadas nas seções OPTIONS e TPDEFAULT de SYS1.PARMLIB(ASCHPMxx). Consulte o Apêndice D, “Configurando o APPC”, na página 323 para obter informações adicionais.
  - A classe de transação APPC usada deve ter iniciadores APPC suficientes para permitir um inicializador para cada usuário simultâneo do Developer para System z.
2. Defina a transação do APPC que agirá como um servidor de comandos. Você pode utilizar a JCL de amostra FEK.#CUST.JCL(FEKAPPCC) para definir essa transação. As instruções sobre como customizar esta JCL estão localizadas na JCL.

**Nota:**

- a. Se você alterou o nome do programa de transações (padrão FEKFRSRV) nesta etapa, o novo nome deverá ser designado à `_FEKFSCMD_TP_NAME_` em `rsed.envvars`, conforme descrito em “Arquivo de Configuração do RSE `rsed.envvars`” na página 31.



- b. A transação APPC utiliza o REXX exec FEKFRSRV, localizado em FEK.SFEKPROC. Não altere esta localização se desejar que seja possível que a manutenção SMP/E seja ativada automaticamente.
  - c. A JCL de amostra também é fornecida para exibir, FEK.#CUST.JCL(FEKAPPCL), ou excluir, FEK.#CUST.JCL(FEKAPPCX), a transação.
3. Ative o RSE para utilizar APPC ao remover o comentário da diretiva RSE\_JAVA\_OPTS="\$RSE\_JAVA\_OPTS -DTSO\_SERVER=APPC" em rsed.envvars, conforme descrito em "Arquivo de Configuração do RSE rsed.envvars" na página 31.
4. Controle a prioridade de dispatch do programa de transações associando FEKFRSRV a um grupo de domínios e de desempenhos no WLM (Workload Manager). Como o FEKFRSRV emite comandos do TSO, ele deve ser designado a um grupo de desempenho TSO.
5. Defina um segmento OMVS padrão para o sistema ou um indivíduo para cada usuário do Developer para System z.
6. Forneça aos usuários do Developer para System z o acesso READ para FEK.SFEKPROC, a biblioteca executável TSO do Developer para System z.

## Considerações sobre Uso de APPC

- Com o uso de APPC para o serviço TSO Commands, o Developer para 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 sobre a customização de TCP/IP e do Resolver, consulte Apêndice B, "Configurando o TCP/IP", na página 307 e *Instruções de Configuração de TCPIP.DATA em Communications Server IP Configuration Reference* (SC31-8776).  
Você pode testar sua configuração TCP/IP iniciando o daemon RSE com o parâmetro IVP=IVP ou com o programa de verificação de instalação fekfivpt (IVP), conforme documentado no Capítulo 7, "Verificação de Instalação", na página 107.
- Com o uso de APPC para o serviço TSO Commands, o Developer para System z requer um soquete extra (porta TCP/IP) para a comunicação limitada pelo host por arquivo MVS aberto. Qualquer porta disponível será usada. Esse mecanismo de seleção de porta não pode ser alterado.
- Com o uso de APPC para o serviço TSO Commands, a leitura e a gravação de um conjunto de dados do MVS exige 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. Consulte o "Falhas ao Abrir Conjuntos de Dados MVS" na página 154.
- Ao usar APPC para o serviço TSO Commands para evitar a configuração do TSO/ISPF Client Gateway do ISPF, lembre-se de que outros serviços, como o SCLM Developer Toolkit, contam com o TSO/ISPF Client Gateway.
- Consulte o Apêndice D, "Configurando o APPC", na página 323 para obter as considerações gerais de uso de APPC

---

## (Opcional) Limpeza de WORKAREA

Esta tarefa de customização não requer assistência, recursos especiais ou tarefas de customização especiais.

---

O TSO/ISPF Client Gateway do ISPF e a função SCLM Developer Toolkit usam o diretório WORKAREA para armazenar arquivos de trabalho temporários, que são removidos antes de a sessão ser fechada. Entretanto, a saída temporária é às vezes deixada para trás, por exemplo, se existir um erro de comunicação durante o processamento. Por essa razão, é recomendável limpar o diretório WORKAREA regularmente.

O z/OS UNIX fornece um script de shell, skulker, que exclui arquivos com base no diretório em que estão e em suas idades. Combinado com o daemon cron do z/OS UNIX, que executa comandos em datas e horas especificadas, você pode configurar uma ferramenta automatizada que limpe periodicamente o diretório WORKAREA. Consulte o *UNIX System Services Command Reference* (SA22-7802) para obter informações adicionais sobre o script skulker e o daemon cron.

**Nota:** O WORKAREA está localizado em /var/rdz/, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes.

---

## Capítulo 7. Verificação de Instalação

Após concluir a customização do produto, é possível usar os Installation Verification Programs (IVPs) descritos neste capítulo para verificar o êxito da configuração dos componentes principais do produto.

---

### Verificar Tarefas Iniciadas

#### JMON, JES Job Monitor

Inicie a tarefa iniciada JMON (ou a tarefa do usuário). As informações de inicialização no DD STDOUT devem terminar com a seguinte mensagem:

```
JM200I Inicialização do servidor concluída.
```

Se a tarefa terminar com o código de retorno 66, FEK.SFEKAUTH não será autorizada pelo APF.

**Nota:** Inicie o JES Job Monitor antes de continuar com os outros testes do IVP.

#### LOCKD, Daemon de Bloqueio

Inicie a tarefa iniciada LOCKD (ou a tarefa do usuário). O daemon de bloqueio emite a seguinte mensagem do console na inicialização bem-sucedida:

```
FEK501I Lock daemon started, port=4036, cleanup interval=1440,  
log level=1
```

#### RSED, Daemon RSE

Inicie a tarefa RSED iniciada (ou tarefa do usuário) com o parâmetro IVP=IVP. Com esse parâmetro, o servidor será encerrado após executar alguns testes de verificação de instalação. A saída desses testes está disponível em DD STDOUT. No caso de alguns erros, os dados também estão disponíveis em DD STDERR. Os dados de STDOUT devem ser semelhantes à seguinte amostra:

**Nota:** Inicie o daemon RSE sem o parâmetro IVP, antes de continuar com os outros testes do IVP. O daemon RSE emite a seguinte mensagem do console na inicialização bem-sucedida:

```
FEK002I RseDaemon started. (port=4035)
```

teste do IVP do daemon RSE

```
Quarta-feira, 2 de julho, 17h11min52s 2008 UTC  
uid=8(STCRSE) gid=1(STCGROUP)
```

A porta do daemon RSE é 4035  
Arquivos de configuração do RSE localizados em /etc/rdz

```
-----  
variáveis de ambiente atuais  
-----
```

```
@="/usr/lpp/rdz/bin/rsed.sh" @[1]="4035" @[2]="/etc/rdz"  
CGI_DTCONF="/var/rdz/scldmt"  
CGI_DTWORK="/var/rdz"  
CGI_TRANTABLE="FEK.#CUST.LSTRANS.FILE"  
CLASSPATH=".:usr/lpp/rdz/lib:usr/lpp/rdz/lib/dstore_core.jar:usr/lpp/  
ERRNO=""  
HOME="/tmp"
```

```

IFS=""
"
JAVA_HOME="/usr/lpp/java/J5.0"
JAVA_PROPAGATE="NO"
LANG="C"
LIBPATH=".:usr/lib:usr/lpp/java/J5.0/bin:usr/lpp/java/J5.0/bin/classi
LINENO="66"
LOGNAME="STCRSE"
MAILCHECK="600"
OLDPWD="/tmp"
OPTIND="1"
PATH=".:usr/lpp/java/J5.0/bin:usr/lpp/rdz/bin:usr/lpp/ispf/bin:bin:/
PPID="33554711"
PS1="\$ "
PS2="> "
PS3="#? "
PS4="+ "
PWD="/etc/rdz"
RANDOM="27298"
RSE_CFG="/etc/rdz"
RSE_HOME="/usr/lpp/rdz"
RSE_LIB="/usr/lpp/rdz/lib"
SECONDS="0"
SHELL="/bin/sh"
STEPLIB="NONE"
TZ="EST5EDT"
_BPX_SHAREAS="YES"
_BPX_SPAWN_SCRIPT="YES"
_CEE_DMPTARG="/tmp"
_CEE_RUNOPTS="ALL31(ON) HEAP(32M,32K,ANYWHERE,KEEP,,) TRAP(ON)"
_CMDSERV_BASE_HOME="/usr/lpp/ispf"
_CMDSERV_CONF_HOME="/etc/rdz"
_CMDSERV_WORK_HOME="/var/rdz"
_RSE_CMDSERV_OPTS="&SESSION=SPAWN"
_RSE_DAEMON_CLASS="com.ibm.etools.zos.server.RseDaemon"
_RSE_DAEMON_IVP_TEST="1"
_RSE_DAEMON_PORT="4035"
_RSE_JAVAOPTS=" -DISPF_OPTS='&SESSION=SPAWN' -DA_PLUGIN_PATH=/usr/lpp/rd
_RSE_POOL_SERVER_CLASS="com.ibm.etools.zos.server.ThreadPoolProcess"
_RSE_PWD="/tmp"
_RSE_SERVER_CLASS="org.eclipse.dstore.core.server.Server"
_RSE_SERVER_TIMEOUT="120000"
_SCLMDT_BASE_HOME="/usr/lpp/rdz"
_SCLMDT_CONF_HOME="/var/rdz/sclmdt"
_SCLMDT_TRANSTABLE="FEK.#CUST.LSTRANS.FILE"
_SCLMDT_WORK_HOME="/var/rdz"
_SCLM_BASE="/var/rdz/WORKAREA"
_SCLM_BWBCALL="/usr/lpp/rdz/bin/BWBCALL"
_SCLM_DWGET="/var/rdz/WORKAREA"
_SCLM_DWTRANSFER="/var/rdz/WORKAREA"
_SCLM_J2EPUT="/var/rdz/WORKAREA"

```

```

-----
teste de inicialização java...
-----

```

```

java versão "1.5.0"
Java(TM) 2 Runtime Environment, Standard Edition (build pmz31dev-2008031
IBM J9 VM (build 2.3, J2RE 1.5.0 IBM J9 2.3 z/OS s390-31 j9vmnz3123-2008
J9VM - 20080314_17962_bHdSMr
JIT - 20080130_0718ifx2_r8
GC - 200802_08)
JCL - 20080314

```

```

-----
teste do IVP de TCP/IP...
-----

```

```
Quarta-feira, 2 de julho de 2008, 13h11min54s EDT
uid=8(STCRSE) gid=1(STCGROUP)
utilizando /etc/rdz/rsed.envvars

-----
Configuração do resolvedor TCP/IP (ordem de procura do z/OS UNIX):
-----
Resolver Trace Initialization Complete -> 2008/07/02 13:11:54.745964

res_init Resolver values:
Global Tcp/Ip Dataset = None
Default Tcp/Ip Dataset = None
Local Tcp/Ip Dataset = /etc/resolv.conf
Translation Table = Default
UserId/JobName = STCRSE
Caller API = LE C Sockets
Caller Mode = EBCDIC
(L) DataSetPrefix = TCPIP
(L) HostName = CDFMVS08
(L) TcpIpJobName = TCPIP
(L) DomainOrigin = RALEIGH.IBM.COM
(L) NameServer = 9.42.206.2
                  9.42.206.3
(L) NsPortAddr = 53 (L) ResolverTimeout = 10
(L) ResolveVia = UDP (L) ResolverUdpRetries = 1
(*) Options NDots = 1
(*) SockNoTestStor
(*) AlwaysWto = NO (L) MessageCase = MIXED
(*) LookUp = DNS LOCAL
res_init Succeeded
res_init Started: 2008/07/02 13:11:54.755363
res_init Ended: 2008/07/02 13:11:54.755371
*****
MVS TCP/IP NETSTAT CS V1R9 TCPIP Name: TCPIP 13:11:54
Tcpiip started at 01:28:36 on 06/23/2008 with IPv6 enabled

-----
endereço IP do host:
-----
hostName=CDFMVS08
hostAddr=9.42.112.75
bindAddr=9.42.112.75
localAddr=9.42.112.75

Êxito, os endereços correspondem

-----
PassTicket IVP test...
-----

Success, PassTicket IVP finished normally

-----
IVP do daemon RSE finalizado
```

Verificar Serviços

A instalação do Developer para System z fornece vários Installation Verification Programs (IVP) para os serviços básicos e opcionais. Os scripts do IVP estão localizados no diretório de instalação, padrão /usr/lpp/rdz/bin/.

Tabela 17. IVPs para Serviços

fekfivpa	“(Opcional) Conexão do Serviço TSO Commands Utilizando APPC” na página 115
----------	---

Tabela 17. IVPs para Serviços (continuação)

fekfivpd	"Conexão do Daemon RSE" na página 113
fekfivpi	"Conexão do TSO/ISPF Client Gateway do ISPF" na página 114
fekfivpj	"Conexão do JES Job Monitor" na página 113
fekfivpl	"Conexão do Daemon de Bloqueio" na página 113
fekfivpr	"(Opcional) Conexão REXEC" na página 118
fekfivps	"(Opcional) Conexão SCLMDT" na página 116
fekfivpt	"Configuração de TCP/IP" na página 112
fekfivpz	"(Opcional) Script de Shell REXEC/SSH" na página 119

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

Um tamanho de região grande é necessário para o ID do usuário que executa os IVPs, pois as funções, como Java, que exigem muita memória, serão executadas. Você deve definir o tamanho da região como 131072 kilobytes (128 megabytes) ou mais alto.

O seguinte erro de amostra é uma indicação clara de um tamanho de região insuficiente. (Mas outros erros podem ocorrer, também. Por exemplo, Java pode falhar ao iniciar).

```
CEE5213S The signal SIGPIPE was received.
%z/OS UNIX command%: command was killed by signal number 13
  %line-number% **  %REXX command%
    +++ RC(137) +++
```

**Nota:** As tarefas iniciadas do Developer para System z devem estar ativas antes de iniciar o teste do IVP.

## Inicialização do IVP

Todos os comandos de amostra nesta seção esperam que certas variáveis de ambiente sejam configuradas. Dessa forma, os scripts do IVP ficam disponíveis através da instrução PATH e o local dos arquivos de configuração customizados é conhecido. Use os comandos **pwd** e **cd** para verificar e alterar seu diretório atual para o diretório com os arquivos de configuração customizados. O shell script **ivpinit** pode então ser usado para configurar as variáveis de ambiente RSE, como na seguinte amostra (\$ é o prompt do z/OS UNIX):

```
$ pwd
/u/userid
$ cd /etc/rdz
$ ./ivpinit
Arquivos de configuração do RSE localizados em /etc/rdz -- padrão
incluído /usr/lpp/rdz/bin em PATH
```

O primeiro "." (ponto) em **./ivpinit** é um comando do z/OS UNIX para executar o shell no ambiente atual, para que os conjuntos de variáveis do ambiente no shell sejam efetivados após sair do shell. O segundo ponto está relacionado ao diretório atual.

#### Nota:

- Se `./ivpinit` NÃO é executado antes dos scripts `fekfivp*`, o caminho para esses scripts deve ser especificado quando eles forem chamados, como na seguinte amostra:

```
/usr/lpp/rdz/bin/fekfivpr 512 USERID
```

Além disso, a maioria dos scripts `fekfivp*` solicitará o local do `rzed.envvars` customizado se `./ivpinit` não for executado primeiro.

- 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 `STEPLIB`. Os seguintes comandos podem ser necessários para que seja possível executar esses testes IVP (\$ é o prompt do z/OS UNIX):

```
$ EXPORT STEPLIB=$STEPLIB:TCPIP.SEZALOAD
```

Observe que a inclusão de uma biblioteca não autorizada pelo APF em um `STEPLIB` existente remove a autorização APF para conjuntos de dados `STEPLIB` existentes.

Observe também que se `CEE.SCEELKED` estiver em `LINKLIST` ou `STEPLIB`, `TCPIP.SEZALOAD` deverá ser colocado antes de `CEE.SCEELKED`. Se isso não for feito, o sistema 0C1 encerrará de forma anormal para as chamadas do soquete TCP/IP REXX.

Para obter informações sobre o diagnóstico de problemas de conexão do RSE, consulte Capítulo 9, “Resolução de Problemas de Configuração”, na página 135 ou as Notas Técnicas na Página de Suporte do Developer para System z, em <http://www-306.ibm.com/software/awdtools/rdz/support/>.

## Disponibilidade de Porta

A disponibilidade da porta do JES Job Monitor, do daemon RSE e, opcionalmente, de REXEC ou SSH pode ser verificada emitindo-se o comando **netstat**. O resultado deve mostrar as portas usadas por esses serviços, como nas seguintes amostras (\$ é o prompt do z/OS UNIX):

#### IPv4

```
$ netstat
MVS TCP/IP NETSTAT CS VxRy TCPIP Name: TCPIP      13:57:36
User Id Conn      Local Socket      Foreign Socket    State
-----
RSED    0000004B 0.0.0.0..4035     0.0.0.0..0        Listen
LOCKD   0000004C 0.0.0.0..4036     0.0.0.0..0        Listen
JMON    00000037 0.0.0.0..6715     0.0.0.0..0        Listen
```

#### IPv6

```
$ netstat
MVS TCP/IP NETSTAT CS VxRy TCPIP Name: TCPIP      14:03:35
User Id Conn      State
-----
RSED    0000004B Listen
Local Socket: 0.0.0.0..4035
Foreign Socket: 0.0.0.0..0
LOCKD   0000004C Listen
Local Socket: 0.0.0.0..4036
Foreign Socket: 0.0.0.0..0
JMON    00000037 Listen
Local Socket: 0.0.0.0..6715
Foreign Socket: 0.0.0.0..0
```



## Configuração de TCP/IP

Com o uso de APPC para o serviço TSO Commands, o Developer para 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 o Apêndice B, "Configurando o TCP/IP", na página 307 para obter informações adicionais sobre a configuração TCP/IP e do Resolver. Verifique as configurações atuais executando o seguinte comando:

```
fekfivpt
```

**Nota:** Esse IVP emite o comando TCPIP **netstat**, que pode ser protegido contra execução por seu software de segurança.

O comando deve retornar uma saída como nesta amostra (\$ é o prompt do z/OS UNIX):

```
$ fekfivpt
```

```
Quarta-feira, 2 de julho de 2008, 13h11min54s EDT
uid=1(USERID) gid=0(GROUP)
utilizando /etc/rdz/rsed.envvars
```

```
o limite de tamanho do espaço de endereço atual é 1914675200 (1826,0 MB)
o limite de tamanho do espaço de endereço máximo é 2147483647 (2048,0 MB)
```

```
-----
Configuração do resolvedor TCP/IP (ordem de procura do z/OS UNIX):
```

```
-----
Resolver Trace Initialization Complete -> 2008/07/02 13:11:54.745964
```

```
res_init Resolver values:
Global Tcp/Ip Dataset = None
Default Tcp/Ip Dataset = None
Local Tcp/Ip Dataset = /etc/resolv.conf
Translation Table = Default
UserId/JobName = USERID
Caller API = LE C Sockets
Caller Mode = EBCDIC
(L) DataSetPrefix = TCPIP
(L) HostName = CDFMVS08
(L) TcpIpJobName = TCPIP
(L) DomainOrigin = RALEIGH.IBM.COM
(L) NameServer = 9.42.206.2
                  9.42.206.3
(L) NsPortAddr = 53 (L) ResolverTimeout = 10
(L) ResolveVia = UDP (L) ResolverUdpRetries = 1
(*) Options NDots = 1
(*) SockNoTestStor
(*) AlwaysWto = NO (L) MessageCase = MIXED
(*) LookUp = DNS LOCAL
```

```
res_init Succeeded
```

```
res_init Started: 2008/07/02 13:11:54.755363
```

```
res_init Ended: 2008/07/02 13:11:54.755371
```

```
*****
```

```
MVS TCP/IP NETSTAT CS V1R9 TCPIP Name: TCPIP 13:11:54
```

```
Tcpip started at 01:28:36 on 06/23/2008 with IPv6 enabled
```

```
-----
endereço IP do host:
```

```
-----
hostName=CDFMVS08
hostAddr=9.42.112.75
```

```
bindAddr=9.42.112.75
localAddr=9.42.112.75
```

Êxito, os endereços correspondem

## Conexão do Daemon RSE

Para verificar a conexão do daemon RSE, execute o seguinte comando. Substitua 4035 pela porta usada 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 essa no seguinte exemplo (\$ é o prompt do z/OS UNIX):

```
$ fekfivpd 4035 USERID
```

```
Quarta-feira, 2 de julho de 2008, 15h00m27s EDT
uid=1(USERID) gid=0(GROUP)
utilizando /etc/rdz/rsed.envvars
```

```
o limite de tamanho do espaço de endereço atual é 1914675200 (1826,0 MB)
o limite de tamanho do espaço de endereço máximo é 2147483647 (2048,0 MB)
```

```
Senha:
SSL está desativado
conectado
8108
570655399
Bem-sucedido
```

**Nota:** Ao testar uma conexão ativada pelo SSL, verifique se você especificou a porta correta, caso receba 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 seguinte amostra (\$ é o prompt do z/OS UNIX):

```
$ fekfivpj 6715
```

```
Quarta-feira, 2 de julho de 2008, 15h00m27s EDT
uid=1(USERID) gid=0(GROUP)
using /etc/rdz/rsed.envvars
```

```
o limite de tamanho do espaço de endereço atual é 1914675200 (1826,0 MB)
o limite de tamanho do espaço de endereço máximo é 2147483647 (2048,0 MB)
```

```
hostName=CDFMVS08
hostAddr=9.42.112.75
Aguardando resposta do JES Job Monitor...
ACKNOWLEDGE01v03
```

Bem-sucedido

## Conexão do Daemon de Bloqueio

Verifique a conexão do daemon de bloqueio executando o seguinte comando.

```
fekfivpl
```

O comando deve retornar uma saída como na seguinte amostra (\$ é o prompt do z/OS UNIX):

```
$ fekfivpl
```

```
Segunda-feira, 29 de junho de 2009, 8h00m38s EDT  
uid=1(USERID) gid=0(GROUP)  
utilizando /etc/rdz/rsed.envvars
```

```
o limite de tamanho do espaço de endereço atual é 1914675200 (1826,0 MB)  
o limite de tamanho do espaço de endereço máximo é 2147483647 (2048,0 MB)
```

```
hostName=CDFMVS08  
hostAddr=9.42.112.75
```

```
Registrando usuário no Daemon de Bloqueio...  
Aguardando resposta do Daemon de Bloqueio...
```

```
Consultando o Daemon de Bloqueio...  
Aguardando resposta do Daemon de Bloqueio...  
USERID
```

```
Cancelando o registro do usuário do Daemon de Bloqueio...  
Aguardando resposta do Daemon de Bloqueio...
```

```
Consultando o Daemon de Bloqueio...  
Aguardando resposta do Daemon de Bloqueio...
```

Bem-sucedido

## Conexão do TSO/ISPF Client Gateway do ISPF

Verifique a conexão com o TSO/ISPF Client Gateway do ISPF executando o seguinte comando:

```
fekfivpi
```

O comando deve retornar o resultado de verificações relacionadas ao TSO/ISPF Client Gateway do ISPF, (variáveis, módulos HFS, início e parada da sessão TSO/ISPF), como na seguinte amostra (\$ é o prompt do z/OS UNIX):

```
$ fekfivpi
```

```
Quarta-feira, 2 de julho de 2008, 15h00m27s EDT  
uid=1(USERID) gid=0(GROUP)  
utilizando /etc/rdz/rsed.envvars
```

```
o limite de tamanho do espaço de endereço atual é 1914675200 (1826,0 MB)  
o limite de tamanho do espaço de endereço máximo é 2147483647 (2048,0 MB)
```

```
-----  
conteúdo do /etc/rdz/ISPF.conf:  
-----
```

```
ispmllib=ISP.SISPMENU  
isptlib=ISP.SISPTENU  
ispplib=ISP.SISPPENU  
ispslib=ISP.SISPSLIB  
sysproc=ISP.SISPCLIB,FEK.SFEKPROC
```

```
-----  
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:

```
Server PATH          =  
/usr/lpp/java/J5.0/bin:/usr/lpp/rdz/lib:/usr/lpp/ispf/bin:  
/bin:/usr/sbin:.
```

```
STEPLIB              = FEK.SFEKAUTH:FEK.SFEKLOAD
```

```
_CMDSERV_BASE_HOME   = /usr/lpp/ispf  
_CMDSERV_CONF_HOME   = /etc/rdz  
_CMDSERV_WORK_HOME   = /var/rdz
```

```
-----  
*** CHECK : USS MODULES  
Verificando o Diretório ISPF : /usr/lpp/ispf  
Verificando módulos em /usr/lpp/ispf/bin directory  
Verificando arquivo de configuração do ISPF ISPF.conf  
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 ISPF falhar, informações mais detalhadas serão mostradas.

fekfivpi tem os seguintes parâmetros opcionais não-posicionais:

**-file** fekfivpi pode produzir grandes quantidades de saída (centenas de linhas). O parâmetro -file envia essa saída para um arquivo, userlog/.eclipse/RSE/\$LOGNAME/fekfivpi.log, em que userlog é o valor da diretiva user.log em rsed.envvars e \$LOGNAME é o ID do usuário (em maiúsculas). Se a diretiva user.log for comentada ou não estiver presente, o caminho inicial será usado. O caminho inicial é definido em seu segmento de segurança OMVS.

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

## (Opcional) Conexão do Serviço TSO Commands Utilizando APPC

Verifique a conexão com o servidor do TSO Commands (usando APPC), ao executar o seguinte comando. Substitua USERID por um ID do 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:
```

```
Quarta-feira, 2 de julho de 2008, 15h00m27s EDT
uid=1(USERID) gid=0(GROUP)
utilizando /etc/rdz/rsed.envvars
```

```
o limite de tamanho do espaço de endereço atual é 1914675200 (1826,0 MB)
o limite de tamanho do espaço de endereço máximo é 2147483647 (2048,0 MB)
```

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

## (Opcional) Conexão SCLMDT

Verifique a conexão com o SCLM Developer Toolkit executando o seguinte comando:

```
fekfivps
```

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

```
$ fekfivps
```

```
Quarta-feira, 2 de julho de 2008, 15h00m27s EDT
uid=1(USERID) gid=0(GROUP)
utilizando /etc/rdz/rsed.envvars
```

```
o limite de tamanho do espaço de endereço atual é 1914675200 (1826,0 MB)
o limite de tamanho do espaço de endereço máximo é 2147483647 (2048,0 MB)
```

```
-----
conteúdo do /etc/rdz/ISPF.conf:
-----
```

```
isplib=ISP.SISPMENU
isptlib=ISP.SISPTENU
ispplib=ISP.SISPPENU
ispslib=ISP.SISPSLIB
sysproc=ISP.SISPCLIB,FEK.SFEKPROC
-----
```

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

```

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

Server PATH          = /usr/lpp/java/J5.0/bin:/usr/lpp/rdz/lib:/usr/lpp/ispf/bin:
/bin:/usr/sbin:.

STEPLIB              = FEK.SFEKAUTH:FEK.SFEKLOAD

_CMDSERV_BASE_HOME   = /usr/lpp/ispf
_CMDSERV_CONF_HOME   = /etc/rdz
_CMDSERV_WORK_HOME    = /var/rdz
_SCLMDT_CONF_HOME     = /var/rdz/sclmdt
_SCLMDT_WORK_HOME     = /var/rdz
_SCLMDT_TRANTABLE     = FEK.#CUST.LSTRANS.FILE

```

```

-----
*** CHECK : VERIFICAÇÃO DE CONFIGURAÇÃO DO CAMINHO JAVA
RC=0
MSG: BEM-SUCEDIDO

```

```

-----
*** CHECK : USS MODULES
Verificando o Diretório ISPF : /usr/lpp/ispf
Verificando módulos no diretório /usr/lpp/ispf/bin
Verificando o arquivo de configuração do ISPF ISPF.conf
Verificando o diretório bin de instalação : /usr/lpp/rdz/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.

fekfivps tem os seguintes parâmetros opcionais e não-posicionais:

**-file** fekfivps pode produzir grandes quantidades de saída (centenas de linhas). O parâmetro -file envia essa saída para um arquivo, userlog/.eclipse/RSE/\$LOGNAME/fekfivps.log, em que userlog é o valor da diretiva user.log em rsed.envvars e \$LOGNAME é o ID do usuário (em maiúsculas). Se a diretiva user.log for comentada ou não estiver presente, o caminho inicial será usado. O caminho inicial é definido em seu segmento de segurança OMVS.

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

## (Opcional) Conexão REXEC

Para verificar a conexão REXEC, execute o seguinte comando. Substitua 512 pela porta usada 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 seguinte amostra (\$ é o prompt do z/OS UNIX):

```
$ fekfivpr 512 USERID
Digite a Senha:
```

```
Quarta-feira, 2 de julho de 2008, 15h00m27s EDT
uid=1(USERID) gid=0(GROUP)
utilizando /etc/rdz/rsed.envvars
```

```
o limite de tamanho do espaço de endereço atual é 1914675200 (1826,0 MB)
o limite de tamanho do espaço de endereço máximo é 2147483647 (2048,0 MB)
```

```
$ EYZRC01I Chamando a função rexec_af com o seguinte:
EYZRC02I Host: CDFMVS08, user USERID, cmd cd /etc/rdz;export RSE_USER_ID=USERI
D;./server.zseries -ivp, port 512
EYZRC19I Data socket = 4, Control socket = 6.
```

teste do IVP do servidor RSE

```
CDFMVS08 -- Quarta-feira, 2 de julho de 2008, 15h00m27s EDT
uid=1(USERID) gid=0(GROUP)
```

Arquivos de configuração RSE localizados em /etc/rdz --default

O ID do usuário RSE éUSERID --default

-----  
Limites de tamanho do Espaço de Endereço

-----  
o limite de tamanho do espaço de endereço atual é 2147483647 (2048,0 MB)  
o limite de tamanho do espaço de endereço máximo é 2147483647 (2048,0 MB)

-----  
histórico do serviço

-----  
Sexta-feira, 19 de junho de 2009 00h01m00s -- COPY -- HHOP760 v7600  
criado em 18 de junho de 2009

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

-----  
iniciando servidor RSE no plano de fundo -- Sexta-feira,  
19 de junho de 2009 15h59m05s EDT

-----  
java versão "1.5.0"  
Java(TM) 2 Runtime Environment, Standard Edition (build pmz31dev-20070201 (SR4))  
IBM J9 VM (build 2.3, J2RE 1.5.0 IBM J9 2.3 z/OS s390-31 j9vmmz3123-20070201 (JI  
T ativado)  
J9VM - 20070131\_11312\_bHdSMr  
JIT - 20070109\_1805ifx1\_r8  
GC - 200701\_09)  
JCL - 20070126

Iniciando Servidor DStore...  
Servidor Iniciado com Êxito  
8108  
Servidor executando em: CDFMVS08



**Nota:**

- Se você não obtiver nenhuma saída Java e do servidor RSE, provavelmente o tamanho da região INETD é muito pequeno (deve ser 2096128 ou maior, se for iniciado a partir de uma sessão de shell TSO/OMVS, ou tamanho de região 0 para BPXBATCH).
- Você pode testar o script de shell usado pelo REXEC separadamente, conforme descrito no próximo teste de IVP “(Opcional) Script de Shell REXEC/SSH”.
- O servidor é iniciado sem a tentativa de conexão de um cliente, portanto atingirá o tempo limite (depois de 5 segundos). Isso resulta em uma mensagem de erro de conexão semelhante à seguinte amostra:

```
Erro de conexão
Servidor: erro ao inicializar soquete: java.net.SocketTimeoutException:
Tempo limite de aceitação excedido
```

## **(Opcional) Script de Shell REXEC/SSH**

Esse teste de IVP pode ser ignorado se o teste anterior, descrito em “(Opcional) Conexão REXEC” na página 118, for concluído com êxito.

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

```
fekfivpz
```

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

```
$ fekfivpz
```

```
Quarta-feira, 2 de julho de 2008, 15h00m27s EDT
uid=1(USERID) gid=0(GROUP)
```

```
utilizando /etc/rdz/rsed.envvars
```

```
o limite de tamanho do espaço de endereço atual é 1914675200 (1826,0 MB)
o limite de tamanho do espaço de endereço máximo é 2147483647 (2048,0 MB)
```

```
-----
Teste do IVP do servidor RSE
```

```
CDFMVS08 -- Quarta-feira, 2 de julho de 2008 15h00m27s EDT
uid=1(USERID) gid=0(GROUP)
```

```
Arquivos de configuração do RSE localizados em /etc/rdz --default
0 ID do usuário RSE éUSERID --default
```

```
-----
Limites de tamanho do Espaço de Endereço
```

```
-----
o limite de tamanho do espaço de endereço atual é 2147483647 (2048,0 MB)
o limite de tamanho do espaço de endereço máximo é 2147483647 (2048,0 MB)
```

```
-----
histórico do serviço
```

```
-----
Sexta-feira, 19 de junho de 2009 00h01m00s -- COPY -- HHOP760 v7600
criado em 18 de junho de 2009
```

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

```

-----
iniciando servidor RSE no plano de fundo -- Sexta-feira,
19 de junho de 2009 15h59m05s EDT
-----
java versão "1.5.0"
Java(TM) 2 Runtime Environment, Standard Edition (build pmz31dev-20070201 (SR4))
IBM J9 VM (build 2.3, J2RE 1.5.0 IBM J9 2.3 z/OS s390-31 j9vmmz3123-20070201 (JI
T ativado)
J9VM - 20070131_11312_bHdSMr
JIT - 20070109_1805ifx1_r8
GC - 200701_09)
JCL - 20070126

Iniciando Servidor DStore...
Servidor Iniciado com Êxito
8108
Servidor executando em: CDFMVS08

```

**Nota:**

- Se você não obtiver nenhuma saída, o tamanho da região (TSO) será, provavelmente, muito pequeno (deve ser 2096128).
- O servidor é iniciado sem a tentativa de conexão de um cliente, portanto atingirá o tempo limite (depois de 5 segundos). Isso resulta em uma mensagem de erro de conexão semelhante à seguinte amostra:  
 Erro de conexão  
 Servidor: erro ao inicializar soquete: java.net.SocketTimeoutException:  
     Tempo limite de aceitação excedido

---

## **Parte 2. Informações do Developer para System z**



---

## Capítulo 8. Comandos do Operador

Este capítulo fornece uma visão geral dos comandos disponíveis do operador (ou console) para o Developer para System z. Consulte o “Como Ler um Diagrama de Sintaxe” na página 132 se não estiver familiarizado com os diagramas de sintaxe usados para explicar o formato de comando.

---

### Start (S)

Use o comando **START** para iniciar dinamicamente uma tarefa iniciada (STC). A versão abreviada do comando é a letra **S**.

### JES Job Monitor

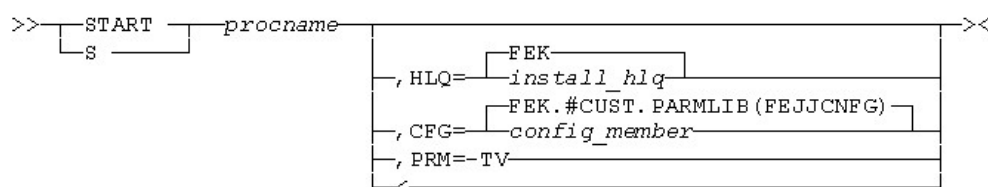


Figura 33. Comando do operador **START JMOM**

#### **procname**

O nome do membro em uma biblioteca de procedimentos que é usada para iniciar o servidor. O nome padrão usado durante a configuração do host é **JMON**.

#### **HLQ=install\_hlq**

Qualificador de alto nível usado para instalar o Developer para System z. O padrão é **FEK**.

#### **CFG=config\_member**

Nome absoluto do conjunto de dados e do membro do arquivo de configuração do JES Job Monitor. O padrão é **FEK.#CUST.PARMLIB (FEJJCNFG)**.

#### **PRM=-TV**

Ativar o modo detalhado (rastreo). O rastreo causará diminuição no desempenho e deverá ser realizado somente sob a orientação do IBM Support Center.

## Daemon RSE

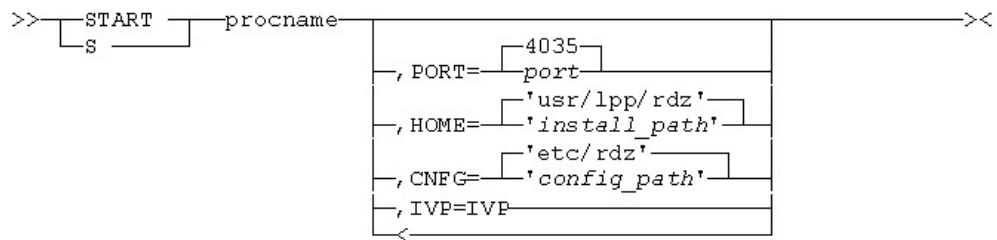


Figura 34. Comando do operador `START RSED`

### **procname**

O nome do membro em uma biblioteca de procedimentos que é usada para iniciar o servidor. O nome padrão usado durante a configuração do host é `RSED`.

### **PORT=port**

A porta usada pelo daemon RSE para os clientes se conectarem. O padrão é 4035.

### **HOME='install\_path'**

O prefixo do caminho e o `/usr/lpp/rdz` obrigatório usado para instalar o Developer para System z. O padrão é `'/usr/lpp/rdz'`. Observe que o caminho do z/OS UNIX faz distinção entre maiúsculas e minúsculas e que deve ser colocado entre aspas simples (') para preservar os caracteres minúsculos.

### **CNFG='config\_path'**

Local absoluto dos arquivos de configuração armazenados no z/OS UNIX. O padrão é `'/etc/rdz'`. Observe que o caminho do z/OS UNIX faz distinção entre maiúsculas e minúsculas e que deve ser colocado entre aspas simples (') para preservar os caracteres minúsculos.

### **IVP=IVP**

Não inicia o servidor, mas executa o Installation Verification Program (IVP) do daemon RSE.

## Daemon de Bloqueio

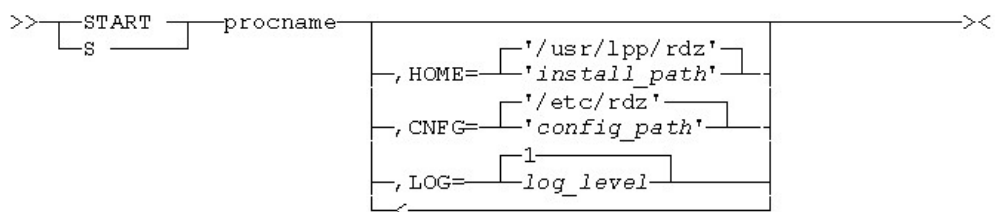


Figura 35. Comando do operador `START LOCKD`

### **procname**

O nome do membro em uma biblioteca de procedimentos que é usada para iniciar o servidor. O nome padrão usado durante a configuração do host é `LOCKD`.

O prefixo do caminho e o `/usr/lpp/rdz` obrigatório usado para instalar o Developer para System z. O padrão é `'/usr/lpp/rdz'`. Observe que o caminho do z/OS UNIX faz distinção entre maiúsculas e minúsculas e que deve ser colocado entre aspas simples (') para preservar os caracteres minúsculos.

Local absoluto dos arquivos de configuração armazenados no z/OS UNIX. O padrão é '/etc/rdz'. Observe que o caminho do z/OS UNIX faz distinção entre maiúsculas e minúsculas e que deve ser colocado entre aspas simples (') para preservar os caracteres minúsculos.

- 0 : Somente mensagens de erro do log.
- 1 : Mensagens de erro e de aviso do log (padrão).
- 2 : Mensagens de erro, aviso e informativas do log.

O comando **MODIFY** permite consultar e alterar dinamicamente as características de uma tarefa ativa. A versão abreviada do comando é a letra F.

```
>> MODIFY procname, APPL=-TV <<
      F, APPL=-TN
```

O nome do membro em uma biblioteca de procedimentos que foi usada para iniciar o servidor. O nome padrão usado durante a configuração do host é JMON.

**-TN** Desativar o modo detalhado (rastreo).

## Daemon RSE

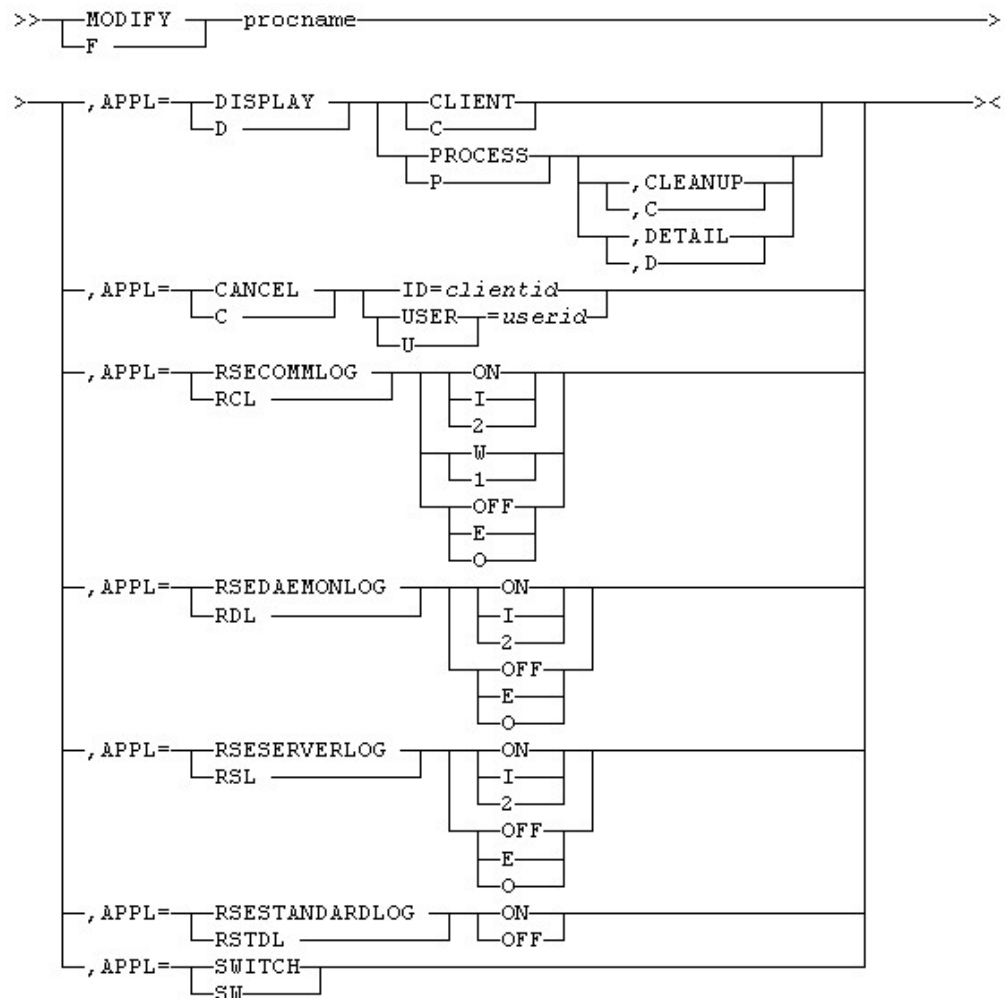


Figura 37. Comando do operador MODIFY RSED

**procname**

O nome do membro em uma biblioteca de procedimentos que foi usada para iniciar o servidor. O nome padrão usado durante a configuração do host é RSED.

## DISPLAY CLIENT

Exibe os clientes ativos.

&lt;id do cliente&gt; : &lt;id do usuário&gt; : &lt;conectado desde&gt;

**DISPLAY PROCESS[,CLEANUP,DETAIL]**

Exibe os processos do conjunto de encadeamento RSE. Pode haver vários processos, que são usados para balanceamento de carga dos usuários conectados.

ID do processo(&lt;ID do processo&gt;) Uso de Memória

```
(<uso do heap Java>%)
```

Clientes(<número de clientes>) Ordem(<ordem de inicialização>)

```
<status do erro>
```

**Nota:**



- <processid> pode ser usado em comandos do operador z/OS UNIX específicos do processo.
- Cada processo tem seu próprio heap Java, cujo tamanho pode ser configurado em `rsed.envvars`.
- <ordem de inicialização> é um número sequencial que indica a ordem em que os conjuntos de encadeamentos foram iniciados. O número corresponde ao número usado no nome do arquivo dos arquivos `stderr.*.log` e `stdout.*.log`.

Em situações normais, o <status do erro> é em branco. O Tabela 18 documenta os possíveis valores não-em branco para <status do erro>.

*Tabela 18. Status do erro do conjunto de encadeamento*

Estado	Descrição
*erro grave*	O processo do conjunto de encadeamento encontrou um erro irreversível e operações interrompidas. Os outros campos de status mostram os últimos valores conhecidos. Use a opção <b>CLEANUP</b> do comando de modificação <b>DISPLAY PROCESS</b> para remover essa entrada da tabela.
*processo interrompido*	O processo do conjunto de encadeamento foi interrompido por Java, z/OS UNIX ou por um comando do operador. Os outros campos de status mostram os últimos valores conhecidos. Use a opção <b>CLEANUP</b> do comando de modificação <b>DISPLAY PROCESS</b> para remover essa entrada da tabela.
*tempo limite*	O processo do conjunto de encadeamento não respondeu de maneira adequada ao daemon RSE durante uma solicitação de conexão do cliente. Os outros campos de status mostram os valores atuais. O conjunto de encadeamento é excluído para solicitações futuras de conexão do cliente. O status *tempo limite* é redefinido quando um cliente atendido por este conjunto de encadeamento efetua logoff.

Informações adicionais são fornecidas quando a opção **DETAIL** do comando de modificação do **DISPLAY PROCESS** é usado:

```
ID do processo(33555087) ASId(002E) Nome da Tarefa (RSED8) Ordem(1)
PROCESS LIMITS:  CURRENT  HIGHWATER  LIMIT
JAVA HEAP USAGE(%)    10        56        100
CLIENTS                0         25         60
MAXFILEPROC            83        103       64000
MAXPROCUSER            97         99        200
MAXTHREADS              9         14       1500
MAXTHREADTASKS         9         14       1500
```

O campo ASId é o ID do espaço de endereço, em nota hexadecimal. A tabela de limites do processo mostra o uso do recurso atual, o limite

máximo para o uso do recurso e o limite de recurso. Note que devido a outros fatores de limitação, o limite definido pode nunca ser alcançado.

#### **CANCEL ID=clientid**

Cancela uma conexão do cliente com base no ID do cliente, que é mostrado no comando **DISPLAY CLIENT** modify.

#### **CANCEL USER=userid**

Cancela uma conexão do cliente com base no ID de usuário cliente, que é mostrado no comando modify **DISPLAY CLIENT**.

#### **RSECOMMLOG {ON,OFF,I,W,E,2,1,0}**

Controla o nível de rastreo do servidor RSE (rsecomm.log) e os serviços do conjunto de dados do MVS (lock.log e ffs\*.log). O padrão de inicialização é definido em rsecomm.properties. Existem três níveis de detalhes disponíveis:

E ou 0 ou OFF	Mensagens de erro apenas.
W ou 1	Mensagens de erro e de aviso. Essa é a configuração padrão em rsecomm.properties.
I ou 2 ou ON	Mensagens de erro, de aviso e informativas.

O rastreo detalhada prejudica o desempenho e deverá ser feito apenas com orientação do centro de suporte IBM.

#### **RSEDAEMONLOG {ON,OFF,I,E,2,0}**

Controla o nível de detalhes do rastreo do daemon RSE (rsedaemon.log). O padrão de inicialização é definido em rsecomm.properties. Existem dois níveis de detalhes disponíveis:

E ou 0 ou OFF	Mensagens de erro apenas.
I ou 2 ou ON	Mensagens de erro, de aviso e informativas.

O rastreo detalhada prejudica o desempenho e deverá ser feito apenas com orientação do centro de suporte IBM.

#### **RSESERVERLOG {ON,OFF,I,E,2,0}**

Controla o nível de detalhes do rastreo dos conjuntos de encadeamento RSE (rseserver.log). O padrão de inicialização é definido em rsecomm.properties. Existem dois níveis de detalhes disponíveis:

E ou 0 ou OFF	Mensagens de erro apenas.
I ou 2 ou ON	Mensagens de erro, de aviso e informativas.

O rastreo detalhada prejudica o desempenho e deverá ser feito apenas com orientação do centro de suporte IBM.

#### **RSESTANDARDLOG {ON,OFF}**

Desativa (OFF) ou ativa (ON) a atualização dos arquivos de log mantidos nos fluxos stdout e stderr dos conjuntos de encadeamento (stdout\*.log e stderr\*.log). O padrão de inicialização é definido pela diretiva enable.standard.log em rsed.envvars.

O rastreo detalhada prejudica o desempenho e deverá ser feito apenas com orientação do centro de suporte IBM.

## SWITCH

Alterna para um novo arquivo de log de auditoria.

### Nota:

- Consulte “Arquivos de Log” na página 136 no Capítulo 9, “Resolução de Problemas de Configuração”, na página 135 para obter informações adicionais sobre os arquivos de log mencionados acima.
- Consulte “Criação de Log de Auditoria” na página 163 no Capítulo 10, “Considerações sobre Segurança”, na página 157 para obter informações adicionais sobre auditoria.

## Daemon de Bloqueio

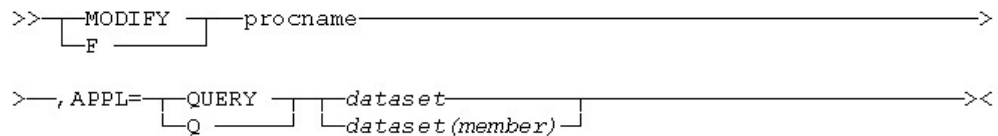


Figura 38. Comando do operador MODIFY LOCKD

### procname

O nome do membro em uma biblioteca de procedimentos que foi usada para iniciar o servidor. O nome padrão usado durante a configuração do host é LOCKD .

### QUERY dataset[(member)]

Consulte o status de bloqueio do conjunto de dados ou membro listado. O servidor responderá com uma das mensagens a seguir:

```
BPXM023I (stclock) dataset[(member)] NOT LOCKED
BPXM023I (stclock) dataset[(member)] LOCKED BY userid
```

### Nota:

- O servidor também relatará bloqueios de relatórios mantidos por outros produtos, como o ISPF.
- Os bloqueios mantidos pelos clientes do Developer para System z que não conseguiram se registrar com o daemon de bloqueio resultarão no espaço de endereço do servidor do conjunto de encadeamento (RSEDx) que está sendo relatado como o proprietário do bloqueio.

A mensagem do console FEK513W é gerada quando o servidor RSE não consegue registrar o cliente com o daemon de bloqueio. Os valores ASID e TCB mencionados nesta mensagem podem ser comparados em relação à saída do comando do operador **D GRS,RES=(\*,dataset[(member))]** para localizar o usuário real que está mantendo o bloqueio.

---

## Stop (P)

Use o comando **STOP** para parar uma tarefa ativa. A versão abreviada do comando é a letra P.

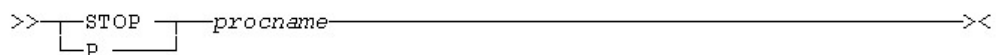


Figura 39. Comando do operador STOP

**procname**  
 O nome do membro em uma biblioteca de procedimentos que foi usada para iniciar o servidor. Os nomes padrão usados durante a configuração do host são JMON, RSED e LOCKD para o JES Job Monitor, o daemon RSE e o daemon de bloqueio, respectivamente.

## Mensagens do Console

### JES Job Monitor

O JES Job Monitor não possui mensagens de console específicas de um produto. O servidor conta com o z/OS e o JES para gerar mensagens do console para ações realizadas por clientes do Developer para System z.

### Daemon RSE, Servidor do Conjunto de Encadeamento RSE e Daemon de Bloqueio

A Tabela 19 lista as mensagens do console específicos de um produto geradas pelo daemon RSE, pelo servidor do conjunto de encadeamentos RSE e pelo daemon de bloqueio.

Tabela 19. Mensagens de Console do RSE

ID da Mensagem	Texto de mensagem
FEK001I	RseDaemon sendo inicializado em modo bit {0}
FEK002I	RseDaemon iniciado. (porta={0})
FEK003I	Pára o comando que está sendo processado
FEK004I	RseDaemon: Tamanho Máximo de Heap={0}MB e Tamanho do AS privado={1}MB
FEK005I	Processo do servidor iniciado. (processId={0})
FEK009I	RseDaemon está aguardando o processo do servidor iniciar.
FEK010I	(local do rsed.envvars = {0})
FEK011I	(diretório do log = {0})
FEK100E	O valor da porta/tempo limite do daemon deve ser dígitos
FEK101E	JRE {0} ou superior necessário
FEK102E	Argumentos inválidos recebidos: {0}
FEK103E	Disco Quase Cheio em {0}
FEK104E	O número máximo de processos foi atingido
FEK105E	Erro ao enviar dados de auditoria (rc={0})
FEK110E	Falha de socket(). razão=({0})
FEK111E	Falha de setsockopt(). razão=({0})
FEK112E	Falha de bind(). razão=({0})
FEK113E	Falha de listen(). razão=({0})
FEK114E	Falha de accept(). razão=({0})
FEK115E	Falha de write(). razão=({0})

Tabela 19. Mensagens de Console do RSE (continuação)

ID da Mensagem	Texto de mensagem
FEK116E	Falha de pipe(). razão={0}
FEK117E	Falha de socketpair(). razão={0}
FEK118E	Falha de select(). razão={0}
FEK119E	Falha de _console(). razão={0}
FEK130E	Falha de gsk_environment_open(). razão={0}
FEK131E	Falha de gsk_attribute_set_enum(GSK_PROTOCOL_SSLV2). razão={0}
FEK132E	Falha de gsk_attribute_set_enum(GSK_PROTOCOL_SSLV3). razão={0}
FEK133E	Falha de gsk_attribute_set_enum(GSK_PROTOCOL_TL SV1). razão={0}
FEK134E	Falha de gsk_attribute_set_buffer(GSK_KEYRING_FILE). razão={0}
FEK135E	Falha de gsk_attribute_set_buffer(GSK_KEYRING_PW). razão={0}
FEK136E	Falha de gsk_environment_init(). razão={0}
FEK137E	Falha de gsk_secure_socket_open(). razão={0}
FEK138E	Falha de gsk_attribute_set_numeric_value(GSK_FD). razão={0}
FEK139E	Falha de gsk_attribute_set_buffer(GSK_KEYRING_LABEL). razão={0}
FEK140E	Falha de gsk_attribute_set_enum(GSK_SESSION_TYPE). razão={0}
FEK141E	Falha de gsk_attribute_set_callback(GSK_IO_CALLBACK). razão={0}
FEK142E	Falha de gsk_secure_socket_init(). razão={0}
FEK143E	Falha de gsk_attribute_set_enum(GSK_CLIENT_AUTH_TYPE). razão={0}
FEK144E	Falha de gsk_get_cert_info. razão={0}
FEK145E	Falha de gsk_secure_socket_read(). razão={0}
FEK146E	Falha de gsk_secure_socket_write(). razão={0}
FEK150E	RseDaemon encerrado de forma anormal; {0}
FEK201I	O comando {0} foi processado
FEK202E	Comando Inválido Fornecido
FEK203E	Comando de Exibição Inválido: Exibir Processo   Cliente
FEK204E	Comando de Cancelamento Inválido: Cancelar ID=   Usuário=
FEK205E	O comando não foi processado devido a ALTERNAÇÕES consecutivas
FEK206E	O Recurso de Log de Auditoria não está ativo
FEK207I	Nenhum Cliente a ser exibido
FEK208I	{0} cancelado
FEK209I	Nenhum Processo a ser exibido
FEK210I	{0} cancelado devido a logon duplicado
FEK501I	Daemon de bloqueio iniciado, porta={0}, intervalo de limpeza={1}, nível de log={2}
FEK502I	Daemon de bloqueio sendo encerrado
FEK510E	Daemon de bloqueio, porta ausente
FEK511E	Daemon de bloqueio, porta errada, porta={0}

Tabela 19. Mensagens de Console do RSE (continuação)

ID da Mensagem	Texto de mensagem
FEK512E	Daemon de bloqueio, erro de soquete, porta={0}
FEK513W	Daemon de bloqueio, falha de registro, ASID={0}, TCB={1}, USER={2}
FEK514W	Daemon de bloqueio, nível de log errado, nível de log={0}
BPXM023I	(stclock) dataset[(member)] NOT LOCKED
BPXM023I	(stclock) dataset[(member)] LOCKED BY userid
BPXM023I	Comando (stclock), WRONG COMMAND
BPXM023I	Comando (stclock), MISSING ARGUMENT
BPXM023I	Argumento (stclock), WRONG ARGUMENT

## Como Ler um Diagrama de Sintaxe

O diagrama de sintaxe mostra como especificar um comando para que o sistema operacional possa interpretar corretamente o que você digitar. Leia o diagrama de sintaxe da esquerda para a direita e de cima para baixo, seguindo a linha horizontal (o caminho principal).

### Símbolos

Os símbolos a seguir são usados em diagramas de sintaxe:

Símbolo	Descrição
>>	Marca o início do diagrama de sintaxe.
>	Indica que o diagrama de sintaxe continua.
	Marca o início e o fim de um fragmento ou parte do diagrama de sintaxe.
><	Marca o fim do diagrama de sintaxe.

### Operandos

Os tipos de operandos a seguir são usados em diagramas de sintaxe:

- Os operandos obrigatórios são exibidos na linha de caminho principal:
- Os operandos opcionais são exibidos abaixo da linha de caminho principal:

```
>>—REQUIRED_OPERAND—><
>>└─OPTIONAL_OPERAND─┘><
```

- Os operandos padrão são exibidos acima da linha de caminho principal:

```
>>└─DEFAULT_OPERAND─┘><
```

Os operandos são classificados como palavras-chave ou variáveis:

- As palavras-chave são constantes que devem ser fornecidas. Se a palavra-chave aparecer no diagrama de sintaxe com letras maiúsculas e minúsculas, a parte de minúsculas será a abreviação para a palavra-chave (por exemplo, KEYword). As palavras-chave não fazem distinção entre maiúsculas e minúsculas. Você pode codificá-las em maiúsculas ou minúsculas.
- As variáveis estão em itálico, aparecem em letras minúsculas e representam nomes ou valores fornecidos. Por exemplo, um nome do conjunto de dados é uma variável. As variáveis podem fazer distinção entre maiúsculas e minúsculas.

## Exemplo de Sintaxe

No exemplo a seguir, o comando USER é uma palavra-chave. O parâmetro variável necessário é user\_id e o parâmetro variável opcional é password. Substitua os parâmetros de variáveis pelos seus próprios valores:

```
>>—USER—user_id—┐—————><
                   └password┘
```

## Caracteres Não Alfanuméricos e Espaços em Branco

Se um diagrama mostrar um caractere não alfanumérico (como parênteses, pontos finais, vírgulas, sinais de igual e espaços em branco), você deverá codificar o caractere como parte da sintaxe. Neste exemplo, você deve codificar OPERAND=(001 0.001):

```
>>—OPERAND—==(—001— —0.001—)—————><
```

## Selecionando Mais de Um Operando

Uma seta para a esquerda em um grupo de operandos significa que mais de um operando pode ser selecionado, ou apenas um pode ser repetido:

```
>>—┐—————><
    └REPEATABLE_OPERAND_1┘
    └REPEATABLE_OPERAND_2┘
    <—
```

## Mais Longo que Uma Linha

Se um diagrama for mais longo que uma linha, a primeira linha terminará com uma ponta de seta única e a segunda linha começará com uma ponta de seta única:

```
>>—| A primeira linha de um diagrama de sintaxe que é mais longo que uma linha |—>
>—| A continuação dos subcomandos, parâmetros ou ambos |—————><
```

## Fragmentos de Sintaxe

Alguns diagramas podem conter fragmentos de sintaxe, que servem para dividir diagramas que são muito longos, complexos ou repetitivos. Os nomes dos fragmentos de sintaxe são compostos por letras maiúsculas e minúsculas e são mostrados no diagrama e no título do fragmento. O fragmento é colocado abaixo do diagrama principal:

```
>>—| Fragmento de sintaxe |—————><
```

Fragmento de sintaxe:

```
|—1ST_OPERAND—,—2ND_OPERAND—,—3RD_OPERAND—|
```





---

## Capítulo 9. Resolução de Problemas de Configuração

Este capítulo é fornecido para ajudá-lo com alguns problemas comuns que você pode encontrar durante a configuração do Developer para System z e contém as seguintes seções:

- “Análise de Log e de Configuração Usando FEKLOGS”
- “Arquivos de Log” na página 136
- “Arquivos de Dump” na página 141
- “Rastreio” na página 143
- “Bits de Permissão do z/OS UNIX” na página 146
- “Portas TCP/IP Reservadas” na página 149
- “Tamanho do Espaço de Endereço” na página 150
- “Transação APPC e Serviço de Comandos do TSO” na página 152
- “Informações Diversas” na página 153

Informações adicionais estão disponíveis na seção Suporte do Web site do Developer para System z (<http://www-306.ibm.com/software/awdtools/rdz/support/>) onde é possível localizar as Notas Técnicas que trazem as informações mais recentes de nossa equipe de suporte.

Na seção Biblioteca do Web site (<http://www-306.ibm.com/software/awdtools/rdz/library/>), você também pode localizar a versão mais recente da documentação do Developer para System z, incluindo whitepapers.

O Developer para System z Centro de Informações do (<http://publib.boulder.ibm.com/infocenter/ratdevz/v7r6/index.jsp>) documenta o cliente Developer para System z e como ele interage com o host (a partir de uma perspectiva do cliente’).

Informações sobre valor também podem ser localizadas na biblioteca do z/OS na Internet, disponível em <http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/>.

Notifique-nos se achar que o Developer para System z perdeu uma determinada função. Você pode abrir uma Solicitação para Aprimoramento (RFE) em <https://www.ibm.com/developerworks/support/rational/rfe/>

---

### Análise de Log e de Configuração Usando FEKLOGS

O Developer para System z fornece uma tarefa de amostra, FEKLOGS, que reúne todos os arquivos de log z/OS UNIX, bem como as informações de instalação e de configuração do Developer para System z.

A tarefa de amostra FEKLOGS está localizada em FEK.#CUST.JCL, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes.

A customização de FEKLOGS é descrita dentro da JCL. A customização inclui a provisão de algumas variáveis principais.

**Nota:** Os clientes SDSF podem usar o comando da linha **XDC** em SDSF para salvar a saída de tarefa em um conjunto de dados, que, por sua vez, pode ser fornecido ao centro de suporte IBM.

---

## Arquivos de Log

O Developer para 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 de arquivos de log que podem ser criados no sistema host do z/OS. Ao lado desses logs específicos do produto, assegure-se de marcar o SYSLOG de todas as mensagens relacionadas.

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:

- userlog/\$LOGNAME/

Os arquivos de log específicos do usuário estão localizados em userlog/\$LOGNAME/, em que userlog é o valor combinado das diretivas user.log e DSTORE\_LOG\_DIRECTORY em rsed.envvars, e \$LOGNAME é o ID do usuário de login (em maiúsculas). Se a diretiva user.log for comentada ou não estiver presente, o caminho inicial do usuário será usado. O caminho inicial é definido no segmento de segurança OMVS do ID de usuário. Se a diretiva DSTORE\_LOG\_DIRECTORY for comentada ou não estiver presente, então .eclipse/RSE/ será anexado ao valor user.log.

- .dstoreMemLogging - Criação de log de uso de memória do armazenamento de dados
- .dstoreTrace - Criação de log de ação do armazenamento de dados
- fa.log - O log do Fault Analyzer Integration
- fekfivpi.log - O log do teste IVP do fekfivpi
- fekfivps.log - O log do teste IVP do fekfivps
- ffs.log - O log do servidor Foreign File System (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
- ffspout.log - O log do gravador de arquivos, que grava um conjunto de dados sequencial ou um membro PDS
- lock.log - O log do gerenciador de bloqueios, que bloqueia/desbloqueia um conjunto de dados sequencial 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 manipula os comandos do cliente e a criação de logs de comunicação de todos os serviços que dependem do RSE (pode conter rastreamento de pilha de exceção Java)
- stderr.log - Os dados direcionados da saída de erro padrão stderr
- stdout.log - Os dados redirecionados de stdout, saída padrão

**Nota:** O diretório .eclipse e os arquivos de log .dstore\* começam com um ponto (.), o que os torna ocultos. Use o comando **ls -lA** do z/OS UNIX para listar arquivos e diretórios ocultos. Ao utilizar o cliente do Developer para System z, selecione a página de preferências **Janela > Preferências... > Sistemas Remotos > Arquivos** e ative “Mostrar Arquivos Ocultos”.

- daemon-home

Os arquivos de log específicos do daemon RSE e do conjunto de encadeamentos do RSE estão localizados em `daemon-home`, em que `daemon-home` é o valor da diretiva `daemon.log` em `rsed.envvars`. Se a diretiva `daemon.log` for comentada ou não estiver presente, o diretório inicial do ID do usuário designado à tarefa iniciada RSED será usado. O diretório inicial é definido no segmento de segurança OMVS do ID do usuário.

- `rsedaemon.log` - O log do daemon RSE
- `rseserver.log` - O log dos conjuntos de encadeamento RSE
- `audit.log` - A trilha de auditoria do RSE
- `serverlogs.count` - Contador para criação de log de fluxos do conjunto de encadeamento RSE
- `stderr.*.log` - Fluxo de erro padrão do conjunto de encadeamento RSE
- `stdout.*.log` - Fluxo de saída padrão do conjunto de encadeamento RSE

**Nota:** Há comandos do operador disponíveis para controlar a quantidade de dados gravados em alguns dos arquivos de log mencionados. Consulte o Capítulo 8, “Comandos do Operador”, na página 123 para obter informações adicionais.

## Criação de Logs do JES Job Monitor

- **SYSOUT DD**

Criação de logs de operações normais. O valor padrão na amostra de JCL `FEK.#CUST.PROCLIB(JMON)` é `SYSOUT=*`.

- **SYSPRINT DD**

Criação de logs de rastreamento. O valor padrão na amostra de JCL `FEK.#CUST.PROCLIB(JMON)` é `SYSOUT=*`. O rastreamento é ativado com o parâmetro `-TV`; consulte “Rastreamento do JES Job Monitor” na página 143 para obter mais detalhes.

## Criação de Log do Daemon de Bloqueio

- **STDOUT DD**

Os dados redirecionados de `stdout`, saída padrão Java. O valor padrão na amostra JCL `FEK.#CUST.PROCLIB(LOCKD)` é `SYSOUT=*`.

- **STDERR DD**

Os dados redirecionados de `stderr`, saída de erro padrão Java. O valor padrão na amostra JCL `FEK.#CUST.PROCLIB(LOCKD)` é `SYSOUT=*`.

## Criação de Log de Daemon RSE e de Conjunto de Encadeamento

- **STDOUT DD**

Os dados redirecionados de `stdout`, saída padrão Java de daemon RSE. O valor padrão no JCL da amostra `FEK.#CUST.PROCLIB(RSED)` é `SYSOUT=*`.

- **STDERR DD**

Os dados redirecionados de `stderr`, saída de erro padrão Java do daemon RSE. O valor padrão no JCL da amostra `FEK.#CUST.PROCLIB(RSED)` é `SYSOUT=*`.

- **daemon-home**

Os arquivos de log específicos do daemon RSE e do conjunto de encadeamentos do RSE estão localizados em `daemon-home`, em que `daemon-home` é o valor da diretiva `daemon.log` em `rsed.envvars`. Se a diretiva `daemon.log` for comentada ou

não estiver presente, o diretório inicial do ID do usuário designado à tarefa iniciada RSED será usado. O diretório inicial é definido no segmento de segurança OMVS do ID do usuário.

- `rsedaemon.log` - O log do daemon RSE
- `rseserver.log` - O log dos conjuntos de encadeamento RSE
- `audit.log` - A trilha de auditoria do RSE
- `serverlogs.count` - Contador para criação de log de fluxos do conjunto de encadeamento RSE
- `stderr.*.log` - Fluxo de erro padrão do conjunto de encadeamento RSE
- `stdout.*.log` - Fluxo de saída padrão do conjunto de encadeamento RSE

**Nota:**

- `serverlogs.count`, `stderr.*.log` e `stdout.*.log` são criados apenas se a diretiva `enable.standard.log` em `rsed.envvars` estiver ativa ou se a função for dinamicamente ativada com o comando do operador **`modify rsestandardlog on`**.
- O \* em `stderr.*.log` e em `stdout.*.log` significa 1 por padrão. Entretanto, pode existir vários conjuntos de encadeamento RSE e, nesse caso, o número é aumentado para cada novo conjunto de encadeamento RSE para garantir nomes de arquivos exclusivos.
- Não haverá arquivos de log `stdout.log` e `stderr.log` específicos do usuário quando a diretiva `enable.standard.log` estiver ativa. Os dados específicos do usuário são gravados agora no fluxo do conjunto de encadeamento do RSE correspondente.
- Há comandos do operador disponíveis para controlar a quantidade de dados gravados em alguns dos arquivos de log mencionados. Consulte o Capítulo 8, “Comandos do Operador”, na página 123 para obter informações adicionais.

## Criação de Log de Usuário do RSE

- `userlog/$LOGNAME/`

Há vários arquivos de log criados pelos componentes relacionados ao RSE. Todos estão localizados em `userlog/$LOGNAME/`, em que `userlog` é o valor combinado das diretivas `user.log` e `DSTORE_LOG_DIRECTORY` em `rsed.envvars`, e `$LOGNAME` é o ID do usuário de logon (em maiúsculas). Se a diretiva `user.log` for comentada ou não estiver presente, o caminho inicial do usuário será usado. O caminho inicial é definido no segmento de segurança OMVS do ID de usuário. Se a diretiva `DSTORE_LOG_DIRECTORY` for comentada ou não estiver presente, então `.eclipse/RSE/` será anexado ao valor `user.log`.

- `.dstoreMemLogging` - Criação de log de uso de memória do armazenamento de dados
- `.dstoreTrace` - Criação de log de ação do armazenamento de dados
- `ffs.log` - O log do servidor Foreign File System (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 sequencial ou um membro PDS
- `lock.log` - O log do gerenciador de bloqueio, que bloqueia ou desbloqueia um conjunto de dados sequencial 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 manipula os comandos do cliente e a criação de logs de comunicação de todos os serviços que dependem do RSE (pode conter rastreamento de pilha de exceção Java)
- `stderr.log` - Os dados redirecionados de `stderr`, saída de erro padrão
- `stdout.log` - Os dados redirecionados de `stdout`, saída padrão

**Nota:**

- O diretório `.eclipse` e os arquivos de log `.dstore*` começam com um ponto (`.`), o que os torna ocultos. Use o comando `ls -lA` do z/OS UNIX para listar arquivos e diretórios ocultos. Ao utilizar o cliente do Developer para System z, selecione a página de preferências **Janela > Preferências... > Sistemas Remotos > Arquivos** e ative “Mostrar Arquivos Ocultos”.
- A criação dos arquivos de log `.dstore*` é controlada pelas opções de inicialização `-DDSTORE_*` Java, conforme descrito em “Definindo Parâmetros Extras de Inicialização Java com `_RSE_JAVAOPTS`” na página 42.
- Os arquivos de log `.dstore*` são criados em ASCII. Use o comando z/OS UNIX `iconv -f ISO8859-1 -t IBM-1047 .dstore*` para exibi-los em EBCDIC (ao usar a página de códigos IBM-1047).
- Não haverá arquivos de log `stdout.log` e `stderr.log` específicos do usuário quando a diretiva `enable.standard.log` estiver ativa. Os dados específicos do usuário são gravados agora no fluxo do conjunto de encadeamento do RSE correspondente.
- Há comandos do operador disponíveis para controlar a quantidade de dados gravados em alguns dos arquivos de log mencionados. Consulte o Capítulo 8, “Comandos do Operador”, na página 123 para obter informações adicionais.

## Criação de Log de Integração do Fault Analyzer

- `userlog/$LOGNAME/`

A criação de log do Fault Analyzer Integration, em que `userlog` é o valor combinado das diretivas `user.log` e `DSTORE_LOG_DIRECTORY` em `rsed.envvars`, e `$LOGNAME` é o ID do usuário de logon (em maiúsculas). Se a diretiva `user.log` for comentada ou não estiver presente, o caminho inicial do usuário será usado. O caminho inicial é definido no segmento de segurança OMVS do ID de usuário. Se a diretiva `DSTORE_LOG_DIRECTORY` for comentada ou não estiver presente, então `.eclipse/RSE/` será anexado ao valor `user.log`.

- `fa.log` - O log do Fault Analyzer Integration
- `rsecomm.log` - Criação de logs de comunicação do Fault Analyzer Integration

## Criação de Log de Integração do File Manager

- `userlog/$LOGNAME/rsecomm.log`

A criação de log de comunicação do File Manager Integration, em que `userlog` é o valor combinado das diretivas `user.log` e `DSTORE_LOG_DIRECTORY` em `rsed.envvars`, e `$LOGNAME` é o ID do usuário de logon (em maiúsculas). Se a diretiva `user.log` for comentada ou não estiver presente, o caminho inicial do usuário será usado. O caminho inicial é definido no segmento de segurança OMVS do ID de usuário. Se a diretiva `DSTORE_LOG_DIRECTORY` for comentada ou não estiver presente, então `.eclipse/RSE/` será anexado ao valor `user.log`.

## Criação de Log do SCLM Developer Toolkit

- **userlog/\$LOGNAME/rsecomm.log**

A criação de log de comunicação do SCLM Developer Toolkit, em que userlog é o valor combinado das diretivas user.log and DSTORE\_LOG\_DIRECTORY em rsed.envvars, e \$LOGNAME é o ID do usuário de logon (em maiúsculas). Se a diretiva user.log for comentada ou não estiver presente, o caminho inicial do usuário será usado. O caminho inicial é definido no segmento de segurança OMVS do ID de usuário. Se a diretiva DSTORE\_LOG\_DIRECTORY for comentada ou não estiver presente, então .eclipse/RSE/ será anexado ao valor user.log.

## Criação de Logs do CARMA

- **Tarefa do Servidor CARMA**

Ao abrir uma conexão com o CARMA e usar a interface em lote, FEK.#CUST.SYSPROC(CRASUBMT) iniciará uma tarefa do servidor (com o ID do usuário como proprietário) denominada CRAport, em que port é a porta TCP/IP usada.

- **CARMALOG DD**

Se a instrução DD CARMALOG for especificada no método de inicialização do CARMA escolhido, a criação de logs do CARMA será redirecionada para essa instrução DD na tarefa do servidor, caso contrário, ela irá para SYSPRINT.

- **SYSPRINT DD**

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

- **userlog/\$LOGNAME/rsecomm.log**

A criação de log de comunicação de CARMA, em que userlog é o valor combinado das diretivas user.log e DSTORE\_LOG\_DIRECTORY em rsed.envvars, e \$LOGNAME é o ID do usuário de logon (em maiúsculas). Se a diretiva user.log for comentada ou não estiver presente, o caminho inicial do usuário será usado. O caminho inicial é definido no segmento de segurança OMVS do ID de usuário. Se a diretiva DSTORE\_LOG\_DIRECTORY for comentada ou não estiver presente, então .eclipse/RSE/ será anexado ao valor user.log.

## 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 da JCL. A criação de logs para mensagens desta fase é controlada pela instrução SYSPRINT DD para o utilitário ATBSDFMU. O valor padrão na amostra de JCL FEK.SFEKSAMP(FEKAPPCC) 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 na amostra de JCL FEK.SFEKSAMP(FEKAPPCC) 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 sobre isso.

## Criação de Log de Teste IVP do fekfivpi

- `userlog/$LOGNAME/fekfivpi.log`

Saída do comando `fekfivpi -file` (teste IVP relacionado ao TSO/ISPF Client Gateway), em que `userlog` é o valor combinado das diretivas `user.log` e `DSTORE_LOG_DIRECTORY` em `rsed.envvars`, e `$LOGNAME` é o ID do usuário de logon (em maiúsculas). Se a diretiva `user.log` for comentada ou não estiver presente, o caminho inicial do usuário será usado. O caminho inicial é definido no segmento de segurança OMVS do ID de usuário. Se a diretiva `DSTORE_LOG_DIRECTORY` for comentada ou não estiver presente, então `.eclipse/RSE/` será anexado ao valor `user.log`.

## Criação de Log de Teste IVP do fekfivps

- `userlog/$LOGNAME/fekfivps.log`

Saída do comando `fekfivps -file` (teste IVP relacionado ao SCLMDT), em que `userlog` é o valor combinado das diretivas `user.log` e `DSTORE_LOG_DIRECTORY` em `rsed.envvars`, e `$LOGNAME` é o ID do usuário de logon (em maiúsculas). Se a diretiva `user.log` for comentada ou não estiver presente, o caminho inicial do usuário será usado. O caminho inicial é definido no segmento de segurança OMVS do ID de usuário. Se a diretiva `DSTORE_LOG_DIRECTORY` for comentada ou não estiver presente, então `.eclipse/RSE/` será anexado ao valor `user.log`.

---

## 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 o local desses dumps dependem 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 a JCL em busca das seguintes instruções DD (dependendo do produto):

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

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

## Dumps de Java

No z/OS UNIX, a maioria dos dumps do Developer para System z é controlada pela Java Virtual Machine (JVM).

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 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 usada para determinar quais agentes de dump existem na conclusão da inicialização.

Os tipos de dump que podem ser produzidos são os seguintes:

### 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 sequencial utilizando-se um nome padrão no formato `%uid.JVM.TDUMP.%job.D%ym%d.T%H%M%S`, ou conforme 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`.

**Nota:** `JAVA_DUMP_TDUMP_PATTERN` permite o uso de variáveis, que são convertidas em um valor real na hora em que o dump de transação é obtido.

*Tabela 20. Variáveis de `JAVA_DUMP_TDUMP_PATTERN`*

Variável	Uso
<code>%uid</code>	ID do usuário
<code>%job</code>	Nome da tarefa
<code>%y</code>	Ano (2 dígitos)
<code>%m</code>	Mês (2 dígitos)
<code>%d</code>	Dia (2 dígitos)
<code>%H</code>	Hora (2 dígitos)
<code>%M</code>	Minuto (2 dígitos)
<code>%S</code>	Segundo (2 dígitos)

### 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.aaaamdd.hhmmss.pid`, em que `aaaamdd` 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 do Dump do z/OS UNIX” na página 143.

### 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 do Dump do z/OS UNIX”.

### 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 do Dump do z/OS UNIX”.

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

## Locais do Dump do z/OS UNIX

A JVM verifica a existência de cada um dos seguintes locais e as permissões de gravação e armazena os arquivos `CEEDUMP`, `HEAPDUMP` e `JAVADUMP` no primeiro local 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 está configurada em `rsed.envvars` como `/tmp`. Ela pode ser alterada para `/dev/null` para evitar a criação de arquivos dump.
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`.

---

## Rastreio

### Rastreio do JES Job Monitor

O rastreio do JES Job Monitor é controlado pelo operador do sistema, conforme descrito no Capítulo 8, “Comandos do Operador”, na página 123.

- Iniciar a tarefa iniciada `JMON` com o parâmetro `PRM=-TV` ativa o modo detalhado (rastreio)
- Os comandos `modify -TV` e `modify -TN` ativam e desativam o rastreio

### Rastreio do RSE

Há vários arquivos de log criados pelos componentes relacionados ao RSE. A maioria está localizada em `userlog/$LOGNAME/`, em que `userlog` é o valor combinado das diretivas `user.log` e `EDSTORE_LOG_DIRECTORY` em `rsed.envvars`, e `$LOGNAME` é o ID do usuário de logon (em maiúsculas). Se a diretiva `user.log` for comentada ou não estiver presente, o caminho inicial do usuário será usado. O caminho inicial é definido no segmento de segurança OMVS do ID de usuário. Se

a diretiva `DSTORE_LOG_DIRECTORY` for comentada ou não estiver presente, então `.eclipse/RSE/` será anexado ao valor `user.log`.

A quantidade de dados gravados em `ffs*.log`, `lock.log` e `rsecomm.log` é controlada pelo comando do operador **modify rsecommlog** ou pela configuração de `debug_level` in `rsecomm.properties`. Consulte Capítulo 8, “Comandos do Operador”, na página 123 e “(Opcional) Rastreo de RSE” na página 94 para obter mais detalhes.

A criação dos arquivos de log `.dstore*` é controlada pelas opções de inicialização `-DDSTORE_*` Java, conforme descrito em “Definindo Parâmetros Extras de Inicialização Java com `_RSE_JVAOPTS`” na página 42.

**Nota:**

- O diretório `.eclipse` e os arquivos de log `.dstore*` começam com um ponto (`.`), o que os torna ocultos. Use o comando `ls -lA` do z/OS UNIX para listar arquivos e diretórios ocultos. Ao utilizar o cliente do Developer para System z, selecione a página de preferências **Janela > Preferências... > Sistemas Remotos > Arquivos** e ative “Mostrar Arquivos Ocultos”.
- Os arquivos de log `.dstore*` são criados em ASCII. Use o comando z/OS UNIX `iconv -f ISO8859-1 -t IBM-1047 .dstore*` para exibi-los em EBCDIC (ao usar a página de códigos IBM-1047).

Os arquivos de log específicos do daemon RSE e do conjunto de encadeamentos do RSE estão localizados em `daemon-home`, em que `daemon-home` é o valor da diretiva `daemon.log` em `rsed.envvars`. Se a diretiva `daemon.log` for comentada ou não estiver presente, o diretório inicial do ID do usuário designado à tarefa iniciada RSED será usado. O diretório inicial é definido no segmento de segurança OMVS do ID do usuário.

A quantidade de dados gravada em `rsedaemon.log` e em `rserver.log` é controlada pelos comandos do operador **modify rsedaemonlog** e **modify rserverlog** ou configurando `debug_level` em `rsecomm.properties`. Consulte Capítulo 8, “Comandos do Operador”, na página 123 e “(Opcional) Rastreo de RSE” na página 94 para obter mais detalhes.

`serverlogs.count`, `stderr*.log` e `stdout*.log` são criados apenas se a diretiva `enable.standard.log` em `rsed.envvars` estiver ativa, ou se a função for dinamicamente ativada com o comando do operador **modify rsestandardlog on**.

## Rastreo do Daemon de Bloqueio

O log específico do daemon de bloqueio está localizado no STDOUT DD da tarefa iniciada LOCKD. A quantidade de dados gravados no log é controlada pelo parâmetro de inicialização LOG. Consulte Capítulo 8, “Comandos do Operador”, na página 123 e “(Opcional) Rastreo de RSE” na página 94 para obter mais detalhes.

## Rastreo do CARMA

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

- Desativar Criação de Log
- Log de Erros
- Log de Avisos

- Log Informativo
- Log de Depuração

O valor padrão é o seguinte:

Log de Erros

Consulte “Arquivos de Log” na página 136 para obter informações adicionais sobre as localizações do arquivo de log.

## 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 neste seção referem-se ao qualificador de alto nível usado durante a instalação do Developer para 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 “Procedimentos de Construção Remota ELAXF\*” na página 25.
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 agora é: EXIT(ADEXIT(''MAXTRACE'',ELAXMGUX)).

3. Execute uma Verificação de Sintaxe Remota no programa COBOL para o qual você deseja rastreo detalhado.
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.SF.Z682746.XML
23 //COBOL.WSEDSF2 DD DISP=MOD,
// DSN=uid.ERRCOB.member.SF.Z682747.XML
```

5. Copie esses quatro conjuntos de dados em seu PC, por exemplo, criando um projeto COBOL local no Developer para System z e incluindo os 4 conjuntos de dados SIDEFILE1->.
6. Copie o log da tarefa do JES completo em seu PC, por exemplo, abrindo a saída de tarefas no Developer para 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.

---

## Bits de Permissão do z/OS UNIX

O Developer para System z requer que o sistema de arquivo z/OS UNIX e alguns arquivos z/OS UNIX tenham certos bits de permissão configurados.

### Atributo de Sistema de Arquivo SETUID

O RSE (Explorador de Sistema Remoto) é o componente Developer para System z que fornece os serviços principais, como de conexão do cliente ao host. Ele deve ter permissão para executar tarefas como criar o ambiente de segurança do usuário.

O sistema de arquivos (HFS ou zFS) em que o Developer para System z está instalado deve ser montado com o bit de permissão SETUID ativado (este é o padrão do sistema). A montagem do sistema de arquivo com o parâmetro NOSETUID impedirá o Developer para System z de criar o ambiente de segurança do usuário e falhará no pedido de conexão.

Use o comando TSO **ISHELL** para listar o status atual do bit SETUID. No painel ISHELL, selecione **Sistemas\_de\_arquivo > 1. Montar tabela...** para listar os sistemas de arquivos montados. O comando da linha **a** mostrará os atributos para o sistema de arquivos selecionado, em que o campo "Ignorar SETUID" deve ser 0.

### Autorização de Controle de Programa

O RSE (Explorador de Sistema Remoto) é o componente Developer para System z que fornece os serviços principais, como de conexão do cliente ao host. Ele deve executar o programa controlado para realizar tarefas como a comutação para o ID do usuário do cliente.

O bit de controle de programa do z/OS UNIX é configurado durante a instalação do SMP/E onde necessário, exceto para a interface Java para seu produto de segurança, conforme documentado no Capítulo 10, "Considerações sobre Segurança", na página 157. Este bit de permissão pode se perder caso você não o tenha preservado durante uma cópia manual dos diretórios Developer para System z.

Os seguintes arquivos do Developer para System z devem ser controlados pelo programa:

- /usr/lpp/rdz/bin/
  - fekfdivp
  - fekfomvs
  - fekfrivp

- /usr/lpp/rdz/lib/
  - fekfdir.dll
  - libfekdcore.so
  - libfekfmain.so
- /usr/lpp/rdz/lib/icuc/
  - libicudata.dll
  - libicudata40.1.dll
  - libicudata40.dll
  - libicudata64.40.1.dll
  - libicudata64.40.dll
  - libicudata64.dll
  - libicuuc.dll
  - libicuuc40.1.dll
  - libicuuc40.dll
  - libicuuc64.40.1.dll
  - libicuuc64.40.dll
  - libicuuc64.dll

**Nota:** Os arquivos libicu\*64.\* somente estão presentes se você tiver aplicado o PTF Developer para System z que endereça o APAR AM07305 para ativar o suporte 64 bits.

Use o comando z/OS UNIX **ls -E** para listar os atributos estendidos, em que o bit de controle de programa está marcado com a letra p, conforme mostrado na seguinte amostra (\$ é o prompt do z/OS UNIX):

```
$ cd /usr/lpp/rdz
$ ls -E lib/fekf*
-rwxr-xr-x -ps- 2 user      group      94208 Jul  8 12:31 lib/fekfdir.dll
```

Use o comando **extattr +p** do z/OS UNIX para configurar o bit de controle de programa manualmente, conforme exibido na seguinte amostra (\$ e # são o prompt do z/OS UNIX):

```
$ cd /usr/lpp/rdz
$ su
# extattr +p lib/fekf*
# exit
$ ls -E lib/fekf*
-rwxr-xr-x -ps- 2 user      group      94208 Jul  8 12:31 lib/fekfdir.dll
```

**Nota:** Para 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 ou ser um superusuário (UID 0) se esse perfil não estiver definido. Para obter mais informações, consulte *UNIX System Services Planning* (GA22-7800).

## Autorização APF

O Remote Systems Explorer (RSE) é o componente do Developer para System z que fornece serviços principais como conectar o cliente ao host. Ele deve executar o APF autorizado para realizar tarefas como exibir uso do recurso do processo detalhado.

O z/OS UNIX APF bit é configurado durante a instalação do SMP/E, onde necessário. Este bit de permissão pode se perder caso você não o tenha preservado durante uma cópia manual dos diretórios do Developer para System z.

Os seguintes arquivos do Developer para System z devem ter autorização APF:

- /usr/lpp/rdz/bin/
  - fekfomvs
  - fekfrivp

Use o comando do z/OS UNIX **ls -E** para listar os atributos estendidos, em que o APF bit é marcado com a letra a, como mostrado na seguinte amostra (\$ é o prompt do z/OS UNIX):

```
$ cd /usr/lpp/rdz
$ ls -E bin/fekfrivp
-rwxr-xr-x  aps-  2 usuário      grupo      114688 Sep 17 06:41 bin/fekfrivp
```

Use o comando do z/OS UNIX **extattr +a** para configurar o APF bit manualmente, como mostrado na seguinte amostra (\$ e # são os prompts do z/OS UNIX):

```
$ cd /usr/lpp/rdz
$ su
# extattr +a bin/fekfrivp
# exit
$ ls -E bin/fekfrivp
-rwxr-xr-x  aps-  2 usuário      grupo      114688 Sep 17 06:41 bin/fekfrivp
```

**Nota:** Para que seja possível usar o comando **extattr +a**, você deve ter ao menos o acesso de LEITURA no perfil BPX.FILEATTR.APF na classe FACILITY do seu software de segurança ou ser um superusuário (UID 0), caso este perfil não esteja definido. Para obter informações adicionais, consulte *Planejamento de Serviços do Sistema UNIX* (GA22-7800).

## Sticky Bit

Alguns dos serviços opcionais do Developer para System z requerem que os módulos de carregamento do MVS estejam disponíveis para o z/OS UNIX. Isso é feito ao criar um stub (um arquivo fictício) no z/OS UNIX com o "sticky" bit ativado. Quando o stub é executado, o z/OS UNIX procurará um módulo de carregamento MVS com o mesmo nome e executará o módulo de carregamento no lugar.

O sticky bit do z/OS UNIX é configurado durante a instalação do SMP/E onde necessário. Esses bits de permissão podem ser perdidos se você não os preservou durante uma cópia manual dos diretórios do Developer para System z.

Os seguintes arquivos do Developer para System z devem ter o sticky bit ativado:

- /usr/lpp/rdz/bin/
  - BWBTSOW
  - CRASTART

Use o comando **ls -l** do z/OS UNIX para listar as permissões, em que o sticky bit é marcado com a letra t, conforme exibido na seguinte amostra (\$ é o prompt do z/OS UNIX):

```
$ cd /usr/lpp/rdz
$ ls -l bin/CRA*
-rwxr-xr-t  2 user      group      71 Jul  8 12:31 bin/CRASTART
```

Use o comando z/OS UNIX **chmod +t** para configurar o sticky bit manualmente, conforme mostrado na seguinte amostra (\$ e # são o prompt do z/OS UNIX):

```
$ cd /usr/lpp/rdz
$ su
# chmod +t bin/CRA*
# exit
$ ls -l bin/CRA*
-rwxr-xr-t  2 user      group          71 Jul  8 12:31 bin/CRASTART
```

**Nota:** Para poder utilizar o comando **chmod**, você deve ter pelo menos o acesso READ ao perfil SUPERUSER.FILESYS.CHANGEPERMS na classe UNIXPRIV do software de segurança ou ser um superusuário (UID 0) se esse perfil não estiver definido. para obter informações adicionais, 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 usadas são o último número (após ".") na coluna "Local Socket". Como estas portas já estão em uso, elas não podem ser usadas para a configuração do Developer para System z.

### IPv4

```
MVS TCP/IP NETSTAT CS VxRy 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
RSED 0000004B 0.0.0.0..4035 0.0.0.0..0 Listen
JMON 00000038 0.0.0.0..6715 0.0.0.0..0 Listen
```

### IPv6

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

Outra limitação que pode existir são as portas TCP/IP reservadas. Há os dois lugares comuns a seguir para reservar as portas TCP/IP:

- **PROFILE.TCPIP**

Esse é o conjunto de dados referido pela instrução PROFILE DD da tarefa iniciada do TCP/IP, muitas vezes chamado 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 *Communications Server: IP Configuration Guide* (SC31-8775) para obter informações adicionais sobre essas 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. Esse valor é necessário somente para CINET (várias pilhas TCP/IP ativas em um único host).
  - **INADDRANYCOUNT**: Especifica o número de portas que o sistema reserva, iniciando com o número de porta especificado no parâmetro INADDRANYPORT. Esse valor é necessário somente para CINET (várias pilhas TCP/IP ativas em um único host).

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

Estas portas reservadas podem ser listadas com o comando **netstat portl** (TSO ou z/OS UNIX), que cria uma saída como esta do exemplo a seguir:

```

MVS TCP/IP NETSTAT CS VxRy TCPIP Name: TCPIP 17:08:32
Port# Prot User Flags Range IP Address
-----
00007 TCP MISC SERV DA
00009 TCP MISC SERV DA
00019 TCP MISC SERV DA
00020 TCP OMVS D
00021 TCP FTPD1 DA
00025 TCP SMTP DA
00053 TCP NAMESRV DA
00080 TCP OMVS DA
03500 TCP OMVS DAR 03500-03519
03501 TCP OMVS DAR 03500-03519

```

Consulte *Communications Server: IP System Administrator's Commands* (SC31-8781) para obter informações adicionais 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.

---

## Tamanho do Espaço de Endereço

O daemon RSE, que é um processo z/OS UNIX Java, exige um tamanho de região grande para executar suas funções. Portanto, é importante definir limites de armazenamento grandes para espaços de endereço do OMVS.

## Requisitos da JCL de Inicialização

O daemon RSE é iniciado pela JCL utilizando **BPXBATSL**, cujo tamanho da região deve ser 0.

## Limitações Definidas em SYS1.PARMLIB(BPXPRMxx)

Configure **MAXASSIZE** em **SYS1.PARMLIB(BPXPRMxx)**, que define o tamanho da região do espaço de endereço (processo) do OMVS como 2G. Esse é o tamanho máximo permitido. Esse é um limite amplo do sistema e, dessa forma, ativo em todos os



espaços de endereço do z/OS UNIX. Se isso não for o que você deseja, então poderá configurar o limite apenas para o Developer para System z em seu software de segurança.

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

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

## Limitações Armazenadas no Perfil de Segurança

Verifique ASSIZEMAX no segmento OMVS do ID do usuário do daemon e configure-o como 2147483647 ou, de preferência, como NONE para utilizar o valor SYS1.PARMLIB(BPXPRMxx).

Utilizando RACF, esse valor pode ser verificado e configurado com os seguintes comandos 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 permitir que saídas do sistema IEFUSI ou IEALIMIT controlem os tamanhos de regiões de espaços de endereços OMVS. Uma forma possível de fazer isso é pela codificação de SUBSYS(OMVS,NOEXITS) em SYS1.PARMLIB(SMFPRMxx).

Os valores 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

## Limitações para Endereçamento de 64 Bits

A palavra-chave MEMLIMIT em SYS1.PARMLIB(SMFPRMxx) limita o quanto uma tarefa de armazenamento virtual de 64 bits pode alocar acima da barra de 2GB. Ao contrário do parâmetro REGION no JCL, o MEMLIMIT=0M significa que o processo não pode usar o armazenamento virtual acima da barra.

Se o MEMLIMIT não estiver especificado em SMFPRMxx, o valor padrão será 0M, assim as tarefas serão limitadas ao (31 bits) 2GB abaixo da barra. O padrão alterado em z/OS 1.10 para 2G, permitindo que tarefas de 64 bits usem até 4GB (os 2GB abaixo da barra e os 2GB acima da barra concedidos pelo MEMLIMIT).

Os valores 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

O MEMLIMIT também pode ser especificado como parâmetro na placa EXEC no JCL. Se nenhum parâmetro MEMLIMIT estiver especificado, o padrão será o valor definido para SMF, exceto quando o REGION=0M estiver especificado, em tal caso o padrão será NOLIMIT.

---

## Transação APPC e Serviço de Comandos do TSO

Se não puder utilizar a versão APPC do serviço TSO Commands, 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 mais comuns de problemas incluem o seguinte:
  - Você não tem autoridade de leitura para o conjunto de dados FEK.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 causas mais prováveis dos problemas são as seguintes:
  - 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.
  - Os pontos JCL para o SYSPROC incorreto (SYSPROC deve incluir FEK.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 usada (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) Transação APPC para o Serviço de Comandos TSO” na página 102 pode ajudar a resolver problemas de APPC, pois você tem a possibilidade de gerenciar o APPC interativamente por meio dos painéis ISPF. Lembre-se, no entanto, de que você pode desativar a transação com essa 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/rdz/support/>). Consulte o Web site de suporte para obter mais informações:

- 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 das seguintes mensagens:
  - IEA995I SAÍDA DO DUMP DO SINTOMA 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. Verifique 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 para System z estão ativos. A maioria dos valores de BPXPRMxx pode ser alterada dinamicamente com os comandos do console **SETOMVS** e **SET OMVS**.

Use o comando do console **SETOMVS LIMMSG=ALL** para que o z/OS UNIX exiba mensagens do console (BPXI040I) quando qualquer dos limites BPXPRMxx estiver prestes a ser atingido.

### Conexão Recusada

Cada conexão RSE inicia diversos processos que são permanentemente ativos. Novas conexões podem ser recusadas devido ao limite configurado em SYS1.PARMLIB(BPXPRMxx) na quantidade de processos, especialmente quando os usuários compartilham o mesmo UID (como ao utilizar o 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 Address Space IDs (ASID) é definida em SYS1.PARMLIB(IEASYSxx) com a palavra-chave MAXUSER e tem 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 de Requisitos Conhecidos

### Falhas ao Abrir Conjuntos de Dados MVS

Ao usar o APPC para o serviços TSO Commands, a leitura e a gravação de um conjunto de dados MVS exige o uso de um domínio do sistema de arquivos 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(2000)
          TYPE(UDS)
```

Outra causa provável para esse problema, durante o uso de APPC para o serviço TSO Commands, é que o TCP/IP Resolver não pode resolver o endereço do host de modo adequado devido a um arquivo de configuração do resolvidor ausente ou incompleto. Uma indicação clara desse problema é a seguinte mensagem em lock.log:

```
clientip(0.0.0.0) <> callerip(<endereço IP do host>)
```

Execute o IVP TCP/IP do fekfivpt, conforme descrito em Capítulo 7, “Verificação de Instalação”, na página 107. A seção de configuração do resolvidor da saída será semelhante à seguinte amostra:

```
Resolver Trace Initialization Complete -> 2008/07/02 13:11:54.745964
```

```
res_init Resolver values:
Global Tcp/Ip Dataset = None
Default Tcp/Ip Dataset = None
Local Tcp/Ip Dataset = /etc/resolv.conf
Translation Table = Default
UserId/JobName = USERID
Caller API = LE C Sockets
Caller Mode = EBCDIC
```

Certifique-se de que as definições no arquivo (ou conjunto de dados) referido pelo “Local Tcp/Ip Dataset” estejam corretas.

Esse campo ficará em branco se você não utilizar um nome padrão para o arquivo resolvidor de IP (utilizando a ordem de procura do z/OS UNIX). Nesse caso,

inclua a seguinte instrução no `rsed.envvars`, em que `<arquivo resolvidor>` ou `<dados do resolvidor>` representam o nome do arquivo resolvidor de IP:

```
RESOLVER_CONFIG=<arquivo resolvidor>
```

ou

```
RESOLVER_CONFIG='<conjunto de dados do resolvidor>'
```

## 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 está utilizando telnet segura (SSL) e trabalhando com certificados que não são assinados por um CA conhecido, cada cliente deve incluir o certificado CA na lista de CAs confiáveis do Emulador de Conexão do Host.
- 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.



---

## Capítulo 10. Considerações sobre Segurança

O Developer para System z fornece acesso ao mainframe para usuários de uma estação de trabalho sem mainframe. A validação dos pedidos de conexão, o fornecimento de comunicação segura entre o host e a estação de trabalho, e a atividade de autorização e auditoria são aspectos importantes da configuração do produto.

Os mecanismos de segurança usados pelos servidores e serviços do Developer para System z dependem de que o sistema de arquivo em que eles residem seja seguro. Isto indica que apenas administradores confiáveis de sistema podem ser capazes de atualizar as bibliotecas de programa e os arquivos de configuração.

Os seguintes tópicos são abordados neste capítulo:

- “Métodos de Autenticação”
- “Segurança de Conexão” na página 159
- “Portas TCP/IP” na página 160
- “Usando os PassTickets” na página 162
- “Criação de Log de Auditoria” na página 163
- “Segurança do JES” na página 164
- “Comunicação Criptografada por SSL” na página 167
- “Autenticação de Cliente Usando Certificados X.509” na página 168
- “Verificação de Port Of Entry (POE)” na página 172
- “Segurança do CICSTS” na página 172
- “Segurança de SCLM” na página 173
- “Arquivos de Configuração do Developer para System z” na página 173
- “Definições de Segurança” na página 175

**Nota:** O Explorador de Sistemas Remotos (RSE), que fornece os principais serviços, como conexão do cliente com o host, consiste em 2 entidades lógicas:

- Daemon RSE, que gerencia a configuração da conexão e que é iniciado como uma tarefa iniciada ou uma tarefa de usuário de longa execução.
- Servidor RSE, que manipula pedidos individuais do cliente e é iniciado como um encadeamento em um ou mais processos-filho pelo daemon RSE.

Consulte o Capítulo 11, “Entendimento do Developer para System z”, na página 189 para saber sobre os conceitos básicos de design do Developer para System z.

---

### Métodos de Autenticação

O Developer para System z suporta várias maneiras de autenticar um ID do usuário fornecido por um cliente an conexão.

- ID do Usuário e Senha
- ID do Usuário e Senha Única
- Certificado X.509

Observe que os dados de autenticação fornecidos pelo cliente são usados somente uma vez durante a configuração da conexão inicial. Quando um ID do usuário é autenticado, o ID do usuário e os PassTickets autogerados são usados para todas as ações que requerem autenticação.

## **ID do Usuário e Senha**

O cliente fornece um ID de usuário e uma senha correspondente na conexão. O ID de usuário e a senha são usados para autenticar o usuário com seu produto de segurança.

## **ID do Usuário e Senha Única**

Com base em um token exclusivo, uma senha única pode ser gerada por um produto de terceiro. Senhas únicas melhoram a configuração da segurança já que o token exclusivo não pode ser copiado e usado sem o conhecimento do usuário e uma senha interceptada é inútil já que é válida somente uma vez.

O cliente fornece um ID de usuário e a senha única na conexão, que é usada para autenticar o ID do usuário com a saída de segurança fornecida por terceiro. Espera-se que essa saída de segurança ignore os PassTickets usados para satisfazer pedidos de autenticação durante o processamento normal. Os PassTickets devem ser processados por seu software de segurança.

## **Certificado X.509**

Um terceiro pode fornecer um ou mais certificados X.509 que podem ser usados para autenticar um usuário. Quando armazenado em dispositivos seguros, os certificados X.509 combinam uma configuração segura com facilidade de uso para o usuário (nenhum ID do usuário ou senha necessário).

Ao conectar, o cliente fornece um certificado selecionado e, como opção, uma extensão selecionada, que é usada para autenticar o ID do usuário com seu produto de segurança.

Observe que esse método de autenticação é suportado apenas pelo método de conexão do daemon RSE e que o SSL deve estar ativado.

## **Autenticação do JES Job Monitor**

A autenticação do cliente é realizada pelo daemon do RSE (ou REXEC/SSH) como parte do pedido de conexão do cliente. Depois de o usuário ser autenticado, os PassTickets gerados automaticamente são usados para todos os pedidos de autenticação futuros, incluindo o logon automático no JES Job Monitor.

Para que o JES Job Monitor valide o ID do usuário e o PassTicket apresentados pelo RSE, o JES Job Monitor deve ter permissão para avaliar o PassTicket. Isso implica no seguinte:

- Carregue o módulo FEJMON, por padrão, localizado na biblioteca de carregamento FEK.SFEKAUTH, deve ser autorizado por APF.
- O RSE e o JES Job Monitor devem usar o mesmo ID de aplicativo (APPLID). Por padrão, ambos os servidores usam FEKAPPL como APPLID, mas isso pode ser alterado pela diretiva APPLID em rsed.envvars para RSE e em FEJCNFG para o JES Job Monitor.



**Nota:** Clientes anteriores (versão 7.0 e mais antiga) comunicam-se diretamente com o JES Job Monitor. Para essas conexões, somente o método de autenticação de ID de usuário e senha é suportado.

---

## Segurança de Conexão

Diferentes níveis de segurança de comunicação são suportados pelo RSE, que controla toda a comunicação entre os serviços de cliente e do Developer para System z:

- A comunicação externa (cliente-host) pode ser limitada a portas especificadas. Esse recurso é desativado por padrão.
- A comunicação externa (cliente-host) pode ser criptografada utilizando SSL. Esse recurso é desativado por padrão.
- A verificação de Port Of Entry (POE) pode ser usada para permitir acesso ao host apenas a endereços TCP/IP confiáveis. Esse recurso é desativado por padrão.

### Limitar Comunicação Externa a Portas Especificadas

O programador de sistemas pode especificar as portas nas quais o servidor RSE pode se comunicar com o cliente. Por padrão, qualquer porta disponível é usada. Esse intervalo de portas não possui conexão com a porta do daemon RSE.

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 do host 4035, daemon do RSE.
2. O daemon do RSE cria um encadeamento do servidor RSE.
3. O servidor 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 daemon RSE retorna o número da porta para o cliente.
5. O cliente se conecta à porta do host.

**Nota:** O processo é semelhante para o método de conexão alternativo (opcional) usando REXEC/SSH, que é descrito em “(Opcional) Usando REXEC (ou SSH)” na página 100.

### Criptografia de Comunicação Usando SSL

Todos os fluxos de dados externos do Developer para System z que passam pelo RSE podem ser criptografados utilizando-se Secure Socket Layer (SSL). O uso de SSL é controlado pelas configurações no arquivo de configuração `ssl.properties`, conforme descrito em “Comunicação Criptografada por SSL” na página 167.

O Emulador de Conexão do Host no cliente se conecta a um servidor TN3270 no host. O uso de SSL é controlado pelo TN3270, conforme documentado em *Communications Server IP Configuration Guide* (SC31-8775).

O cliente Application Deployment Manager usa o Serviço da Web CICS TS ou a interface RESTful para chamar os serviços de host do Application Deployment Manager. O uso de SSL é controlado pelo CICS TS, conforme documentado no *RACF Security Guide for CICS TS*.

## Verificação de Port Of Entry

O Developer para System z suporta a verificação de Port Of Entry (POE), que permite que o host acesse apenas os endereços TCP/IP confiáveis. O uso de POE é controlado pela definição de perfis específicos em seu software de segurança e a diretiva `enable.port.of.entry` em `rsed.envvars`, conforme descrito em “Verificação de Port Of Entry (POE)” na página 172.

Observe que a ativação de POE afetará outros aplicativos TCPIP que suportam a verificação de POE, como o INETD.

---

## Portas TCP/IP

*Figura 40. Portas TCP/IP*

A Figura 40 mostra as portas TCP/IP que podem ser usadas pelo Developer para System z. As setas mostram a parte que faz a conexão (lado da ponta da seta) e o que conecta.

## Comunicação Externa

Defina as seguintes portas para seu firewall protegendo o host do z/OS, uma vez que são usadas para a comunicação de cliente-host (utilizando o protocolo tcp):

- Daemon RSE para a configuração da comunicação do cliente-host, porta padrão 4035. A comunicação nesta porta pode ser criptografada utilizando SSL.

- Servidor RSE para comunicação de host do cliente. Por padrão, qualquer porta disponível é usada, mas isso pode ser limitado a um intervalo especificado com a definição `_RSE_PORTRANGE` no `rsed.envvars`. A comunicação nesta porta pode ser criptografada utilizando SSL.
- (opcional) Serviço INETD para ações remotas (baseadas em host) em subprojetos z/OS UNIX:
  - REXEC (versão z/OS UNIX), porta padrão 512.
  - SSH (versão z/OS UNIX), porta padrão 22. A comunicação nessa porta é criptografada utilizando SSL.
- (opcional) Serviço Telnet TN3270 para o Emulador de Conexão do Host, porta padrão 23. A comunicação pode ser criptografada utilizando SSL (porta padrão 992). A porta padrão designada ao serviço Telnet TN3270 depende de o usuário escolher ou não o uso de criptografia.
- (opcional) Uma ou as duas interfaces de aplicativo CICSTS para o Application Deployment Manager:
  - interface RESTful, porta padrão 5130.
  - interface de Serviços da Web, porta padrão 5129. A comunicação nesta porta pode ser criptografada utilizando SSL.

**Nota:**

- Clientes anteriores (versão 7.0 e mais antiga) comunicam-se diretamente com o JES Job Monitor, porta padrão 6715.
- Durante uma sessão de depuração remota para Cobol, PL/I ou Assembler, o IBM Debug Tool para z/OS é chamado. Esse produto comunica-se diretamente com o cliente. Essa comunicação é iniciada no host e conecta-se à porta 8001 no cliente.

## Comunicação Interna

Vários serviços do host Developer para System z são executados em encadeamentos ou espaços de endereço separados e utilizam soquetes TCP/IP como mecanismo de comunicação. Todos esses serviços usam o RSE para comunicação com o cliente, tornando seu fluxo de dados confinado apenas ao host. Para alguns serviços, será usada qualquer porta disponível, para outros, o programador de sistema poderá escolher a porta ou o intervalo de portas que será usado:

- JES Job Monitor para serviços relacionados ao JES, porta padrão 6715. A porta pode ser configurada no membro de configuração FEJJCNGF.
- O Daemon de bloqueio para serviços relacionados a bloqueio de conjunto de dados, porta padrão 4036. A porta pode ser configurada no membro de configuração `rsed.envvars`.
- (opcional) Integração do File Manager para interagir com o IBM File Manager, porta padrão 1960.
- (opcional) Comunicação do CARMA, intervalo de porta padrão 5227-5326 (100 portas). O intervalo de portas pode ser configurado no arquivo de configuração `CRASRV.properties`.
- (opcional) A versão APPC do serviço TSO Commands utiliza qualquer soquete disponível para se comunicar com o gerenciador de bloqueio (que enfileira conjuntos de dados do MVS para clientes). Você não pode configurar um intervalo de portas específico para ser usado.

**Nota:** Os clientes anteriores (versão 7.0 e anteriores) comunicam-se diretamente com o servidor JES Job Monitor, porta padrão 6715.

## Portas CARMA e TCP/IP

Na maioria dos casos, como no daemon RSE, um servidor se conecta a uma porta e atende a pedidos de conexão. O CARMA, no entanto, usa uma abordagem diferente, uma vez que o servidor CARMA não está ativo ainda quando o cliente inicia o pedido de conexão.

Quando o cliente envia um pedido de conexão, o extrator do CARMA, que fica ativo como um encadeamento de usuário em um conjunto de encadeamento RSE, encontrará uma porta livre em um intervalo especificado no arquivo de configuração CRASRV.properties e se conectará a ela. O extrator inicia então o servidor CARMA e transmite o número da porta, para que o servidor saiba a que porta se conectar. Quando o servidor estiver conectado, o cliente pode enviar pedidos ao servidor e receber os resultados.

Portanto, a partir de uma perspectiva TCP/IP, RSE (por meio do extrator do CARMA) é o servidor que se conecta à porta e o servidor CARMA é o cliente que se conecta a ela.

---

## Usando os PassTickets

Após o logon, os PassTickets são usados para estabelecer segurança de encadeamento no servidor RSE. Esse recurso não pode ser desativado. Os PassTickets são senhas geradas pelo sistema com um lifespan de aproximadamente 10 minutos. Os PassTickets gerados baseiam-se no algoritmo de criptografia DES, no ID do usuário, no ID do aplicativo, em um registro de data e hora e em uma chave secreta. Essa chave secreta é um número de 64 bits (16 caracteres hexadecimais) que deve ser definido para seu software de segurança.

Para ajudá-lo a compreender o uso de PassTicket, a seguir há uma breve descrição do processo de segurança do RSE:

1. O cliente se conecta à porta do host 4035, daemon do RSE.
2. O daemon RSE autentica o cliente usando as credenciais apresentadas pelo cliente.
3. O daemon RSE cria um ID de cliente exclusivo e um encadeamento do servidor RSE.
4. O servidor RSE gera um PassTicket e cria um ambiente de segurança para o cliente, utilizando o PassTicket como senha.
5. O cliente se conecta à porta do host retornada pelo daemon RSE.
6. O servidor RSE valida o cliente utilizando o ID do cliente.
7. O servidor RSE utiliza um PassTicket recém-gerado como senha para todas as ações futuras que requerem uma senha.

A senha real do cliente não é mais necessária após a autenticação inicial, pois os produtos de segurança compatíveis com SAF podem avaliar as senhas PassTickets e comuns. O servidor RSE gera e utiliza um PassTicket cada vez que uma senha é solicitada, resultando em uma senha válida (temporária) para o cliente.

O uso de PassTickets permite que o RSE configure um ambiente de segurança específico do usuário à vontade, sem a necessidade de armazenar todos os IDs de usuário e senhas em uma tabela, o que poderia ser comprometido. Ele também permite métodos de autenticação de cliente que não usam senhas reutilizáveis, como certificados X.509.

Os perfis de segurança nas classes APPL e PTKTDATA são necessários para que se possa usar os PassTickets. Esses perfis são específicos do aplicativo e, portanto, não afetam a configuração atual do sistema.

Os PassTickets sendo específicos do aplicativo implicam em o RSE e o JES Job Monitor usarem o mesmo ID de aplicativo (APPLID). Por padrão, ambos os servidores usam FEKAPPL como APPLID, mas isso pode ser alterado pela diretiva APPLID em rsed.envvars para o RSE e em FEJCNFG para o JES Job Monitor.

Não é recomendável usar o OMVSAPPL como ID do aplicativo porque ele abrirá uma chave secreta para a maioria dos aplicativos z/OS UNIX. Também não é recomendável usar o ID do aplicativo padrão MVS, que é MVS seguido pelo ID do sistema SMF, porque isto abrirá uma chave secreta para a maioria dos aplicativos MVS (incluindo as tarefa em lote do usuário).

**Atenção:** O pedido de conexão do cliente falhará se os PassTickets não estiverem configurados corretamente.

---

## Criação de Log de Auditoria

O Developer para System z suporta a criação de log de auditoria das ações que são gerenciadas pelo daemon RSE. Os logs de auditoria são armazenados como arquivos de texto no diretório de log do daemon, utilizando o formato CSV (Comma Separated Value).

### Controle de Auditoria

Várias opções em rsed.envvars influenciam a função de auditoria, conforme documentado em “Definindo Parâmetros Extras de Inicialização Java com \_RSE\_JAVAOPTS” na página 42.

- A função de auditoria é ativada/desativada pela opção `enable.audit.log`.
- Os padrões de auditoria são controlados pelas opções de `audit.*`.
- O local dos arquivos de log de auditoria é controlado pela opção `daemon.log`.
- A página de códigos usada para gravação do log de auditoria é controlada pela diretiva `_RSE_HOST_CODEPAGE`, conforme documentado em “Arquivo de Configuração do RSE rsed.envvars” na página 31.

O comando do operador **modify switch** pode ser usado para alternar manualmente para um novo arquivo de log de auditoria, conforme documentado em Capítulo 8, “Comandos do Operador”, na página 123.

Uma mensagem de aviso é enviada para o console quando o sistema de arquivos que contém os arquivos de log de auditoria estiver em execução com pouco espaço livre. Essa mensagem do console (FEK103E) é repetida regularmente até que o problema de pouco espaço seja resolvido. Consulte “Mensagens do Console” na página 130 para obter uma lista de mensagens do console geradas pelo RSE.

### Dados de Auditoria

Um novo arquivo de log de auditoria é iniciado após um momento predeterminado ou quando o comando do operador **modify switch** é emitido. O arquivo de log antigo é salvo como `audit.log.yyyymmdd.hhmmss`, em que `yyymmdd.hhmmss` é a data/registro de data e hora no qual o log foi fechado. A data/registro de data e hora do sistema designados ao arquivo indicam a criação

do arquivo de log. A combinação das duas datas mostra o período de tempo abrangido por esse arquivo de log de auditoria.

As seguintes ações são registradas:

- Acesso ao sistema (conexão, desconexão)
- Acesso ao spool JES (envio, exibição, suspensão, liberação, cancelamento, limpeza)
- Acesso ao conjunto de dados (leitura, gravação, criação, exclusão, renomeação, compactação, migração, rechamada)
- Execução de comandos do TSO

Cada ação registrada é armazenada (com uma data/registro de data e hora) utilizando o formato Comma Separated Value (CSV), que pode ser lido por uma ferramenta de automação ou de análise de dados.

Os arquivos de log de auditoria possuem a máscara de bit de permissão 640 (-rw-r-----), o que significa que o proprietário (uid do daemon RSE z/OS UNIX) possui acesso de leitura e gravação, e o grupo do proprietário (padrão) possui acesso de leitura. Todas as outras tentativas de acesso são negadas, a menos que isso seja feito por um superusuário (UID 0) ou alguém com permissão suficiente para o perfil SUPERUSER.FILESYS na classe UNIXPRIV.

---

## Segurança do JES

O Developer para System z permite acesso do cliente ao spool do JES através do JES Job Monitor. O servidor fornece limitações de acesso básico, que podem ser estendidas com os recursos de proteção padrão do arquivo de spool de seu produto de segurança. As ações do operador (Suspend, Liberar, Cancelar e Limpar) nos arquivos de spool são feitas através do console EMCS, para o qual é necessário configurar permissões condicionais.

### Ações nas Tarefas - Limitações de Destino

O JES Job Monitor não fornece aos usuários do Developer para System z acesso total de operador ao spool do JES. Apenas os comandos Suspend, Liberar, Cancelar e Limpar estão disponíveis e, por padrão, somente para arquivos em spool que pertencem ao usuário. Os comandos são emitidos selecionando a opção apropriada na estrutura de menus do cliente (sem prompt de comandos). O escopo dos comandos pode ser ampliado, utilizando perfis de segurança para definir para quais tarefas os comandos estão disponíveis.

Semelhante ao caractere de ação SDSF SJ, o JES Job Monitor também suporta o comando Mostrar JCL para recuperar a JCL que criou a saída de tarefa selecionada e o mostra em uma janela de editor. O JES Job Monitor recupera a JCL do JES, tornando-o uma função útil para situações em que o membro JCL original não é facilmente localizado.

*Tabela 21. Comandos do Console do JES Job Monitor*

Ações	JES2	JES3
Suspend der	\$Hx(jobid) with x = {J, S or T}	*F,J=jobid,H
Libera ção	\$Ax(jobid) with x = {J, S or T}	*F,J=jobid,R

Tabela 21. Comandos do Console do JES Job Monitor (continuação)

Ações	JES2	JES3
Cancelar	\$Cx(jobid) with x = {J, S or T}	*F,J=jobid,C
Limpar	\$Cx(jobid),P with x = {J, S or T}	*F,J=jobid,C
Mostrar JCL	não-aplicável	não-aplicável

Os comandos JES disponíveis listados na Tabela 21 na página 164 são limitados por padrão às tarefas que pertencem ao usuário. Isso pode ser alterado com a diretiva `LIMIT_COMMANDS`, conforme documentado em “FEJJCNFG, Arquivo de Configuração do JES Job Monitor” na página 27.

Tabela 22. Matriz de Permissão do Comando `LIMIT_COMMANDS`

LIMIT_COMMANDS	Proprietário da tarefa	
	Usuário	Outros
USERID (padrão)	Permitido	Não permitido
LIMITED	Permitido	Permitido somente se for permitido explicitamente por perfis de segurança
NOLIMIT	Permitido	Permitido se for permitido pelos perfis de segurança ou quando a classe JESSPOOL não estiver ativa

O JES utiliza a classe JESSPOOL para proteger os conjuntos de dados `SYSIN/SYSOUT`. Semelhante a SDSF, o JES Job Monitor estende o uso da classe JESSPOOL para proteger os recursos de tarefas também.

Se `LIMIT_COMMANDS` não for `USERID`, o JES Job Monitor consultará se há permissão para o perfil relacionado na classe JESSPOOL, conforme mostrado na tabela a seguir.

Tabela 23. Perfis JESSPOOL Estendidos

Comando	Perfil JESSPOOL	Acesso Necessário
Suspender	nodeid.userid.jobname.jobid	ALTER
Liberação	nodeid.userid.jobname.jobid	ALTER
Cancelar	nodeid.userid.jobname.jobid	ALTER
Limpar	nodeid.userid.jobname.jobid	ALTER
Mostrar JCL	nodeid.userid.jobname.jobid.JCL	READ

Use as seguintes substituições na tabela anterior:

nodeid	O ID do nó NJE do subsistema JES de destino
userid	ID do usuário local do proprietário da tarefa
jobname	Nome da tarefa



jobid	ID da tarefa do JES
-------	---------------------

Se a classe JESSPOOL não estiver ativa, haverá um comportamento diferente para o valor LIMITED e NOLIMIT de LIMIT\_COMMANDS, conforme descrito em Tabela 9 na página 30. O comportamento é idêntico quando JESSPOOL está ativo, já que a classe, por padrão, nega a permissão se um perfil não estiver definido.

## Ações nas Tarefas - Limitações de Execução

A segunda fase da segurança de comando em spool do JES, depois de especificar os destinos permitidos, inclui as permissões necessárias para realmente executar o comando do operador. Essa autorização de execução é aplicada pelas verificações de segurança do z/OS e do JES.

Observe que Mostrar JCL não é um comando de operador, como os outros comandos JES Job Monitor (Suspend, Liberar, Cancelar e Limpar), portanto, as limitações abaixo não se aplicam porque não há nenhuma verificação de segurança adicional.

O JES Job Monitor emite todos os comandos do operador JES solicitados por um usuário por meio de um console MCS estendido (EMCS), cujo nome é controlado com a diretiva CONSOLE\_NAME, conforme documentado em “FEJJCNFG, Arquivo de Configuração do JES Job Monitor” na página 27.

Essa configuração permite que o administrador de segurança defina permissões granulares de execução de comandos usando as classes OPERCMDS e CONSOLE.

- Para utilizar um console EMCS, um usuário deve ter (pelo menos) a autoridade READ para o perfil MVS.MCSOPER.console-name na classe OPERCMDS. Observe que, se nenhum perfil estiver definido, o sistema concederá o pedido de autoridade.
- Para executar um comando do operador JES, um usuário deverá ter autoridade suficiente para o perfil JES%.\*\* (ou mais específico) na classe OPERCMDS. Observe que, se nenhum perfil estiver definido, ou a classe OPERCMDS não estiver ativa, o JES causará falha do comando.
- O administrador de segurança também pode exigir que um usuário utilize o JES Job Monitor ao executar o comando do operador especificando WHEN(CONSOLE(JMON)) na definição **PERMIT**. A classe CONSOLE deverá estar ativa para que esta configuração funcione. Observe que a classe CONSOLE estando ativa é suficiente; nenhum perfil é verificado para consoles EMCS.

Supondo que a identidade do servidor JES Job Monitor, criando um console JMON a partir de uma sessão do TSO, seja impedida por seu software de segurança. Embora o console possa ser criado, o ponto de entrada é diferente (JES Job Monitor versus TSO). Os comandos JES emitidos a partir desse console falharão na verificação de segurança se a segurança estiver configurada conforme documentado nesta publicação e o usuário não tiver autoridade para comandos JES por outros meios.

Observe que o JES Job Monitor não poderá criar o console quando um comando tiver que ser executado se o nome do console já estiver sendo usado. Para evitar isso, o programador de sistema pode configurar a diretiva GEN\_CONSOLE\_NAME=ON no arquivo de configuração do JES Job Monitor ou o administrador de segurança pode definir perfis de segurança para que os usuários do TSO parem de criar um console. Os comandos de amostra do RACF a seguir impedem que qualquer indivíduo (exceto aqueles permitidos) crie um console TSO ou SDSF:



- RDEFINE TSOAUTH CONSOLE UACC(NONE)
- PERMIT CONSOLE CLASS(TSOAUTH) ACCESS(READ) ID(#userid)
- RDEFINE SDSF ISFCMD.ODSP.ULOG.\* UACC(NONE)
- PERMIT ISFCMD.ODSP.ULOG.\* CLASS(SDSF) ACCESS(READ) ID(#userid)

**Nota:** Sem autorização para esses comandos do operador, os usuários ainda poderão enviar tarefas e ler a saída da tarefa por meio do JES Job Monitor, se tiverem autoridade suficiente para possíveis perfis que protejam esses recursos (como aqueles das classes JESINPUT, JESJOBS e JESSPOOL).

Consulte o *Security Server RACF Security Administrator's Guide* (SA22-7683) para obter informações adicionais sobre proteção de comandos do operador.

## Acesso aos Arquivos de Spool

O JES Job Monitor permite acesso de procura a todos os arquivos em spool, por padrão. Isso pode ser alterado com a diretiva `LIMIT_VIEW`, conforme documentado em “FEJJCNFG, Arquivo de Configuração do JES Job Monitor” na página 27.

Tabela 24. Matriz de permissão de navegação `LIMIT_VIEW`

LIMIT_VIEW	Proprietário da tarefa	
	Usuário	Outros
USERID	Permitido	Não permitido
NOLIMIT (padrão)	Permitido	Permitido se for permitido pelos perfis de segurança ou quando a classe JESSPOOL não estiver ativa

Para limitar os usuários às suas próprias tarefas no spool JES, defina a instrução “`LIMIT_VIEW=USERID`” no arquivo de configuração do JES Job Monitor, `FEJJCNFG`. Se os usuários precisarem de acesso a um intervalo maior de tarefas, mas não todas, use os recursos de proteção de arquivo de spool padrão do seu produto de segurança, como a classe JESSPOOL.

Ao definir a proteção adicional, lembre-se que o JES Job Monitor utiliza a SAPI (SYSOUT Application Program Interface) para acessar o spool. Isto significa que o usuário precisa de, pelo menos, acesso `UPDATE` aos arquivos de spool, mesmo para a funcionalidade de procura. Esse requisito não se aplicará se você executar o z/OS 1.7 (z/OS 1.8 para JES3) ou superior. Aqui, a permissão `READ` é suficiente para a funcionalidade de procura.

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

---

## Comunicação Criptografada por SSL

A comunicação externa (cliente-host) pode ser criptografada utilizando-se SSL (Secure Socket Layer). Esse recurso é desativado por padrão e é controlado pelas configurações em `ssl.properties`, conforme documentado em “(Opcional) Criptografia SSL do RSE” na página 91.

O daemon RSE e o servidor RSE suportam diferentes mecanismos para armazenar certificados devido a diferenças de arquitetura entre eles. Isso implica que as definições e os certificados SSL são necessários para o daemon RSE e para o

servidor RSE. Um certificado compartilhado poderá ser usado se o daemon RSE e o servidor RSE usarem o mesmo método de gerenciamento de certificado.

*Tabela 25. Mecanismos de armazenamento de certificado SSL*

Armazenamento de certificado	Criado e gerenciado por	Daemon RSE	servidor RSE
conjunto de chaves	produto de segurança compatível com SAF	suportados	suportados
banco de dados de chaves	gskkyman do z/OS UNIX	suportados	/
keystore	keytool Java	/	suportados

**Nota:** Conjuntos de chaves compatíveis com SAF é o método preferido para gerenciar certificados.

Os conjuntos de chaves compatíveis com SAF podem armazenar a chave privada do certificado no banco de dados de segurança ou usando ICSF (Integrated Cryptographic Service Facility), a interface para o hardware de criptografia do System z.

ICSF é recomendado para o armazenamento de chaves privadas associadas a certificados digitais, porque é uma solução mais segura do que o gerenciamento de chaves privadas não-ICSF. O ICSF garante que as chaves privadas sejam criptografadas na chave mestra do ICSF e que o acesso a elas seja controlado por recursos gerais das classes de segurança CSFKEYS e CSFSERV. Além disso, o desempenho operacional é aprimorado, pois o ICSF utiliza o Coprocessador Criptográfico de hardware.

O daemon RSE utiliza funções SSL do Sistema para gerenciar as comunicações criptografadas por SSL. Isso implica que SYS1.SIEALNKE deve ser controlado pelo programa pelo software de segurança e estar disponível para o RSE através de LINKLIST ou da diretiva STEPLIB em rsed.envvars.

O ID do usuário do RSE (STCRSE nos comandos de amostra abaixo) precisa de autorização para acessar esse conjunto de chaves e os certificados relacionados quando os conjuntos de chaves compatíveis com SAF são usados para o daemon RSE ou o servidor RSE.

- RDEFINE FACILITY IRR.DIGTCERT.LIST UACC(NONE)
- RDEFINE FACILITY IRR.DIGTCERT.LISTRING UACC(NONE)
- PERMIT IRR.DIGTCERT.LIST CLASS(FACILITY) ACCESS(READ) ID(stcrse)
- PERMIT IRR.DIGTCERT.LISTRING CLASS(FACILITY) ACCESS(READ) ID(stcrse)
- SETROPTS RACLIST(FACILITY) REFRESH

Consulte o Apêndice A, “Configurando o SSL e a Autenticação X.509”, na página 293 para obter mais detalhes sobre como ativar o SSL para Developer para System z.

---

## Autenticação de Cliente Usando Certificados X.509

O daemon RSE suporta que os próprios usuários se autenticuem com um certificado X.509. Usar a comunicação criptografada SSL é um pré-requisito para essa função, uma vez que é uma extensão para a autenticação de host com um certificado usado no SSL.

O daemon RSE inicia o processo de autenticação de cliente pela validação do certificado de cliente. Alguns aspectos chave que são verificados são as datas de validade do certificado e a fidelidade da Autoridade de Certificação (CA) usada para assinar o certificado. Opcionalmente, também é possível consultar uma Lista de Revogação de Certificado (CRL) (terceiros).

Depois que o daemon RSE valida o certificado, ele é processado para autenticação. O certificado é passado adiante para seu produto de segurança para autenticação, a menos que a diretiva `rsed.envvars.enable.certificate.mapping` esteja configurada como `false`, quando o daemon do RSE fará a autenticação.

Se bem-sucedido, o processo de autenticação determinará o ID de usuário a ser usado para esta sessão, que será, então, testado pelo daemon RSE para assegurar que seja útil no sistema host em que o daemon RSE está em execução.

A última verificação (que é feita para cada mecanismo de autenticação, não apenas certificados X.509) verifica se o ID do usuário tem permissão para usar o Developer para System z.

Se você estiver familiarizado com as classificações de segurança do SSL usadas por TCP/IP, a combinação dessas etapas de validação corresponderão às especificações de “Autenticação de Cliente Nível 3” (a mais alta disponível).

## Validação da Autoridade de Certificação (CA)

Parte do processo de validação do certificado inclui verificar se o certificado foi assinado por uma Autoridade de Certificação (CA) de confiança. Para fazer isso, o daemon do RSE deve ter acesso a um certificado que identifique a CA.

Ao usar o banco de dados de chaves **gskkyman** para sua conexão SSL, o certificado da CA deve ser incluído no banco de dados de chaves.

Ao usar um conjunto de chaves SAF (que é o método aconselhado), você deve incluir o certificado da CA em seu banco de dados de segurança como o certificado CERTAUTH com o atributo TRUST ou HIGHTRUST, conforme mostrado neste comando RACF de amostra:

- `RACDCERT CERTAUTH ADD(dsn) HIGHTRUST WITHLABEL('label')`

Observe que a maioria dos produtos de segurança já tem os certificados para as CAs reconhecidas disponíveis em seu banco de dados com um status NOTRUST. Use os comandos RACF de amostra a seguir para listar os certificados de CA existentes e marcar um como confiável com base no rótulo designado a ele.

- `RACDCERT CERTAUTH LIST`
- `RACDCERT CERTAUTH ALTER(LABEL('HighTrust CA')) HIGHTRUST`

**Nota:** O status HIGHTRUST será necessário se você depender do RACF para autenticar o usuário com base na extensão `HostIdMappings` do certificado. Consulte o “Autenticação por Software de Segurança” na página 170 para obter informações adicionais.

Quando o certificado da CA for incluído em seu banco de dados de segurança, ele deverá ser conectado ao conjunto de chaves RSE, conforme mostrado neste comando RACF de amostra:

- `RACDCERT ID(stcrse) CONNECT(CERTAUTH LABEL('HighTrust CA') RING(rdzssl.racf))`

Consulte o *Security Server RACF Command Language Reference* (SA22-7687) para obter informações adicionais sobre o comando **RACDCERT**.

**Atenção:** Se você depender do daemon RSE em vez de seu software de segurança para autenticar um usuário, deverá tomar cuidado para não confundir as CAs com os status TRUST e HIGHTRUST no conjunto de chaves SAF ou no banco de dados de chaves **gskkyman**. O daemon do RSE não é capaz de diferenciar entre os dois, portanto, os certificados assinados por uma CA com status TRUST será válido para propósitos de autenticação de ID do usuário.

## (Opcional) Consulte uma Certificate Revocation List (CRL)

Se desejado, é possível instruir o daemon do RSE para verificar uma ou mais Certificate Revocation Lists (CRL) para incluir segurança extra para o processo de validação. Isso é feito incluindo variáveis de ambiente relacionadas à CRL em `rsed.envvars`. Consulte “Arquivo de Configuração do RSE `rsed.envvars`” na página 31 para obter informações sobre estas variáveis de amostra:

- GSK\_CRL\_SECURITY\_LEVEL
- GSK\_LDAP\_SERVER
- GSK\_LDAP\_PORT
- GSK\_LDAP\_USER
- GSK\_LDAP\_PASSWORD

Consulte *Cryptographic Services System Secure Sockets Layer Programming* (SC24-5901) para obter informações adicionais sobre essas e outras variáveis de ambiente usadas pelo z/OS System SSL.

**Nota:** Tome cuidado ao especificar outras variáveis de ambiente z/OS System SSL (GSK\_\*) em `rsed.envvars`, pois elas podem alterar a maneira como o daemon RSE trata as conexões SSL e a autenticação de certificado.

## Autenticação por Software de Segurança

O RACF executa várias verificações para autenticar um certificado e retornar o ID do usuário associado. Observe que outros produtos de segurança podem fazer isso de forma diferente. Consulte a documentação de seu produto de segurança para obter informações adicionais sobre a função `initACEE` usada para realizar a autenticação (modo de consulta).

1. O RACF verifica se o certificado está definido na classe DIGTCERT. Se estiver, o RACF retornará o ID do usuário que estava associado a este certificado quando ele foi incluído no banco de dados RACF.

Os certificados são definidos como RACF usando o comando **RACDCERT**, como no seguinte exemplo:

```
RACDCERT ID(userid) ADD(dsn) TRUST WITHLABEL('label')
```

2. Se o certificado não estiver definido, o RACF verificará para ver se há um filtro de nome de certificado correspondente definido nas classes DIGTNMAP ou DIGTCRIT. Se esse for o caso, retorna o ID do usuário associado ao filtro de correspondência mais específico.

**Nota:** Aconselha-se não usar filtros de nomes para certificados usados pelo Developer para System z, pois esses filtros mapeiam todos os certificados para um único ID de usuário. O resultado é que todos os seus usuários do Developer para System z efetuarão logon com o mesmo ID de usuário.

3. Se não houver nenhum filtro de nome correspondente, o RACF localizará a extensão de certificado `HostIdMappings` e extrairá o par de ID de usuário e nome do host integrado. Se localizado e validado, o RACF retornará o ID do usuário definido na extensão `HostIdMappings`.

O par ID do usuário e nome do host é válido se todas estas condições forem verdadeiras:

- O certificado da CA usado para assinar esse certificado é marcado como `HIGHTRUST` na classe `DIGTCERT`.
- O ID do usuário armazenado na extensão tem um comprimento válido (1 a 8 caracteres).
- O ID do usuário designado ao daemon do RSE tem (pelo menos) autoridade `READ` para o perfil `IRR.HOST.hostname` na classe `SERVAUTH`, onde `hostname` é o nome do host armazenado na extensão. É geralmente um nome de domínio, como `CDFMVS08.RALEIGH.IBM.COM`.

A definição da extensão `HostIdMappings` na sintaxe ASN.1 é:

```
id-ce-hostIdMappings OBJECT IDENTIFIER ::= { 1 3 18 0 2 18 1 }
HostIdMappings ::= SET OF HostIdMapping
HostIdMapping ::= SEQUENCE {
    hostName          IMPLICIT[1] IA5String,
    subjectId         IMPLICIT[2] IA5String,
    proofOfIdPossession IdProof OPTIONAL
}
IdProof ::= SEQUENCE {
    secret            OCTET STRING,
    encryptionAlgorithm OBJECT IDENTIFIER
}
```

**Nota:** Uma extensão `HostIdMappings` não é honrada se o ID do usuário de destino tiver sido criado após o início do período de validade para o certificado contendo a extensão `HostIdMappings`. Portanto, se você estiver criando IDs de usuários especificamente para certificados com extensões `HostIdMappings`, certifique-se de que você tenha criado os IDs de usuários antes de os pedidos de certificados serem enviados.

Consulte *Security Server RACF Security Administrator's Guide* (SA22-7683) para obter informações adicionais sobre certificados X.509, como são gerenciados pelo RACF e como definir filtros de nomes de certificados. Consulte o *Security Server RACF Command Language Reference* (SA22-7687) para obter informações adicionais sobre o comando **RACDCERT**.

## Autenticação por Daemon do RSE

O Developer para System z pode realizar autenticação básica de certificado X.509 sem depender de seu produto de segurança. A autenticação realizada pelo daemon do RSE requer que um ID do usuário e nome do host sejam definidos em uma extensão de certificado e está ativa somente se a diretiva `enable.certificate.mapping` em `rsed.envvars` estiver configurada para `FALSE`.

Essa função deverá ser usada se o seu produto de segurança não suportar autenticação de um usuário com base em um certificado X.509 ou se o seu certificado for causar falha no(s) teste(s) feito(s) por seu produto de segurança (por exemplo, o certificado possui um identificador falho para a extensão `HostIdMappings` e não há nenhum filtro ou definição em `DIGTCERT`).

O cliente consultará o usuário pelo identificador da extensão (OID) a ser usado, que é, por padrão, o OID de `HostIdMappings`, {1 3 18 0 2 18 1}.

O daemon do RSE extrairá o ID do usuário e o nome do host do mesmo usando o formato da extensão HostIdMappings. Esse formato está descrito em “Autenticação por Software de Segurança” na página 170.

O par ID do usuário e nome do host é válido se todas estas condições forem verdadeiras:

- O ID do usuário armazenado na extensão tem um comprimento válido (1 a 8 caracteres).
- O ID do usuário designado ao daemon do RSE tem (pelo menos) autoridade READ para o perfil IRR.HOST.hostname na classe SERVAUTH, onde hostname é o nome do host armazenado na extensão. É geralmente um nome de domínio, como CDFMVS08.RALEIGH.IBM.COM.

**Atenção:** Depende do administrador de segurança assegurar que todas as CAs conhecidas do daemon RSE sejam altamente confiáveis, já que o daemon RSE não pode verificar se aquele que assinou o certificado de cliente é altamente confiável ou apenas confiável. Consulte “Validação da Autoridade de Certificação (CA)” na página 169 para obter informações adicionais sobre certificados de CA acessíveis.

---

## Verificação de Port Of Entry (POE)

O Developer para System z suporta a verificação de Port Of Entry (POE), que permite que o host acesse apenas os endereços TCP/IP confiáveis. Esse recurso fica desativado por padrão e requer a definição do perfil de segurança BPX.POE, conforme mostrado nos seguintes comandos RACF de amostra:

- RDEFINE FACILITY BPX.POE UACC(NONE)
- PERMIT BPX.POE CLASS(FACILITY) ACCESS(READ) ID(STCRSE)
- SETROPTS RACLIST(FACILITY) REFRESH

### **Nota:**

- O RSE deve ser configurado para utilizar o POE removendo-se o comentário da opção “enable.port.of.entry=true” em rsed.envvars, conforme documentado em “Definindo Parâmetros Extras de Inicialização Java com \_RSE\_JAVAOPTS” na página 42.
- O ID de usuário do RSE STCRSE requer o UID(0) quando esse perfil não está definido e a verificação de POE está ativada em rsed.envvars.
- A definição de BPX.POE afetará outros aplicativos TC/PIP que suportam a verificação de POE, como INETD.
- As zonas de segurança (perfis EZB.NETACCESS.\*\*, que são os intervalos de endereços IP) devem ser configuradas na classe SERVAUTH para utilizar toda a força da verificação de POE.

Consulte o *Communications Server IP Configuration Guide* (SC31-8775) para obter informações adicionais sobre o controle de acesso à rede usando a verificação de POE.

---

## Segurança do CICS

O Developer para System z permite, através do Application Deployment Manager, que administradores do CICS controlem quais definições de recurso do CICS podem ser editadas pelo desenvolvedor, seus valores padrão e a exibição de uma definição de recurso do CICS por meio de um servidor CICS Resource Definition



(CRD). Consulte Capítulo 15, “Considerações sobre o CICSTS”, na página 251 para obter informações adicionais sobre as definições de segurança necessárias do CICS TS.

## Repositório do CRD

O conjunto de dados de VSAM do repositório do servidor CRD contém todas as definições de recurso padrão e deve, portanto, ser protegido contra atualizações, mas os desenvolvedores devem ter permissão para ler os valores armazenados aqui.

## Transações do CICS

O Developer para System z fornece várias transações que são usadas pelo servidor CRD durante a definição e a consulta de recursos do CICS. Quando a transação é conectada, a verificação de segurança de recurso do CICS, se ativada, garante que o ID do usuário esteja autorizado para executar o ID de transação.

## Comunicação Criptografada por SSL

O cliente Application Deployment Manager usa os Serviços da Web do CICS TS ou a interface RESTful para chamar o servidor CRD. O uso de SSL para essa comunicação é controlado pela definição TCPIPService do CICS TS, conforme documentado no *RACF Security Guide for CICS TS*.

---

## Segurança de SCLM

O serviço SCLM Developer Toolkit oferece funcionalidade de segurança opcional para as funções Construir, Promover e Implementar.

Se a segurança for ativada para uma função pelo administrador de SCLM, as chamadas SAF serão feitas para verificar a autoridade para executar a função protegida com o ID do usuário do responsável pela chamada ou do substituto.

Consulte *SCLM Developer Toolkit Administrator's Guide* (SC23-9801), para obter informações adicionais sobre as definições de segurança SCLM necessárias.

---

## Arquivos de Configuração do Developer para System z

Há vários arquivos de configuração do Developer para System z cujas diretivas afetam a configuração da segurança. Com base nas informações deste capítulo, o administrador de segurança e o programador de sistemas podem decidir quais devem ser as configurações para as diretivas a seguir.

### JES Job Monitor - FEJCNFG

- `LIMIT_COMMANDS={USERID | LIMITED | NOLIMIT }`

Definir em relação a quais ações as tarefas podem ser realizadas (excluindo navegação e envio). Para obter informações adicionais, consulte “Ações nas Tarefas - Limitações de Destino” na página 164.

- `LIMIT_VIEW={USERID | NOLIMIT}`

Definir quais arquivos de spool podem ser navegados. Para obter informações adicionais, consulte “Acesso aos Arquivos de Spool” na página 167.

- `APPLID={FEKAPPL | *}`

ID do aplicativo usado para criação/validação do PassTicket. Para obter informações adicionais, consulte “Usando os PassTickets” na página 162.

**Nota:** Detalhes sobre essas e outras diretivas FEJCNFG estão disponíveis em “FEJCNFG, Arquivo de Configuração do JES Job Monitor” na página 27.

## RSE - rsed.envvars

- (`_RSE_JAVAOPTS`) `-DDENY_PASSWORD_SAVE={true | false}`  
Nega aos usuários o salvamento de sua senha do host no cliente. Para obter informações adicionais, consulte “Definindo Parâmetros Extras de Inicialização Java com `_RSE_JAVAOPTS`” na página 42.
- (`_RSE_JAVAOPTS`) `-DDSTORE_IDLE_SHUTDOWN_TIMEOUT=value`  
Cronômetro para desconectar clientes inativos. Para obter informações adicionais, consulte “Definindo Parâmetros Extras de Inicialização Java com `_RSE_JAVAOPTS`” na página 42.
- (`_RSE_JAVAOPTS`) `-DAPPLID={FEKAPPL | *}`  
ID do aplicativo usado para criação/validação do PassTicket. Para obter informações adicionais, consulte “Usando os PassTickets” na página 162.
- (`_RSE_JAVAOPTS`) `-Denable.port.of.entry={true | false}`  
Ativar a verificação de Port Of Entry. Para obter informações adicionais, consulte “Verificação de Port Of Entry (POE)” na página 172.
- (`_RSE_JAVAOPTS`) `-Denable.certificate.mapping={true | false}`  
Use o seu produto de segurança para autenticar usuários com um certificado X.509. Para obter informações adicionais, consulte “Autenticação de Cliente Usando Certificados X.509” na página 168.
- (`_RSE_JAVAOPTS`) `-Ddaemon.log={/var/rdz/logs | *}`  
Local dos arquivos de log de auditoria. Para obter informações adicionais, consulte “Criação de Log de Auditoria” na página 163.

**Nota:** Detalhes sobre essas e outras diretivas `rsed.envvars` estão disponíveis em “Arquivo de Configuração do RSE `rsed.envvars`” na página 31.

## RSE - ssl.properties

- `daemon_keydb_file={SAF key ring name | gskkyman key database name}`  
Local do certificado do daemon RSE. Para obter informações adicionais, consulte “Comunicação Criptografada por SSL” na página 167.
- `daemon_key_label=certificate label`  
Nome do certificado do daemon RSE. Para obter informações adicionais, consulte “Comunicação Criptografada por SSL” na página 167.
- `server_keystore_file={SAF key ring name | Java key store name}`  
Local do certificado do servidor RSE. Para obter informações adicionais, consulte “Comunicação Criptografada por SSL” na página 167.
- `server_keystore_label=certificate label`  
Nome do certificado do servidor RSE. Para obter informações adicionais, consulte “Comunicação Criptografada por SSL” na página 167.
- `server_keystore_type={JKS | JCECARCFKS | JCECCARCFKS}`  
Tipo de armazenamento de chaves usado (armazenamento de chaves Java ou conjunto de chaves SAF). Para obter informações adicionais, consulte “Comunicação Criptografada por SSL” na página 167.

**Nota:** Detalhes sobre essas e outras diretivas `ssl.properties` estão disponíveis em “(Opcional) Criptografia SSL do RSE” na página 91.



## Definições de Segurança

Customize e envie o membro de amostra FEKRACF, que possui os comandos de amostra do RACF e z/OS UNIX para criar as definições de segurança básicas para o Developer para System z.

FEKRACF está localizado em FEK.#CUST.JCL, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes.

Consulte o *RACF Command Language Reference* (SA22-7687), para obter informações adicionais sobre os comandos RACF.

### Nota:

- Para esses sites que usam CA ACF2™ para z/OS, consulte o seguinte link, <https://support.ca.com/irj/portal/kbtech?ipLogNrow=0&docid=492389&searchID=TEC492389>, para obter detalhes sobre os comandos de segurança necessários para configurar corretamente o Developer para System z.
- Para aqueles sites que usam CA Top Secret® para z/OS, consulte a sua página do produto no site de suporte do CA (<https://support.ca.com>) e verifique o Documento de Conhecimento relacionado do Developer para System z. Este Documento de Conhecimento possui detalhes dos comandos de segurança necessários para configurar apropriadamente o Developer para System z.

As seções a seguir descrevem as etapas necessárias, a configuração opcional e as possíveis alternativas.

## Requisitos e Lista de Verificação

Para concluir a configuração de segurança, o administrador de segurança precisa conhecer os valores listados na Tabela 26. Esses valores foram definidos durante as etapas anteriores da instalação e da customização do Developer para System z.

Tabela 26. Variáveis de configuração de segurança

Descrição	<ul style="list-style-type: none"><li>• Valor Padrão</li><li>• Onde encontrar a resposta</li></ul>	Valor
Qualificador de alto nível do produto Developer para System z	<ul style="list-style-type: none"><li>• FEK</li><li>• Instalação SMP/E</li></ul>	
Qualificador de alto nível de customização do Developer para System z	<ul style="list-style-type: none"><li>• FEK.#CUST</li><li>• FEK.SFEKSAMP(FEKSETUP), conforme descrito em “Configuração da Customização” na página 15.</li></ul>	
Nome da tarefa iniciada do JES Job Monitor	<ul style="list-style-type: none"><li>• JMON</li><li>• FEK.#CUST.PROCLIB(JMON), conforme descrito em “Alterações do PROCLIB” na página 21.</li></ul>	

Tabela 26. Variáveis de configuração de segurança (continuação)

Descrição	<ul style="list-style-type: none"> <li>• Valor Padrão</li> <li>• Onde encontrar a resposta</li> </ul>	Valor
Nome da tarefa iniciada do daemon RSE	<ul style="list-style-type: none"> <li>• RSED</li> <li>• FEK.#CUST.PROCLIB(RSED), conforme descrito em “Alterações do PROCLIB” na página 21.</li> </ul>	
Nome da tarefa iniciada do daemon de bloqueio	<ul style="list-style-type: none"> <li>• LOCKD</li> <li>• FEK.#CUST.PROCLIB(LOCKD), conforme descrito em “Alterações do PROCLIB” na página 21.</li> </ul>	
ID do aplicativo	<ul style="list-style-type: none"> <li>• FEKAPPL</li> <li>• /etc/rdz/rsed.envvars, como descrito em “Definindo Parâmetros Extras de Inicialização Java com _RSE_JAVAOPTS” na página 42</li> </ul>	

A lista a seguir é uma visão geral das ações necessárias para concluir a configuração de segurança básica de Developer para System z. Conforme documentado nas seções abaixo, é possível usar diferentes métodos para preencher esses requisitos, dependendo do nível de segurança desejado. Consulte as seções anteriores para obter informações sobre a configuração de segurança de serviços opcionais do Developer para System z.

- “Ativar Configurações e Classes de Segurança”
- “Definição de um Segmento OMVS para os Usuários do Developer para System z” na página 177
- “Definir os Perfis do Conjunto de Dados” na página 178
- “Definir as Tarefas Iniciadas do Developer para System z” na página 180
- “Definir a Segurança de Comando do JES” na página 182
- “Definir o RSE como um Servidor z/OS UNIX Seguro” na página 183
- “Definir Bibliotecas Controladas pelo Programa MVS para o RSE” na página 184
- “Definir Proteção de Aplicativo para RSE” na página 185
- “Definir o Suporte PassTicket para o RSE” na página 185
- “Definir Arquivos Controlados pelo Programa z/OS UNIX para o RSE” na página 186
- “Verificar Configurações de Segurança” na página 186

## Ativar Configurações e Classes de Segurança

O Developer para System z utiliza uma variedade de mecanismos de segurança para assegurar um ambiente de host seguro e controlado para o cliente. Para fazer isso, várias classes e configurações de segurança devem estar ativas, conforme mostrado com os comandos de amostra do RACF a seguir:

- Exibir configurações atuais
  - SETROPTS LIST
- Ativar classe de recurso do z/OS UNIX e os perfis de certificados digitais

- SETROPTS GENERIC(FACILITY)
- SETROPTS CLASSACT(FACILITY) RACLIST(FACILITY)
- Ativar definições de tarefa iniciada
  - SETROPTS GENERIC(STARTED)
  - RDEFINE STARTED \*\* STDATA(USER(=MEMBER) GROUP(STCGROUP) TRACE(YES))
  - SETROPTS CLASSACT(STARTED) RACLIST(STARTED)
- Ativar a segurança do console para JES Job Monitor
  - SETROPTS GENERIC(CONSOLE)
  - SETROPTS CLASSACT(CONSOLE) RACLIST(CONSOLE)
- Ativar a proteção do comando do operador para JES Job Monitor
  - SETROPTS GENERIC(OPERCMDS)
  - SETROPTS CLASSACT(OPERCMDS) RACLIST(OPERCMDS)
- Ativar a proteção do aplicativo para RSE
  - SETROPTS GENERIC(APPL)
  - SETROPTS CLASSACT(APPL) RACLIST(APPL)
- Ativar a conexão protegida usando PassTickets para RSE
  - SETROPTS GENERIC(PKTCDATA)
  - SETROPTS CLASSACT(PKTCDATA) RACLIST(PKTCDATA)
- Ativar o controle de programa para garantir que apenas o código confiável possa ser carregado pelo RSE
  - RDEFINE PROGRAM \*\* ADDMEM('SYS1.CMDLIB'//NOPADCHK) UACC(READ)
  - SETROPTS WHEN(PROGRAM)

**Nota:** Não crie o perfil \*\* se você já tiver um perfil \* na classe PROGRAM. Ele confunde e complica o caminho de procura usado pelo software de segurança. Nesse caso, você deve mesclar as definições \* existentes com a \*\* nova. A IBM recomenda utilizar o perfil \*\*, conforme documentado em *Security Server RACF Security Administrator's Guide* (SA22-7683).

**Atenção:** Alguns produtos, como o FTP, precisarão ser controlados pelo programa se "WHEN PROGRAM" estiver ativo. Teste isto antes de ativá-lo em um sistema de produção.

- (Opcional) Ative o X.509 HostIdMappings e o suporte Port Of Entry (POE) estendido
  - SETROPTS GENERIC(SERVAUTH)
  - SETROPTS CLASSACT(SERVAUTH) RACLIST(SERVAUTH)

## Definição de um Segmento OMVS para os Usuários do Developer para System z

Um segmento RACF OMVS (ou equivalente) que especifica um ID de usuário (UID), um diretório inicial e um comando shell z/OS UNIX válidos diferentes de zero, devem ser definidos para cada usuário do Developer para System z. Seus grupos padrão também requerem um segmento OMVS com um ID do grupo.

Substitua os seguintes marcadores #userid, #user-identifier, #group-name e #group-identifier dos comandos RACF de amostra por valores reais:

- ALTUSER #userid  
OMVS(UID(#user-identifier) HOME(/u/#userid) PROGRAM(/bin/sh) NOASSIZEMAX)
- ALTGROUP #group-name OMVS(GID(#group-identifier))

Embora isso não seja recomendado, você pode utilizar o segmento OMVS compartilhado definido no perfil BPX.DEFAULT.USER da classe FACILITY para atender aos requisitos do segmento OMVS para Developer para System z.

## Definir os Perfis do Conjunto de Dados

O acesso READ para usuários e ALTER para programadores de sistema é suficiente para a maioria dos conjuntos de dados do Developer para System z. Substitua o marcador #sysprog por IDs de usuário válidos ou nomes de grupos do RACF. Além disso, pergunte ao programador de sistema que instalou e configurou o produto os nomes corretos do conjunto de dados. FEK é o qualificador de alto nível padrão usado durante a instalação e FEK.#CUST é o qualificador de alto nível padrão para conjuntos de dados criados durante o processo de customização.

- ADDGROUP (FEK) OWNER(IBMUSER) SUPGROUP(SYS1)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - HLQ STUB')
- ADDSD 'FEK.\*.\*\*' UACC(READ)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
- PERMIT 'FEK.\*.\*\*' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
- SETROPTS GENERIC(DATASET) REFRESH

### Nota:

- É altamente recomendado proteger FEK.SFEKAUTH contra atualizações, já que esse conjunto de dados é autorizado pelo APF. O mesmo ocorre para FEK.SFEKLOAD e FEK.SFEKLPA que, nesse caso, esses conjuntos de dados são controlados pelo programa.
- Os comandos de amostra nesta publicação e na tarefa FEKRACF assumem que a Enhanced Generic Naming (EGN) esteja ativa. Isso permite o uso do qualificador \*\* para representar qualquer número de qualificadores na classe DATASET. Substitua \*\* por \* se a EGN não estiver ativa em seu sistema. Consulte *Security Server RACF Security Administrator's Guide* (SA22-7683) para obter informações adicionais sobre EGN.

Alguns dos componentes opcionais do Developer para System z exigem perfis adicionais do conjunto de dados de segurança. Substitua os marcadores #sysprog, #ram-developer e #cicsadmin por um ID de usuário válido ou nomes de grupos RACF:

- Se a tradução de nome longo/abreviado do SCLM Developer Toolkit for usada, os usuários precisarão de acesso UPDATE ao VSAM de mapeamento, FEK.#CUST.LSTRANS.FILE.
  - ADDSD 'FEK.#CUST.LSTRANS.\*.\*\*' UACC(UPDATE)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - SCLMDT')
  - PERMIT 'FEK.#CUST.LSTRANS.\*.\*\*' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
  - SETROPTS GENERIC(DATASET) REFRESH
- Os desenvolvedores CARMA RAM (Repository Access Manager) exigem acesso de UPDATE ao CARMA VSAMs, FEK.#CUST.CRA\*.
  - ADDSD 'FEK.#CUST.CRA\*.\*' UACC(READ)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - CARMA')
  - PERMIT 'FEK.#CUST.CRA\*.\*' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
  - PERMIT 'FEK.#CUST.CRA\*.\*' CLASS(DATASET) ACCESS(UPDATE) ID(#ram-developer)
  - SETROPTS GENERIC(DATASET) REFRESH
- Se o servidor CRD do Application Deployment Manager (CICS Resource Definition) for usado, então os administradores do CICS exigirão acesso de UPDATE para ao VSAM do repositório do CRD.
  - ADDSD 'FEK.#CUST.ADNREP\*.\*' UACC(READ)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - ADN')
  - PERMIT 'FEK.#CUST.ADNREP\*.\*' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
  - PERMIT 'FEK.#CUST.ADNREP\*.\*' CLASS(DATASET) ACCESS(UPDATE) ID(#cicsadmin)
  - SETROPTS GENERIC(DATASET) REFRESH

- Se o repositório de manifesto do Application Deployment Manager estiver definido, então todos os usuários do CICS Transaction Server exigirão acesso de UPDATE ao VSAM do repositório de manifesto.
  - ADDSD 'FEK.#CUST.ADNMAN\*.\*' UACC(UPDATE)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - ADN')
  - PERMIT 'FEK.#CUST.ADNMAN\*.\*' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
  - SETROPTS GENERIC(DATASET) REFRESH

Use os comandos de amostra do RACF a seguir para obter uma configuração mais segura onde o acesso READ também é controlado.

- proteção do conjunto de dados uacc(none)
  - ADDGROUP (FEK)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - HLQ STUB')  
OWNER(IBMUSER) SUPGROUP(SYS1)"
  - ADDSD 'FEK.\*.\*' UACC(NONE)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
  - ADDSD 'FEK.SFEKAUTH' UACC(NONE)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
  - ADDSD 'FEK.SFEKLOAD' UACC(NONE)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
  - ADDSD 'FEK.SFEKPROC' UACC(NONE)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
  - ADDSD 'FEK.#CUST.PARMLIB' UACC(NONE)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
  - ADDSD 'FEK.#CUST.CNTL' UACC(NONE)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
  - ADDSD 'FEK.#CUST.LSTRANS\*.\*' UACC(NONE)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - SCLMDT')
  - ADDSD 'FEK.#CUST.CRA\*.\*' UACC(NONE)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - CARMA')
  - ADDSD 'FEK.#CUST.ADNREP\*.\*' UACC(NONE)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - ADN')
  - ADDSD 'FEK.#CUST.ADNMAN\*.\*' UACC(NONE)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - ADN')
- permitir que o programador de sistema gerencie todas as bibliotecas
  - PERMIT 'FEK.\*.\*' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
  - PERMIT 'FEK.SFEKAUTH' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
  - PERMIT 'FEK.SFEKLOAD' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
  - PERMIT 'FEK.SFEKLOAD' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
  - PERMIT 'FEK.SFEKLOAD' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
  - PERMIT 'FEK.SFEKPROC' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
  - PERMIT 'FEK.#CUST.PARMLIB' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
  - PERMIT 'FEK.#CUST.CNTL' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
  - PERMIT 'FEK.#CUST.LSTRANS\*.\*' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
  - PERMIT 'FEK.#CUST.CRA\*.\*' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
  - PERMIT 'FEK.#CUST.ADNREP\*.\*' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
  - PERMIT 'FEK.#CUST.ADNMAN\*.\*' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
- permitir que os clientes acessem as bibliotecas de carregamento e exec
  - PERMIT 'FEK.SFEKAUTH' CLASS(DATASET) ACCESS(READ) ID(\*)
  - PERMIT 'FEK.SFEKLOAD' CLASS(DATASET) ACCESS(READ) ID(\*)
  - PERMIT 'FEK.SFEKPROC' CLASS(DATASET) ACCESS(READ) ID(\*)
  - PERMIT 'FEK.#CUST.CNTL' CLASS(DATASET) ACCESS(READ) ID(\*)

**Nota:** Não é necessária nenhuma permissão para FEK.SFEKLPA, já que todo código que reside no LPA é acessível por todos.

- permitir que o JES Job Monitor acesse a biblioteca de carregamento & de parâmetros
  - PERMIT 'FEK.SFEKAUTH' CLASS(DATASET) ACCESS(READ) ID(STCJMON)

- PERMIT 'FEK.#CUST.PARMLIB' CLASS(DATASET) ACCESS(READ) ID(STCJMON)
- (opcional) permitir que os clientes atualizem o VSAM de conversão de nomes longos/abreviados para o SCLMDT
  - PERMIT 'FEK.#CUST.LSTRANS\*.\*\*' CLASS(DATASET) ACCESS(UPDATE) ID(\*)
- (opcional) permitir que os desenvolvedores de RAM atualizem os CARMA VSAMs do CARMA
  - PERMIT 'FEK.#CUST.CRA\*.\*\*' CLASS(DATASET) ACCESS(UPDATE) ID(#ram-developer)
- (opcional) permitir que usuários do CICS leiam o VSAM do repositório do CRD para o Application Deployment Manager
  - PERMIT 'FEK.#CUST.ADNREP\*.\*\*' CLASS(DATASET) ACCESS(READ) ID(\*)
- (opcional) permitir que administradores do CICS atualizem o VSAM do repositório do CRD para Application Deployment Manager
  - PERMIT 'FEK.#CUST.ADNREP\*.\*\*' CLASS(DATASET) ACCESS(UPDATE) ID(#cicsadmin)
- (opcional) permitir que os usuários do CICS atualizem o VSAM do repositório de manifesto do Application Deployment Manager
  - PERMIT 'FEK.#CUST.ADNMAN\*.\*\*' CLASS(DATASET) ACCESS(UPDATE) ID(\*)
- (opcional) permitir que o servidor CICS TS acesse a biblioteca de carregamento para bidirecional e Application Deployment Manager
  - PERMIT 'FEK.SFEKLOAD' CLASS(DATASET) ACCESS(READ) ID(#cicsts)
- (opcional) permitir que o servidor DB2 acesse a biblioteca de exec para o construtor de procedimento armazenado do DB2
  - PERMIT 'FEK.SFEKPROC' CLASS(DATASET) ACCESS(READ) ID(#db2)
- ativar perfis de segurança
  - SETROPTS GENERIC(DATASET) REFRESH

Ao controlar o acesso READ para conjuntos de dados do sistema, você deve fornecer aos servidores Developer para System z e usuários a permissão READ para os seguintes conjuntos de dados:

- CEE.SCEERUN
- CEE.SCEERUN2
- CBC.SCLBDLL
- ISP.SISPLoad
- ISP.SISPLPA
- SYS1.LINKLIB
- SYS1.SIEALNKE
- REXX.V1R4M0.SEAGLPA

**Nota:** Quando você utiliza a Biblioteca Alternativa para o pacote do produto REXX, o nome da biblioteca de tempo de execução REXX padrão é REXX.\*.SEAGALT. em vez de REXX.\*.SEAGLPA, conforme usado na amostra acima.

## Definir as Tarefas Iniciadas do Developer para System z

Os comandos de amostra RACF a seguir criam as tarefas iniciadas JMON, RSED e LOCKD, com IDs de usuário protegidos (STCJMON, STCRSE e STCLOCK, respectivamente) e agrupam o STCGROUP designado a eles. Substitua os marcadores #group-id e #user-id-\* pelos IDs de OMVS válidos.

- ADDGROUP STCGROUP OMVS(GID(#group-id))  
DATA('GROUP WITH OMVS SEGMENT FOR STARTED TASKS')

- ADDUSER STCJMON DFLTGROUP(STCGROUP) NOPASSWORD NAME('RDZ - JES JOBMONITOR')  
OMVS(UID(#user-id-jmon) HOME(/tmp) PROGRAM(/bin/sh) NOASSIZEMAX  
NOTHREADSMAX)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
- ADDUSER STCRSE DFLTGROUP(STCGROUP) NOPASSWORD NAME('RDZ - RSE DAEMON')  
OMVS(UID(#user-id-rse) HOME(/tmp) PROGRAM(/bin/sh) ASSIZEMAX(2147483647)  
NOTHREADSMAX)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
- ADDUSER STCLOCK DFLTGROUP(STCGROUP) NOPASSWORD NAME('RDZ - LOCK DAEMON')  
OMVS(UID(#user-id-lock) HOME(/tmp) PROGRAM(/bin/sh) NOASSIZEMAX)  
NOTHREADSMAX)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
- RDEFINE STARTED JMON.\* DATA('RDZ - JES JOBMONITOR')  
STDATA(USER(STCJMON) GROUP(STCGROUP) TRUSTED(NO))
- RDEFINE STARTED RSED.\* DATA('RDZ - RSE DAEMON')  
STDATA(USER(STCRSE) GROUP(STCGROUP) TRUSTED(NO))
- RDEFINE STARTED LOCKD.\* DATA('RDZ - LOCK DAEMON')  
STDATA(USER(STCLOCK) GROUP(STCGROUP) TRUSTED(NO))
- SETROPTS RACLIST(STARTED) REFRESH

#### Notas:

1. Assegure-se de que os IDs de usuário das tarefas iniciadas sejam protegidos especificando-se a palavra-chave NOPASSWORD.
2. Assegure-se de que o servidor RSE tenha um uid OMVS exclusivo devido aos privilégios relacionados ao z/OS UNIX concedidos a esse uid.
3. O daemon RSE exige um tamanho de espaço de endereço grande (2GB) para operação adequada. Você deve definir esse valor na variável ASSIZEMAX do segmento OMVS para o ID do usuário STCRSE. Isso para garantir que o daemon RSE obterá o tamanho da região necessário, independentemente de alterações em MAXASSIZE em SYS1.PARMLIB(BPXPRMxx).
4. O RSE exige também um grande número de encadeamentos para operação adequada. Você pode definir o limite na variável THREADSMAX do segmento OMVS do ID do usuário STCRSE. Isso garante que o RSE obterá o limite de encadeamento necessário, independentemente de alterações em MAXTHREADS ou MAXTHREADTASKS em SYS1.PARMLIB(BPXPRMxx). Consulte o Capítulo 13, “Considerações de Ajuste”, na página 211 para determinar o valor correto para o limite de encadeamento.
5. O ID do usuário STCJMON é outro bom candidato para configurar THREADSMAX no segmento OMVS, pois o JES Job Monitor usa um encadeamento por conexão do cliente.

Talvez você queira restringir o ID do usuário STCRSE. Os usuários com o atributo RESTRICTED não podem acessar recursos (MVS) protegidos que eles não estão especificamente autorizados a acessar.

```
ALTUSER STCRSE RESTRICTED
```

Para assegurar que os usuários restritos não obtenham acesso aos recursos do sistema de arquivos z/OS UNIX por meio de “outros” bits de permissão, é necessário definir o perfil RESTRICTED.FILESYS.ACCESS na classe UNIXPRIV com UACC(NONE). Consulte *Security Server RACF Security Administrator's Guide* (SA22-7683) para obter informações adicionais sobre a restrição de IDs de usuários.



**Atenção:** Se você usar IDs de usuário restritos, deverá incluir explicitamente a permissão para acessar um recurso com os comandos **PERMIT** do TSO ou z/OS UNIX **setfacl**. Isso inclui recursos em que a documentação do Developer para System z usa UACC (como o perfil \*\* da classe PROGRAM) ou em que depende das convenções comuns do z/OS UNIX (como se todos tivessem permissão de leitura e execução para bibliotecas Java). Teste isto antes de ativá-lo em um sistema de produção.

## Definir a Segurança de Comando do JES

O JES Job Monitor emite todos os comandos do operador JES solicitados por um usuário por meio de um console MCS estendido (EMCS), cujo nome é controlado com a diretiva `CONSOLE_NAME`, conforme documentado em “FEJJCNFG, Arquivo de Configuração do JES Job Monitor” na página 27.

Os seguintes comandos de amostra RACF oferecem aos usuários do Developer para System z acesso condicional a um conjunto limitado de comandos JES (Suspend, Liberar, Cancelar e Limpar). Os usuários só terão permissão de execução se emitirem os comandos por meio do JES Job Monitor. Substitua o marcador `#console` pelo nome real do console.

- `RDEFINE OPERCMDS MVS.MCSOPER.#console UACC(READ)`  
`DATA('RATIONAL DEVELOPER FOR SYSTEM Z')`
- `RDEFINE OPERCMDS JES.** UACC(NONE)`
- `PERMIT JES.** CLASS(OPERCMDS) ACCESS(UPDATE) WHEN(CONSOLE(JMON)) ID(*)`
- `SETOPTS RACLIST(OPERCMDS) REFRESH`

### Nota:

- O uso do console será permitido se nenhum perfil `MVS.MCSOPER.#console` estiver definido
- A classe `CONSOLE` deverá estar ativa para que `WHEN(CONSOLE(JMON))` funcione, mas não há registro de entrada real de perfil na classe `CONSOLE` para consoles EMCS.
- Não substitua `JMON` pelo nome real do console na cláusula `WHEN(CONSOLE(JMON))`. A palavra-chave `JMON` representa o aplicativo de ponto de entrada, não o nome do console.

**Atenção:** Definir os comandos JES com o acesso universal `NONE` no software de segurança pode afetar outros aplicativos e operações. Teste isto antes de ativá-lo em um sistema de produção.

A Tabela 27 e a Tabela 28 na página 183 mostram os comandos do operador emitidos para JES2 e JES3 e os perfis de segurança distintos que podem ser usados para protegê-los.

*Tabela 27. Comandos do Operador do JES2 Job Monitor*

Ações	Comando	Perfil OPERCMDS	Acesso Necessário
Suspen der	<code>\$Hx(jobid)</code> with x = {J, S or T}	<code>jesname.MODIFYHOLD.BAT</code> <code>jesname.MODIFYHOLD.STC</code> <code>jesname.MODIFYHOLD.TSU</code>	UPDATE
Libe ração	<code>\$Ax(jobid)</code> with x = {J, S or T}	<code>jesname.MODIFYRELEASE.BAT</code> <code>jesname.MODIFYRELEASE.STC</code> <code>jesname.MODIFYRELEASE.TSU</code>	UPDATE



Tabela 27. Comandos do Operador do JES2 Job Monitor (continuação)

Ações	Comando	Perfil OPERCMDS	Acesso Necessário
Cancelar	\$Cx(jobid) with x = {J, S or T}	jesname.CANCEL.BAT jesname.CANCEL.STC jesname.CANCEL.TSU	UPDATE
Limpar	\$Cx(jobid),P with x = {J, S or T}	jesname.CANCEL.BAT jesname.CANCEL.STC jesname.CANCEL.TSU	UPDATE

Tabela 28. Comandos do Operador do JES3 Job Monitor

Ações	Comando	Perfil OPERCMDS	Acesso Necessário
Suspen der	*F,J=jobid,H	jesname.MODIFY.JOB	UPDATE
Libe ração	*F,J=jobid,R	jesname.MODIFY.JOB	UPDATE
Cancelar	*F,J=jobid,C	jesname.MODIFY.JOB	UPDATE
Limpar	*F,J=jobid,C	jesname.MODIFY.JOB	UPDATE

**Nota:**

- Os comandos do operador JES Suspend, Liberar, Cancelar e Limpar e o comando Mostrar JCL só podem ser executados em arquivos de spool pertencentes ao ID de usuário cliente, a não ser que LIMIT\_COMMANDS= com o valor LIMITED ou NOLIMIT esteja especificado no arquivo de configuração do JES Job Monitor. Consulte “Ações nas Tarefas - Limitações de Destino” na página 164 para obter informações adicionais sobre isso.
- Os usuários podem procurar qualquer arquivo em spool, a menos que LIMIT\_VIEW=USERID esteja definido no arquivo de configuração do JES Job Monitor. Consulte “Acesso aos Arquivos de Spool” na página 167 para obter informações adicionais sobre isso.
- Sem autorização para esses comandos do operador, os usuários ainda poderão enviar tarefas e ler a saída da tarefa por meio do JES Job Monitor, se tiverem autoridade suficiente para possíveis perfis que protejam esses recursos (como aqueles das classes JESINPUT, JESJOBS e JESSPOOL).

Supondo que a identidade do servidor JES Job Monitor, criando um console JMON a partir de uma sessão do TSO, seja impedida por seu software de segurança. Embora o console possa ser criado, o ponto de entrada é diferente (JES Job Monitor versus TSO). Os comandos JES emitidos a partir desse console falharão na verificação de segurança se a segurança estiver configurada conforme documentado nesta publicação e o usuário não tiver autoridade para os comandos JES por outros meios.

## Definir o RSE como um Servidor z/OS UNIX Seguro

O RSE requer acesso UPDATE para o perfil BPX.SERVER para criar/excluir o ambiente de segurança do encadeamento do cliente. Se esse perfil não estiver definido, UID(0) será necessário para o RSE.

- RDEFINE FACILITY BPX.SERVER UACC(NONE)
- PERMIT BPX.SERVER CLASS(FACILITY) ACCESS(UPDATE) ID(STCRSE)
- SETROPTS RACLIST(FACILITY) REFRESH

**Atenção:** Definir o perfil BPX.SERVER torna o z/OS UNIX um computador completo da segurança de nível UNIX para a segurança de nível z/OS UNIX, que é mais segura. Isso pode afetar outros aplicativos e operações z/OS UNIX. Teste isto antes de ativá-lo em um sistema de produção. Consulte o *UNIX System Services Planning* (GA22-7800) para obter informações adicionais sobre os diferentes níveis de segurança.

## Definir Bibliotecas Controladas pelo Programa MVS para o RSE

Os servidores com autoridade para BPX.SERVER devem ser executados em um ambiente limpo controlado pelo programa. Isso implica que todos os programas chamados pelo RSE também devem ser controlados pelo programa. Para as bibliotecas de carregamento do MVS, o controle de programa é gerenciado pelo seu software de segurança.

O RSE usa o tempo de execução do sistema (SYS1.LINKLIB), Language Environment' (CEE.SCEERUN\*) e a biblioteca de carregamento do ISPF' TSO/ISPF Client Gateway (ISP.SISPLoad).

- RALTER PROGRAM \*\* UACC(READ) ADDMEM('SYS1.LINKLIB'//NOPADCHK)
- RALTER PROGRAM \*\* UACC(READ) ADDMEM('CEE.SCEERUN'//NOPADCHK)
- RALTER PROGRAM \*\* UACC(READ) ADDMEM('CEE.SCEERUN2'//NOPADCHK)
- RALTER PROGRAM \*\* UACC(READ) ADDMEM('ISP.SISPLoad'//NOPADCHK)
- SETROPTS WHEN(PROGRAM) REFRESH

**Nota:** Não utilize o perfil \*\* se já tiver um perfil \* na classe PROGRAM. Ele confunde e complica o caminho de procura usado pelo software de segurança. Nesse caso, você deve mesclar as definições \* existentes com a \*\* nova. A IBM recomenda usar o perfil \*\*, conforme documentado em *Security Server RACF Security Administrator's Guide* (SA22-7683).

As bibliotecas adicionais a seguir (pré-requisito) devem ser controladas pelo programa para suportar o uso de serviços opcionais. Essa lista não inclui conjuntos de dados que são específicos para um produto com o qual o Developer para System z interage, como a Ferramenta de Depuração IBM.

- Biblioteca de tempo de execução alternativa do REXX (para o SCLM Developer Toolkit)
  - REXX.\*.SEAGALT
- Biblioteca de carregamento do sistema (para criptografia de SSL)
  - SYS1.SIEALNKE
- Biblioteca de carregamento de listener do File Manager (para integração do File Manager)
  - FMN.SFMNMODA

**Nota:** As bibliotecas que são projetadas para colocação de LPA também requerem autorizações de controle de programa se forem acessadas através de LINKLIST ou STEPLIB. Esta publicação documenta o uso das seguintes bibliotecas LPA:

- ISPF (para TSO/ISPF Client Gateway)
  - ISP.SISPLPA
- Biblioteca de tempo de execução REXX (para o SCLM Developer Toolkit)
  - REXX.\*.SEAGLPA

- Developer para System z (para CARMA)
  - FEK.SFEKLPA

## Definir Proteção de Aplicativo para RSE

Durante o logon do cliente, o daemon RSE verifica se um usuário tem permissão para usar o aplicativo.

- RDEFINE APPL FEKAPPL UACC(READ) DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
- SETROPTS RACLIST(APPL) REFRESH

### Nota:

Conforme descrito com mais detalhes em “Definir o Suporte PassTicket para o RSE”, o RSE suporta o uso de um ID do aplicativo diferente de FEKAPPL. A definição de classe do APPL deve corresponder ao ID do aplicativo real usado pelo RSE.

**Atenção:** O pedido de conexão com o cliente falhará se o perfil do aplicativo não estiver definido ou quando o usuário não tiver acesso de LEITURA para o perfil.

## Definir o Suporte PassTicket para o RSE

A senha do cliente, (ou outro meio de identificação como um certificado X.509), somente é usada para verificar a sua identidade na conexão. Depois disso, os PassTickets são usados para manter a segurança do encadeamento.

PassTickets são senha geradas pelo sistema com uma duração de aproximadamente 10 minutos. Os PassTickets baseiam-se em uma chave secreta. Essa chave é um número de 64 bits (16 caracteres hexadecimais). Substitua nos comandos de amostra do RACF, a seguir, o marcador key16 por uma cadeia hexadecimal de 16 caracteres fornecida pelo usuário (caracteres de 0 a 9 e de A a F).

- RDEFINE PTKTDATA FEKAPPL UACC(NONE) SSIGNON(KEYMASKED(key16))  
APPLDATA('NO REPLAY PROTECTION – DO NOT CHANGE')  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
- RDEFINE PTKTDATA IRRPTAUTH.FEKAPPL.\* UACC(NONE)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
- PERMIT IRRPTAUTH.FEKAPPL.\* CLASS(PTKTDATA) ACCESS(UPDATE) ID(STCRSE)
- SETROPTS RACLIST(PTKTDATA) REFRESH

O RSE suporta o uso de um ID do aplicativo diferente de FEKAPPL. Remova o comentário e customize a opção "APPLID=FEKAPPL" no rsed.envvars para ativar isto, como documentado em “Definindo Parâmetros Extras de Inicialização Java com \_RSE\_JAVAOPTS” na página 42. As definições de classe PTKTDATA devem corresponder ao ID do aplicativo real usado pelo RSE.

Não é recomendável usar OMVSAPPL como ID do aplicativo porque ele abrirá a chave secreta para a maioria dos aplicativos do z/OS UNIX. Também não é recomendável usar o ID do aplicativo padrão MVS, que é MVS seguido pelo ID do sistema SMF, porque isto abrirá a chave secreta para a maioria dos aplicativos MVS (incluindo as tarefas em lote).

### Nota:

- Se a classe PTKTDATA já estiver definida, verifique se ela está definida como uma classe genérica antes de criar os perfis listados acima. O suporte para caracteres genéricos da classe PTKTDATA é novo desde o z/OS release 1.7, com a introdução de uma interface Java para PassTickets.

- Substitua o curinga (\*) da definição IRRPTAUTH.FEKAPPL.\* por uma máscara de ID do usuário válida para limitar os IDs do usuário para os quais o RSE pode gerar um PassTicket.
- Dependendo das configurações do RACF, o usuário que define um perfil também poderá estar na lista de acesso do perfil. É aconselhável remover essa permissão dos perfis PTKTDATA.
- O JES Job Monitor e o RSE devem ter o mesmo ID de aplicativo para permitir que o JES Job Monitor avalie os PassTickets apresentados pelo RSE.
- Se o sistema tiver um produto criptográfico instalado e disponível, você poderá criptografar a chave de aplicativo de conexão protegida para proteção adicional. Para isso, utilize a palavra-chave KEYENCRYPTED em vez de KEYMASKED. Consulte o *Security Server RACF Security Administrator's Guide* (SA22-7683) para obter informações adicionais sobre isso.

**Atenção:** O pedido de conexão do cliente falhará se os PassTickets não estiverem configurados corretamente.

## Definir Arquivos Controlados pelo Programa z/OS UNIX para o RSE

Os servidores com autoridade para BPX.SERVER devem ser executados em um ambiente limpo controlado pelo programa. Isso implica que todos os programas chamados pelo RSE também devem ser controlados pelo programa. Para arquivos do z/OS UNIX, o controle do programa é gerenciado pelo comando **extattr**. Para executar esse comando, você precisa de acesso READ para o BPX.FILEATTR.PROGCTL na classe FACILITY ou ser UID(0).

O servidor RSE usa a biblioteca compartilhada Java do RACF (/usr/lib/libIRRacf.so).

- `extattr +p /usr/lib/libIRRacf.so`

### Nota:

- A partir do z/OS 1.9, o /usr/lib/libIRRacf.so é um programa instalado e controlado durante a instalação do SMP/E RACF.
- A partir do z/OS 1.10, o /usr/lib/libIRRacf.so que faz parte do SAF, fornecido com o z/OS de base, também estará disponível para clientes não-RACF.
- A configuração pode ser diferente se você utilizar um produto de segurança diferente do RACF. Consulte a documentação do produto de segurança para obter informações adicionais.
- A instalação do SMP/E do Developer para System z configura o bit de controle de programa para programas RSE internos.
- Use o comando **ls -Eog** z/OS UNIX para exibir o status atual do bit de controle do programa (o arquivo será controlado pelo programa se a letra **p** for mostrada na segunda cadeia).

```
$ ls -Eog /usr/lib/libIRRacf.so
-rwxr-xr-x  aps-  2    69632 Oct  5  2007 /usr/lib/libIRRacf.so
```

## Verificar Configurações de Segurança

Use os seguintes comandos de amostra para exibir os resultados de suas customizações relacionadas à segurança.

- Configurações e classes de segurança

- SETROPTS LIST
- Segmento OMVS para usuários
  - LISTUSER #userid NORACF OMVS
  - LISTGRP #group-name NORACF OMVS
- Perfis do conjunto de dados
  - LISTGRP FEK
  - LISTDSD PREFIX(FEK) ALL
- Tarefas iniciadas
  - LISTGRP STCGROUP OMVS
  - LISTUSER STCJMON OMVS
  - LISTUSER STCRSE OMVS
  - LISTUSER STCLOCK OMVS
  - RLIST STARTED JMON.\* ALL STDATA
  - RLIST STARTED RSED.\* ALL STDATA
  - RLIST STARTED LOCKD.\* ALL STDATA
- Segurança do comando JES
  - RLIST CONSOLE JMON ALL
  - RLIST OPERCMDS MVS.MCSOPER.JMON ALL
  - RLIST OPERCMDS JES%.\*\* ALL
- RSE como um servidor z/OS UNIX seguro
  - RLIST FACILITY BPX.SERVER ALL
- Bibliotecas controladas pelo programa MVS para RSE
  - RLIST PROGRAM \*\* ALL
- Proteção do aplicativo para RSE
  - RLIST APPL FEKAPPL ALL
- Suporte de PassTicket para RSE
  - RLIST PTKTDATA FEKAPPL ALL SSIGNON
  - RLIST PTKTDATA IRRPTAUTH.FEKAPPL.\* ALL
- Arquivos controlados pelo programa z/OS UNIX para RSE
  - ls -E /usr/lib/libIRRRacf.so



---

## Capítulo 11. Entendimento do Developer para System z

O host do Developer para System z consiste em vários componentes que interagem para oferecer acesso ao cliente para os serviços e dados do host. Entender o design desses componentes pode ajudá-lo a tomar as decisões corretas de configuração.

Os seguintes tópicos são abordados neste capítulo:

- “Visão Geral do Componente”
- “RSE como um Aplicativo Java” na página 191
- “Donos das Tarefas” na página 192
- “Fluxo de Conexão” na página 194
- “Daemon de Bloqueio” na página 196
- “Estrutura do Diretório z/OS UNIX” na página 198

---

### Visão Geral do Componente

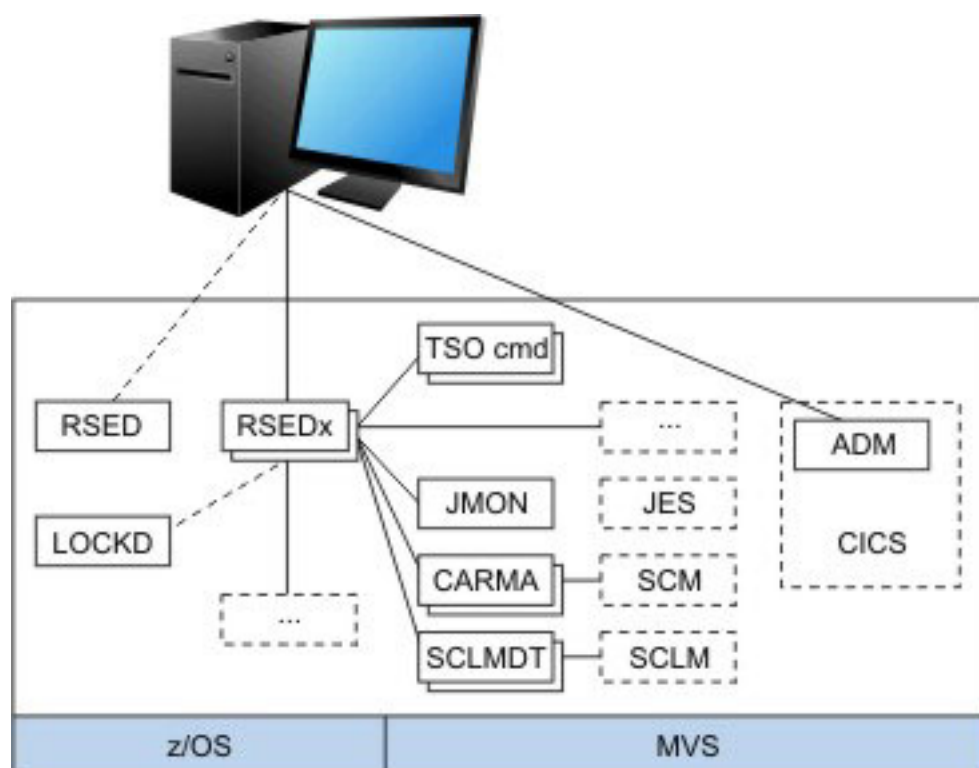


Figura 41. Visão geral do componente

A Figura 41 mostra uma visão geral generalizada do layout do Developer para System z em seu sistema host.

- O Remote Systems Explorer (RSE) fornece os serviços principais, como conectar o cliente ao host e iniciar outros servidores para serviços específicos. O RSE consiste em duas entidades lógicas:
  - O daemon RSE (RSED), que gerencia a configuração de conexão. O daemon RSE também é responsável pela execução em modo de servidor único. Para

fazer isso, o daemon RSE cria um ou mais processos-filho conhecidos como conjuntos de encadeamento do RSE (RSEDx).

- Servidor RSE, que manipula pedidos individuais do cliente. Um servidor RSE é ativado como um encadeamento dentro de um conjunto de encadeamento do RSE.
- O Daemon de Bloqueio (LOCKD) fornece serviços de rastreamento para bloqueios de conjuntos de dados.
- O Serviço TSO Commands (TSO cmd) fornece uma interface como um lote para os comandos TSO e ISPF.
- O JES Job Monitor (JMON) fornece todos os serviços relacionados ao JES.
- O Common Access Repository Manager (CARMA) fornece uma interface para interagir com Software Configuration Managers (SCMs), como o CA Endeavor.
- O SCLM Developer Toolkit (SCLMDT) fornece uma interface para aprimorar e interagir com SCLM.
- O Application Deployment Manager (ADM) fornece vários serviços relacionados ao CICS.
- Mais serviços estão disponíveis, que podem ser fornecidos pelo próprio Developer para System z ou pelo software de correquisito.

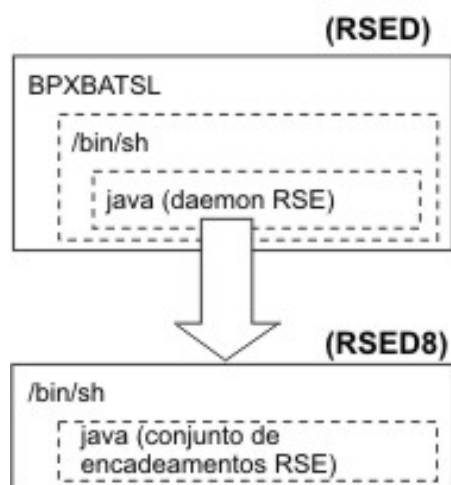
A descrição no parágrafo e a lista mostram a função central designada ao RSE. Com algumas exceções, toda a comunicação do cliente passa pelo RSE. Isso é permitido para uma configuração fácil da rede relacionada à segurança, uma vez que apenas um conjunto limitado de portas é usado para a comunicação entre o cliente e o host.

Para gerenciar as conexões e as cargas de trabalho a partir dos clientes, o RSE é formado por um espaço de endereço do daemon, que controla os espaços de endereços do conjunto de encadeamento. O daemon atua como um ponto focal para fins de conexão e gerenciamento, enquanto os conjuntos de encadeamento processam as cargas de trabalho do cliente. Com base nos valores definidos no arquivo de configuração `rsed.envvars` e na quantidade real de conexões do cliente, vários espaços de endereço do conjunto de encadeamento podem ser iniciados pelo daemon.

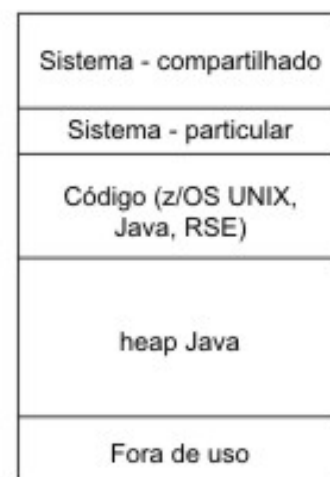


## RSE como um Aplicativo Java

### Processos z/OS UNIX



### Uso de armazenamento Java



JOBNAME	Estado	PID	PPID	Comando
RSED	FILE SYS KERNEL WAIT	50331904	1	BPXBATSL
RSED	WAITING FOR CHILD	67109114	50331904	/bin/sh...
RSED	FILE SYS KERNEL WAIT	50331949	67109114	java...
RSED8	WAITING FOR CHILD	307	50331949	/bin/sh...
RSED8	FILE SYS KERNAL WAIT	308	307	java...

Figura 42. RSE como um aplicativo Java

A Figura 42 mostra uma visualização básica do uso de recursos (processos e armazenamento) pelo RSE.

O RSE é um aplicativo Java, que significa que ele está ativo no ambiente z/OS UNIX. Isso é permitido para facilitar o porting em diferentes plataformas host e comunicação direta com o cliente do Developer para System z, que é também um aplicativo Java (baseado na estrutura do Eclipse). Portanto, o conhecimento básico de como z/OS UNIX e Java funcionam é muito útil quando você tenta compreender o Developer para System z.

No z/OS UNIX, um programa é executada em um processo, que é identificado por um PID (ID do Processo). Cada programa é ativado em seu próprio processo, portanto invocar outro programa criará um novo processo. O processo que iniciou um processo é referenciado com um PPID (PID Pai), o novo processo é chamado de processo-filho. O processo-filho pode ser executado no mesmo espaço de endereço ou pode ser gerado (criado) em um novo espaço de endereço. Um novo processo que é executado no mesmo espaço de endereço pode ser comparado a executar um comando em TSO, enquanto aquele gerado em um novo espaço de endereço é semelhante a enviar uma tarefa em lote.

Observe que um processo pode ser único ou multiencadeado. Em um aplicativo multiencadeado (como o RSE), os diferentes encadeamentos competem por recursos do sistema como se eles fossem espaços de endereço separados (com menos gasto adicional).

Mapeando essas informações de processo para a amostra RSE na Figura 42 na página 191, obtivemos o seguinte fluxo:

1. Quando a tarefa RSED for iniciada, ela executa BPXBATSL, que invoca z/OS UNIX e cria um ambiente de shell – PID 50331904.
2. Neste processo, o shell script rsed.sh é executado, que executa em um processo separado (/bin/sh) – PID 67109114.
3. O shell script configura as variáveis de ambiente definidas em rsed.envvars e executa o Java com os parâmetros necessários para iniciar o daemon RSE – PID 50331949.
4. O daemon RSE fará spawn off do novo shell em um processo-filho (RSED8) – PID 307.
5. Neste shell, as variáveis de ambiente definidas em rsed.envvars são configuradas e o Java é executado com os parâmetros necessários para iniciar o conjunto de encadeamento do RSE – PID 308.

Aplicativos Java, como o RSE, não alocam armazenamento diretamente, mas usam serviços de gerenciamento de memória Java. Esses serviços, como alocação de armazenamento, liberação de armazenamento e coleta de lixo, funcionam dentro dos limites do heap Java. Os tamanhos mínimo e máximo do heap é definido (implicitamente ou explicitamente) durante a inicialização de Java.

Isso significa que a obtenção da maioria do tamanho de espaço de endereço disponível é um ato de equilíbrio da definição de um tamanho grande de heap enquanto deixa espaço suficiente para o z/OS armazenar uma quantidade variável de blocos de controle do sistema (dependendo do número de encadeamentos ativos).

---

## Donos das Tarefas

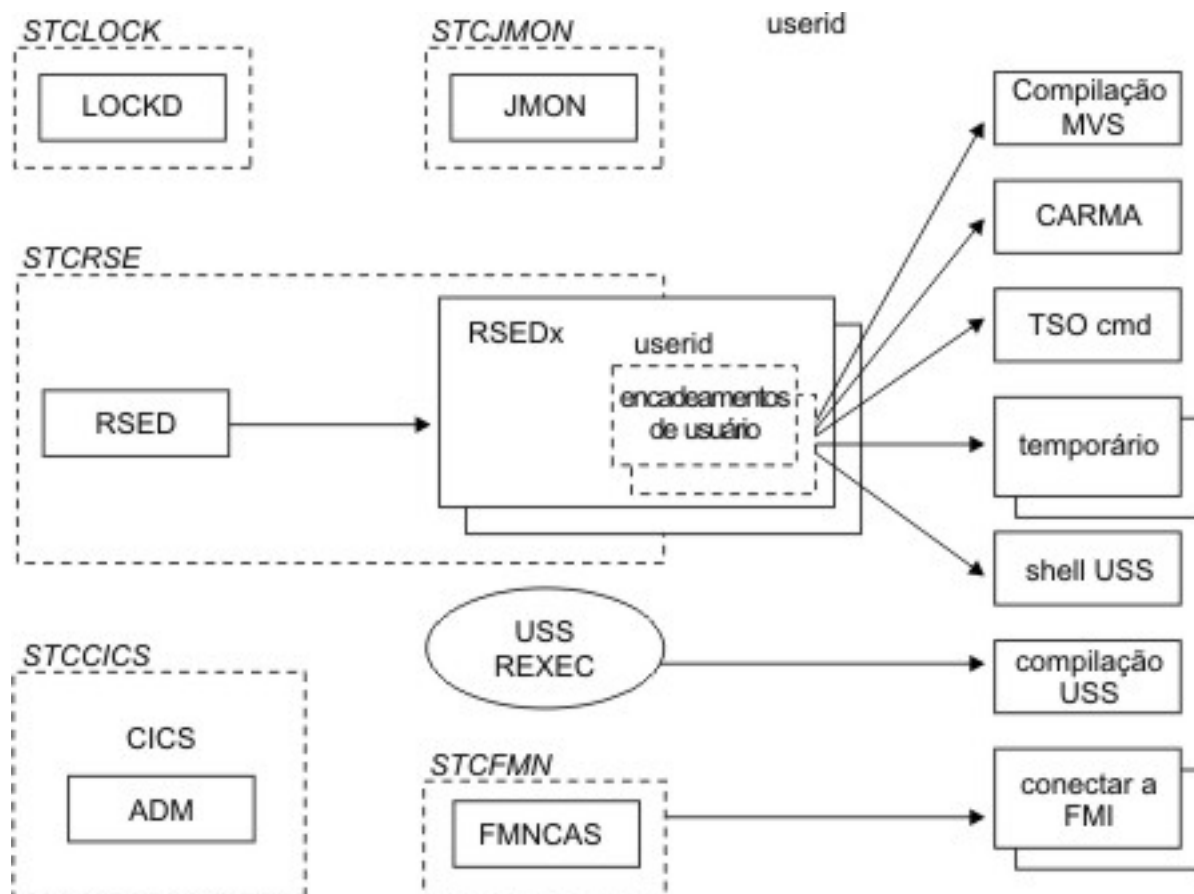


Figura 43. donos das tarefas

O Figura 43 mostra uma visão geral básica do dono das credenciais de segurança usadas por várias tarefas do Developer para System z.

A propriedade de uma tarefa pode ser dividida em duas seções. As tarefas iniciadas são propriedades do ID do usuário que é designado para a tarefa iniciada em seu software de segurança. Todos as outras tarefas, com os conjuntos de encadeamentos (RSEDx) como exceção, são propriedades do ID de usuário cliente.

O Figura 43 mostra as tarefas iniciadas do Developer para system z (LOCKD, JMON and RSED), as tarefas iniciadas de amostra e os serviços do sistema com os quais o Developer para System z se comunica. O Application Deployment Manager (ADM) é ativo dentro de uma região CICS. O FMNCAS é a tarefa iniciada do File Manager. A tag USS REXEC representa o serviço do z/OS UNIX REXEC (ou SSH).

O daemon RSE (RSED) cria um ou mais espaços de endereço do conjunto de encadeamentos (RSEDx) para processar os pedidos dos clientes. Cada conjunto de encadeamentos RSE suporta múltiplos clientes e é propriedade do mesmo usuário como um daemon RSE. Cada cliente possui seus próprios encadeamentos dentro de um conjunto de encadeamentos, e estes encadeamentos são propriedades do ID de usuário cliente.

Dependendo das ações concluídas pelo cliente, um ou mais espaços de endereço adicionais podem ser iniciados, todos propriedades do ID de usuário cliente, para executar uma ação solicitada. Estes espaços de endereço podem ser uma tarefa em lote MVS, uma transação APPC ou um processo-filho do z/OS UNIX. Note que

um processo-filho do z/OS UNIX é ativo em um iniciador z/OS UNIX (BPXAS) e o processo-filho aparece como uma tarefa iniciada no JES.

A criação destes espaços de endereços é mais frequentemente acionada por um encadeamento do usuário em um conjunto de encadeamentos, diretamente ou usando serviços do sistema como ISPF. Mas o espaço de endereço também poderia ser criado por um terceiro. Por exemplo, o File Manager iniciará um espaço de endereço novo para cada conjunto de dados (ou membro) que ele tenha que processar em nome do Developer para System z. z/OS UNIX REXEC ou SSH são envolvidos ao iniciar as construções no z/OS UNIX.

Os espaços de endereços do usuário terminam com a conclusão da tarefa ou quando um cronômetro de inatividade vence. As tarefas iniciadas permanecem ativas. Os espaços de endereços listados em Figura 43 na página 193 permanecem no sistema tempo suficiente para serem visíveis. Porém, é recomendável estar ciente que, devido a maneira com que o z/OS UNIX é designado, há também vários espaços de endereços temporários de curta duração.

## Fluxo de Conexão

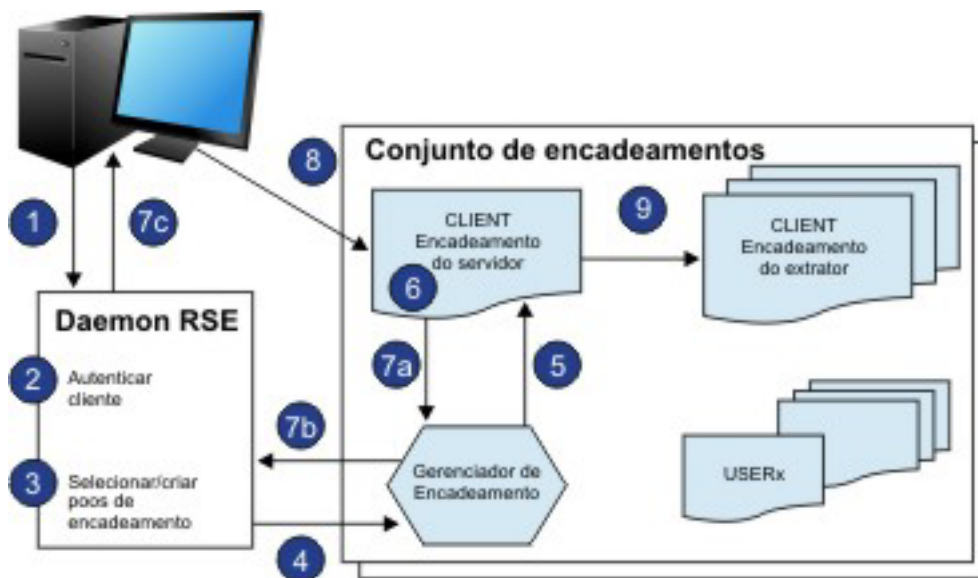


Figura 44. Fluxo de conexão

A Figura 44 mostra uma visão geral esquemática de como um cliente se conecta ao host usando o Developer para System z. Ela explica também resumidamente como os PassTickets são usados.

1. O cliente efetua login no daemon (porta 4035).
2. O daemon RSE autentica o cliente usando as credenciais apresentadas pelo cliente.
3. O daemon RSE seleciona um conjunto de encadeamento existente ou inicia um novo se todos estiverem cheios.
4. O daemon RSE passa o ID do usuário cliente para o conjunto de encadeamento.
5. O conjunto de encadeamento cria um encadeamento do servidor RSE específico do cliente, usando o ID do usuário cliente e um PassTicket para autenticação.

6. O encadeamento do servidor cliente se conecta a uma porta para comunicação futura com o cliente.
7. O encadeamento do servidor cliente retorna o número da porta à qual o cliente deve se conectar.
8. O cliente desconecta do daemon RSE e se conecta ao número da porta fornecido.
9. O encadeamento do servidor cliente inicia outros encadeamentos específicos do usuário (extratores), sempre usando o ID do usuário cliente e um PassTicket para autenticação. Esses encadeamentos fornecem os serviços específicos do usuário necessários pelo cliente.

A descrição acima mostra o design orientado a encadeamentos do RSE. Em vez de iniciar um espaço de endereço por usuário, vários usuários são atendidos por um espaço de endereço de um único conjunto de encadeamento. Dentro do conjunto de encadeamento, cada extrator (um serviço específico do usuário) fica ativo em seu próprio encadeamento com o contexto de segurança do usuário designado a ele, garantindo uma configuração segura. Esse design acomoda um grande número de usuários com uso de recursos limitado, mas não significa que cada cliente usará vários encadeamentos (16 ou mais, dependendo das tarefas executadas).

Do ponto de vista da rede, o Developer para system z atua de forma semelhante ao FTP no modo passivo. O cliente se conecta a um ponto focal (daemon RSE) e, em seguida, elimina a conexão e reconecta ao número de porta fornecido pelo ponto focal. A seguinte lógica controla a seleção da porta que é usada na segunda conexão:

1. Se o cliente especificou um número de porta diferente de zero na guia de propriedades de subsistema, então o servidor RSE usará essa porta para a conexão. Se essa porta não estiver disponível, a conexão falhará.
2. Se `_RSE_PORTRANGE` for especificado em `rsed.envvars`, então o servidor RSE se conectará a uma porta desse intervalo. Se nenhuma porta estiver disponível, a conexão falhará. Observe que o servidor RSE não precisa da porta exclusivamente pela duração da conexão do cliente. Ela só é necessária no momento da expansão entre (servidor) a ligação e (cliente) a conexão que nenhum outro servidor RSE pode se conectar à porta.
3. Se nenhuma limitação for configurada, o servidor RSE se conectará à porta 0. O resultado é que o TCP/IP escolhe o número da porta.

O uso de PassTickets para todos os serviços z/OS que exigem autenticação permite que o Developer para System z invoque esses serviços à vontade, sem armazenar a senha ou solicitá-la constantemente ao usuário. O uso de PassTickets para todos os serviços z/OS permite também métodos de autenticação alternativos durante o logon, como senhas únicas e certificados X.509.

## Daemon de Bloqueio

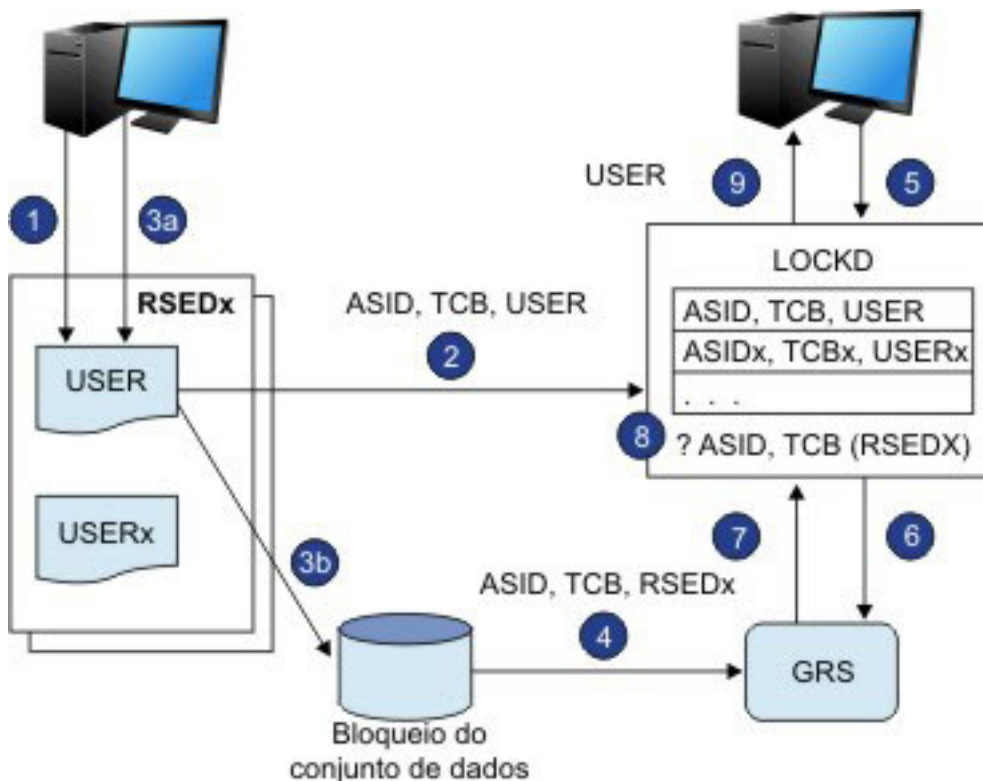


Figura 45. Fluxo do daemon de bloqueio

O Figura 45 mostra uma visão geral esquemática de como um daemon de bloqueio determina qual o cliente do Developer para System z possui um bloqueio do conjunto de dados.

1. O cliente efetua login, que cria um encadeamento do servidor RSE específico do usuário (USER) dentro de um conjunto de encadeamento (RSEDx).
2. O servidor RSE registra um usuário recém-conectado no daemon de bloqueio. As informações de registro contêm o Identificador de Espaço de Endereço (que é o ASID do conjunto de encadeamento), o identificador (específico do usuário) do Task Control Block (TCB) e o ID do usuário.
3. O cliente abre um conjunto de dados na edição, que instrui o servidor RSE para obter um bloqueio exclusivo no conjunto de dados.
4. O sistema registra o ASID, TCB e o nome da tarefa (RSEDx) do solicitante como parte do processo de bloqueio. Essas informações são armazenadas nas filas GRS (Global Resource Serialization).
5. Um operador (ou servidor RSE em nome de um cliente) consulta o daemon de bloqueio do status de bloqueio do conjunto de dados.
6. O daemon de bloqueio varre as filas GRS para aprender se o conjunto de dados está bloqueado.
7. O daemon recupera o ASID, o TCB e o nome da tarefa do proprietário de bloqueio.
8. O ASID e o TCB recuperado são comparados em relação às combinações ASID e TCB de clientes registrados.



9. O ID do usuário do cliente relacionado é retornado ao solicitante quando uma correspondência for encontrada. Caso contrário, o nome da tarefa recuperado de GRS é retornado.

Com uma configuração de um único servidor de Developer para System z, em que vários usuários são designados a um único espaço de endereço do conjunto de encadeamento, o z/OS perdeu a capacidade de controlar quem possui um bloqueio em um conjunto de dados ou membro. Os comandos do sistema param no nível de espaço de endereço, que é o conjunto de encadeamento.

Para especificar esse problema, o Developer para System z fornece o daemon de bloqueio. O daemon de bloqueio pode controlar todos os bloqueios de conjuntos de dados/membros realizados pelos usuários do RSE, bem como os bloqueios realizados por outros produtos, como o ISPF.

O servidor RSE registra um usuário recém-conectado no daemon de bloqueio. As informações de registro contêm o Identificador de Espaço de Endereço (que é o ASID do conjunto de encadeamento), o ID (específico do usuário) do Task Control Block (TCB) e o ID do usuário.

Observe que o registro é feito apenas no momento da conexão, de modo que todos os usuários ativos do RSE antes de o daemon de bloqueio ser iniciado (ou reiniciado) não serão registrados.

Quando o daemon de bloqueio recebe uma consulta de conjunto de dados (por meio de um comando do operador de consulta de modificação ou do cliente por meio do servidor RSE), o daemon varre as filas Global Resource Serialization (GRS) do sistema. Se o ASID e o TCB corresponderem ao de um usuário registrado, o ID do usuário é retornado como um proprietário de bloqueio. Caso contrário, o nome da tarefa/ID do usuário relacionado ao ASID é retornado como proprietário de bloqueio.

Uma mensagem do console (FEK513W) com as informações de registro será exibida se o registro falhar. Isso permite que um operador corresponda os valores em relação à saída de um comando do operador **DISPLAY GRS,RES=(\*,dataset\*)** a fim de localizar o proprietário do bloqueio.

**Nota:** Os registros bem-sucedidos também são listados no DD STDOUT do servidor se `log_level` estiver configurado para 2. Isso é útil para realizar o mapeamento manual para registros bem-sucedidos que foram removidos após um reinício do daemon de bloqueio.

## Liberando um Bloqueio

Sob circunstâncias normais, um conjunto de dados ou um membro fica bloqueado quando o cliente o abre no modo de edição, e liberado quando o cliente fechar a sessão de edição.

Determinadas condições de erro podem evitar que esse mecanismo funcione como designado. Nesse caso, o usuário que mantém o bloqueio pode ser cancelado usando o comando de operação **cancelar modificação** do RSE, conforme descrito no Capítulo 8, “Comandos do Operador”, na página 123. Os bloqueios do conjunto de dados ativo que pertencem a esse usuário são liberados durante o processo.

## Estrutura do Diretório z/OS UNIX

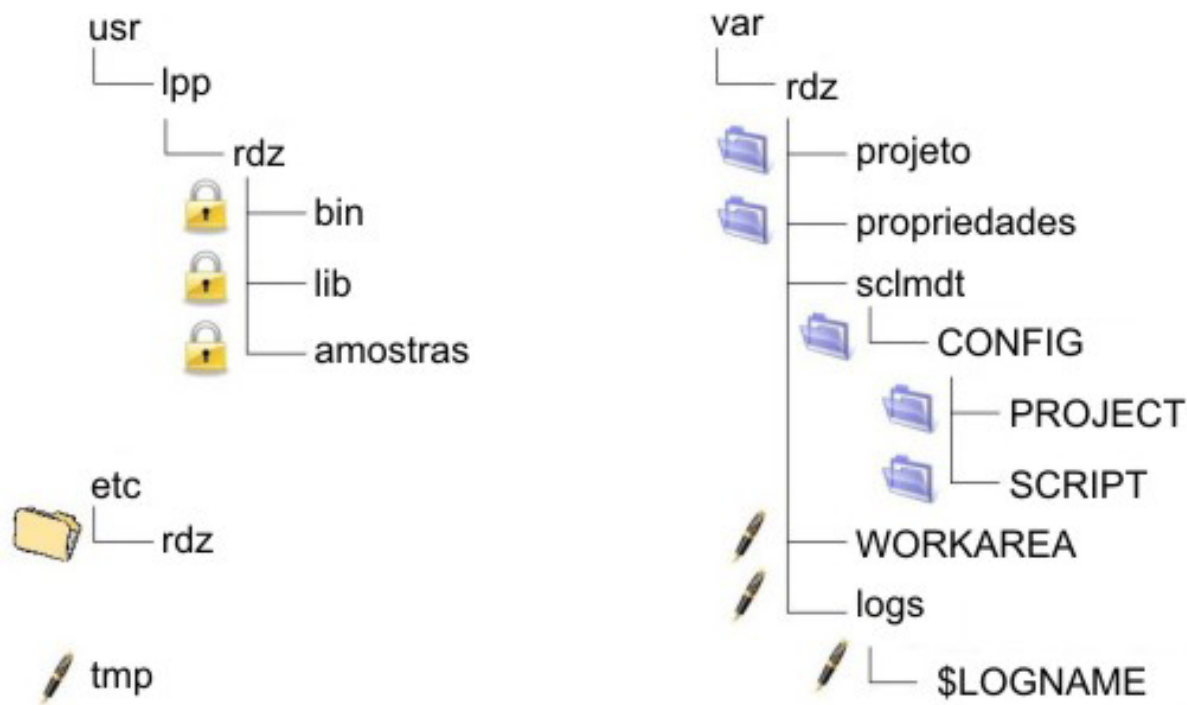


Figura 46. estrutura do diretório z/OS UNIX

O Figura 46 mostra uma visão geral dos diretórios do z/OS UNIX usados pelo Developer para System z. A seguinte lista descreve cada diretório tocado pelo Developer para System z, como o local pode ser alterado e quem mantém os dados dentro dele.

- `/usr/lpp/rdz/` é o caminho-raiz do código do produto Developer para System z. O local real é especificado nas tarefas iniciadas RSED e LOCKD (variável HOME). Os arquivos contidos serão mantidos por SMP/E.
- `/etc/rdz/` mantém o RSE e os arquivos de configuração relacionados ao extrator. O local real é especificado nas tarefas iniciadas RSED e LOCKD (variável CNFG). Os arquivos contidos são mantidos pelo programador de sistema.
- `/var/rdz/sclmdt/CONFIG/` mantém os arquivos de configuração gerais do SCLMDT. O local real é especificado em `r sed.envvars` (variável `SCLMDT_CONF_HOME`). Os arquivos contidos são mantidos pelo administrador do SCLM.
- `/var/rdz/sclmdt/CONFIG/PROJECT/` mantém os arquivos de configuração do projeto SCLMDT. O local real é especificado em `r sed.envvars` (variável `SCLMDT_CONF_HOME`). Os arquivos contidos são mantidos pelo administrador do SCLM.
- `/var/rdz/sclmdt/CONFIG/script/` mantém os scripts relacionados ao SCLMDT que podem ser usados por outros produtos. O local real não é especificado em lugar algum. Os arquivos contidos são mantidos pelo administrador do SCLM.
- `/var/rdz/projects/` mantém os arquivos de definição do projeto baseados em host. O local real é especificado em `projectcfg.properties` (variável `PROJECT-HOME`). Os arquivos contidos são mantidos por um gerenciador de projetos ou pelo desenvolvedor principal.



- `/var/rdz/properties/` mantém os grupos de propriedades baseados em host. O local real é especificado em `propertiescfg.properties` (variáveis `PROPERTY-GROUP` e `DEFAULT-VALUES`). Os arquivos contidos são mantidos por um gerenciador de projetos ou pelo desenvolvedor principal.
- `/var/rdz/logs/` mantém os logs do daemon RSE e dos servidores do conjunto de encadeamento do RSE. O local real é especificado em `rzed.envvars` (variável `daemon.log`). Os arquivos contidos são mantidos por RSE.
- `/var/rdz/logs/$LOGNAME/` mantém os logs específicos do usuário dos extratores e do servidor RSE. O local real é especificado em `rzed.envvars` (variáveis `user.log` e `DSTORE_LOG_DIRECTORY`). Os arquivos contidos são mantidos pelo RSE e pelos extratores.

**Nota:** `/var/rdz/logs/` exige a máscara de bit de permissão 777 para permitir que cada cliente crie seu diretório `$LOGNAME` e armazene os arquivos de log específicos do usuário.

- `/var/rdz/WORKAREA/` é usado pelo TSO/ISPF Client Gateway e SCLMDT do ISPF para transferir dados entre o z/OS UNIX e os espaços de endereço baseados em MVS. O local real é especificado em `rzed.envvars` (variável `_CMDSEV_WORK_HOME`). Os arquivos contidos são mantidos pelo ISPF e SCLMDT.

**Nota:** `/var/rdz/WORKAREA/` exige a máscara de bit de permissão 777 para permitir que cada cliente crie arquivos temporários.

- `/tmp/` é usado pelo TSO/ISPF Client Gateway do ISPF e vários extratores para armazenar dados temporários. O local não é customizável. Os arquivos contidos são mantidos pelo ISPF e pelos extratores. É também o local padrão para arquivos dump Java, que podem ser customizados com a variável `_CEE_DUMPTARG` em `rzed.envvars`.

**Nota:** `/tmp/` exige a máscara de bit de permissão 777 para permitir que cada cliente crie arquivos temporários.

## Atualizar Privilégios para Administradores que Não São de Sistema

Os dados em alguns diretórios, como `/var/rdz/projects/`, são mantidos por administradores que não são de sistema, como gerentes de projetos, que podem não ter muitos privilégios de atualização em z/OS UNIX. Se houver apenas um ID de usuário mantendo os arquivos, isso não será problema depois que o ID do usuário tiver se tornado proprietário do diretório e de tudo no diretório.

```
chown -R IBMUSER /var/rdz/projects/
```

Quando vários IDs de usuário precisarem de permissão de atualização no diretório, você poderá trabalhar com os bits de permissão do grupo.

1. Crie um grupo em seu software de segurança que tenha um segmento OMVS válido e conecte todos os IDs do usuário que precisarem de acesso de atualização. Esse é preferivelmente seu grupo padrão.  

```
ADDGROUP RDZPROJ OMVS(GID(1200))
CONNECT IBMUSER GROUP(RDZPROJ)
ALTUSER IBMUSER DFLTGRP(RDZPROJ)
```
2. Use o comando z/OS UNIX **chgrp** para designar o grupo ao diretório e a todos os arquivos contidos nele. Esse comando deve ser repetido sempre que um novo arquivo for incluído e o ID do grupo desejado não for o grupo padrão do ID do usuário que incluiu o arquivo.  

```
chgrp -R IBMUSER /var/rdz/projects/
```

3. Use o comando z/OS UNIX **chmod** para conceder permissão de atualização ao grupo inteiro para o diretório e a todos os arquivos contidos nele.  
`chmod -R 775 /var/rdz/projects/`

## Capítulo 12. Considerações WLM

Ao contrário dos aplicativos tradicionais z/OS, o Developer para System z não é um aplicativo monolítico que possa ser identificado facilmente para Workload Manager (WLM). O Developer para System z consiste de vários componentes que interagem para fornecer ao cliente acesso para os serviços e dados do host. Como descrito em Capítulo 11, “Entendimento do Developer para System z”, na página 189, alguns destes serviços estão ativos em espaços de endereços diferentes, resultando em classificações WLM diferentes.

Os seguintes tópicos são abordados neste capítulo:

- “Classificação de Carga de Trabalho”
- “Configurando Objetivos” na página 203

### Classificação de Carga de Trabalho

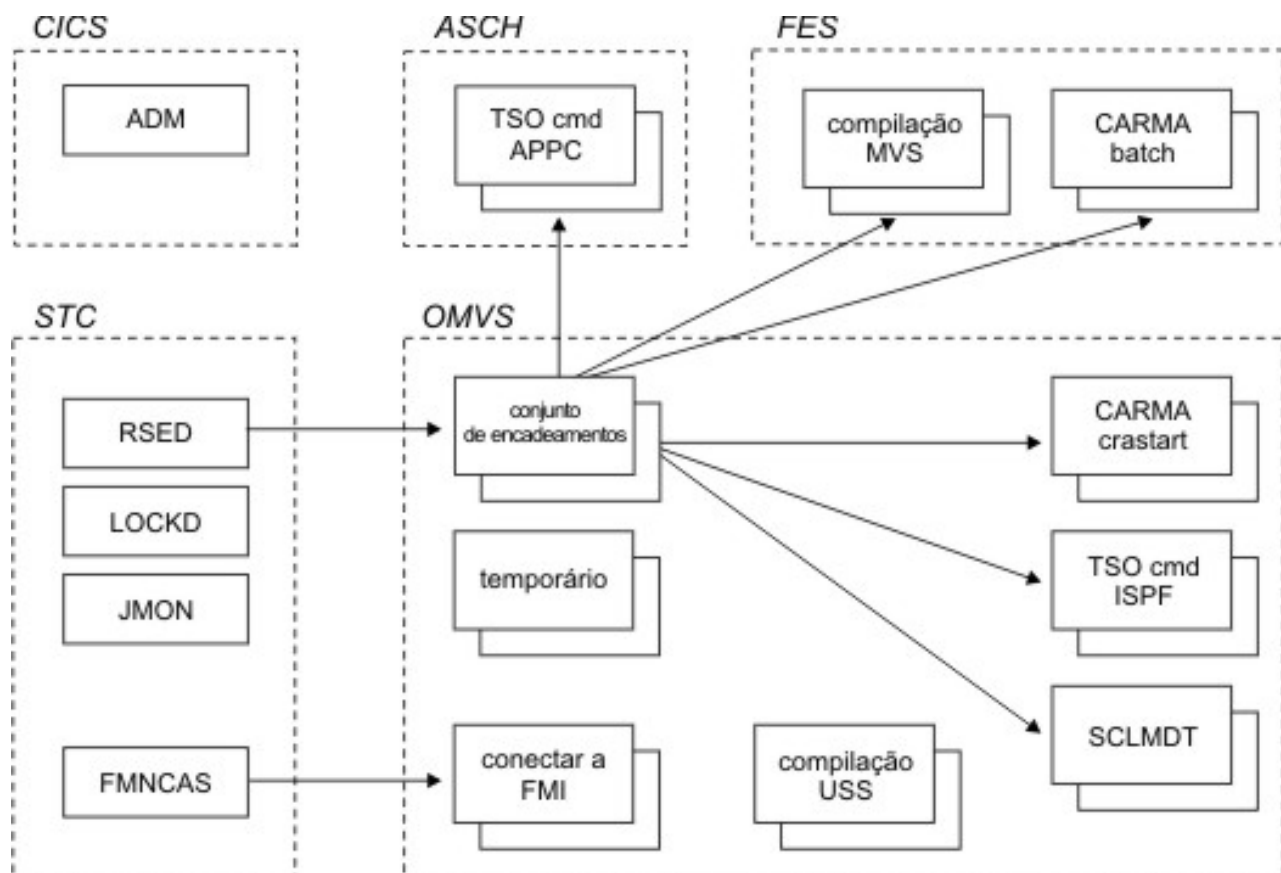


Figura 47. classificação WLM

O Figura 47 mostra uma visão geral básica dos subsistemas por meio dos quais cargas de trabalho são apresentadas para WLMDeveloper para System z.

O Application Deployment Manager (ADM) é ativo dentro de uma região CICS e, por essa razão, seguirá as regras de classificação CICS no WLM.

O daemon RSE (RSED), daemon de bloqueio (LOCKD) e o JES Job Monitor (JMON) são Developer para System z tarefas iniciadas (ou tarefas em lote de longa execução), cada uma com seu espaço de endereço individual.

Como documentado em “RSE como um Aplicativo Java” na página 191, o daemon RSE gera um processo-filho para cada servidor de conjunto de encadeamentos RSE (que suporta um número variável de clientes). Cada conjunto de encadeamentos é ativo em um espaço de endereço separado (usando um inicializador z/OS UNIX, BPXAS). Porque estes são processos gerados, eles são classificados usando as regras de classificação WLM OMVS, e não as regras de classificação de tarefa iniciada.

Os clientes que estão ativos em um conjunto de encadeamentos podem criar uma infinidade de outros espaços de endereços, dependendo das ações realizadas pelos usuários. Dependendo da configuração de Developer para System z algumas cargas de trabalho, como o o serviço TSO Commands (TSO cmd) ou CARMA, podem executar em subsistemas diferentes.

Os espaços de endereços listados em Figura 47 na página 201 permanecem no sistema tempo suficiente para serem visíveis, mas é recomendável estar ciente que, devido a maneira como o z/OS UNIX é designado, há também vários espaços de endereços temporários de curta duração. Estes espaços de endereços temporários estão ativos nos subsistemas OMVS.

Note que enquanto os conjuntos de encadeamento RSE usam o mesmo ID do usuário e um nome de tarefa similar como o daemon RSE, todos os espaços de endereços iniciados por um conjunto de encadeamento são de propriedade do ID do usuário do cliente solicitando a ação. O ID de usuário cliente também é usado como (parte do) nome de tarefa para todos os espaços de endereços baseados em OMVS, estabelecidos pelo conjunto de encadeamentos.

Mais espaços de endereços são criados por outros serviços que o Developer para System z usa, como o File Manager (FMNCAS) ou z/OS UNIX REXEC (USS build).

## Regras de Classificação

O WLM usa regras de classificações para mapear o trabalho entrando no sistema para uma classe de serviço. Esta classificação é baseada nos qualificadores de trabalho. O primeiro qualificador (obrigatório) é um tipo de subsistema que recebe o pedido de trabalho. O Tabela 29 lista os tipos de subsistema que podem receber cargas de trabalho Developer para System z.

*Tabela 29. subsistemas de ponto de entrada WLM*

Tipos de subsistemas	Descrição do trabalho
ASCH	Os pedidos de trabalho incluem todos os programas de transação APPC planejados pelo planejador de transação da IBM-supplied APPC/MVS, ASCH.
CICS	Os pedidos de trabalho incluem todas as transações processadas pelo CICS.
JES	Os pedidos de trabalho incluem todos os trabalhos que o JES2 ou JES3 iniciam.
OMVS	Os pedidos de trabalho incluem o trabalho processado nos espaços de endereços filhos bifurcados no z/OS UNIX System Services.

Tabela 29. subsistemas de ponto de entrada WLM (continuação)

Tipos de subsistemas	Descrição do trabalho
STC	Os pedidos de trabalho incluem todo o trabalho iniciado pelos comandos START e MOUNT. O STC também inclui os espaços de endereços do componente do sistema.

Tabela 30Tabela 2 lista qualificadores adicionais que podem ser usados para designar uma carga de trabalho para uma classe de serviço específica. Consulte MVS Planejamento: Gerenciamento de Carga de Trabalho (SA22-7602) para obter detalhes adicionais sobre os qualificadores de trabalho listados.

Tabela 30. qualificadores de trabalho WLM

		ASCH	CICS	JES	OMVS	STC
AI	Dados da Conta	x		x	x	x
LU	Nome LU (*)		x			
PF	Perform (*)			x		x
PRI	Prioridade			x		
SE	Nome de Ambiente de Planejamento			x		
SSC	Nome de Coleta de Subsistema			x		
SI	Instância de Subsistema (*)		x	x		
SPM	Parâmetro de Subsistema					x
PX	Nome Sysplex	x	x	x	x	x
SY	Nome do Sistema (*)	x			x	x
TC	Transação/Classe de Trabalho (*)	x		x		
TN	Transação/Nome do Trabalho (*)	x	x	x	x	x
UI	ID do usuário (*)	x	x	x	x	x

**Nota:** Para os qualificadores marcados com (\*), é possível especificar os grupos de classificação ao incluir um G na abreviação do tipo. Por exemplo, um grupo de nome de transação seria TNG.

## Configurando Objetivos

Como documentado em “Classificação de Carga de Trabalho” na página 201, o Developer para System z cria tipos diferentes de cargas de trabalho no seu sistema. Estas diferentes tarefas comunicam-se entre si, o que implica que o tempo decorrido real tornar-se importante para evitar problemas de tempo limite para as conexões entre as tarefas. Como resultado, Developer para System z é recomendável que as tarefas sejam colocadas em classes de serviço de alto desempenho ou em classes de serviço de desempenho moderado com uma alta prioridade.

Uma revisão, e possivelmente uma atualização, dos seus objetivos WLM atuais é, por essa razão, aconselhado. Isto é especialmente verdadeiro para empresas tradicionais MVS novas no que diz respeito às cargas de trabalho OMVS de tempo crítico.

#### Nota:

- As informações sobre os objetivos nesta seção são deliberadamente mantidas em um nível descritivo porque os objetivos de desempenho reais são muito específicos do site.
- Para ajudar no entendimento do impacto de uma tarefa específica no seu sistema, termos como recursos substanciais, moderados e mínimos são usados. Todos eles são relativos ao uso total de recurso Developer para System z, ele mesmo, mas não de todo o sistema.

O Tabela 31 lista os espaços de endereços que são usados pelo Developer para System z. O z/OS UNIX substituirá "x" na coluna "Nome da Tarefa" por um número aleatório de 1 dígito.

*Tabela 31. cargas de trabalho WLM*

Descrição	Nome da tarefa	Carga de trabalho
JES Job Monitor	JMON	STC
Daemon de bloqueio	LOCKD	STC
Daemon RSE	RSED	STC
Conjunto de encadeamento do RSE	RSEDx	OMVS
ISPF Client Gateway (Serviço do TSO Commands e do SCLMDT)	<userid>x	OMVS
Serviço do TSO Commands (APPC)	FEKFRSRV	ASCH
CARMA (batch)	CRA<port>	JES
CARMA (crastart)	<userid>x	OMVS
CARMA (ISPF Client Gateway)	<userid> e <userid>x	OMVS
Build MVS (tarefa em lote)	*	JES
Construção do z/OS UNIX build (comandos do shell)	<userid>x	OMVS
Shell z/OS UNIX	<userid>	OMVS
Tarefa do File Manager	<userid>x	OMVS
Application Deployment Manager	CICSTS	CICS

## Considerações para Seleção de Objetivos

As seguintes considerações gerais do WLM podem ajudar a definir apropriadamente as definições de objetivos corretas para Developer para System z:

- É recomendável que você baseie os objetivos no que é realmente possível ser atingido e não no que você deseja que aconteça. Se estabelecer objetivos mais altos do que o necessário, o WLM moverá recursos de trabalho de importância menor para trabalho de importância mais alta, que pode não necessariamente precisar de recursos.
- Limite a quantia de trabalho designada para classes de serviços SYSTEM e SYSSTC porque essas classes possuem uma prioridade de despacho mais alta que qualquer classe WLM gerenciada. Use estas classes para trabalho que seja de importância alta, mas que use pouco CPU.
- O trabalho que é desaprovado pelas regras de classificação acaba na classe SYSOTHER, que possui um objetivo discricionário. Um objetivo discricionário diz ao WLM para apenas fazer o seu melhor, quando o sistema possuir recursos sobressalentes.

Quando usar os objetivos de tempo de resposta:

- Deve haver uma taxa de chegada de tarefas fixa (pelo menos 10 tarefas em 20 minutos) para que o WLM gerencie apropriadamente um objetivo de tempo de resposta.
- Use objetivos de tempo médio de resposta somente para cargas de trabalho bem controladas, porque uma única transação longa possui um grande impacto sobre o tempo médio de resposta e pode fazer com que o WLM reaja exageradamente.

Quando usar objetivos de velocidade:

- Normalmente não é possível atingir um objetivo de velocidade acima de 90% por várias razões. Por exemplo, todos os espaços de endereços SYSTEM e SYSSTC possuem uma prioridade de despacho mais alto que o objetivo de tipo de velocidade.
- O WLM usa um número mínimo de (uso e atraso) amostras nas quais baseia suas decisões sobre objetivo de velocidade. Assim quando menor for a execução de trabalho em uma classe de serviço, maior será o tempo para coletar o número requerido de amostras e ajustar a política de despacho.
- Reavalie os objetivos de velocidade quando alterar o seu hardware. Em particular, mover para processadores mais rápidos e menores requer mudanças nos objetivos de velocidade.

## STC

Todas as tarefas iniciadas, daemon RSE, daemon de Bloqueio e JES Job Monitor do Developer para System z estão atendendo pedidos de clientes em tempo real.

*Tabela 32. cargas de trabalho WLM STC*

Descrição	Nome da tarefa	Carga de trabalho
JES Job Monitor	JMON	STC
Daemon de bloqueio	LOCKD	STC
Daemon RSE	RSED	STC

- JES Job Monitor

O JES Job Monitor fornece todos os serviços JES relacionados, como submeter tarefas, navegar em arquivos de spool e executar comandos do operador JES. É recomendável especificar um objetivo de velocidade de um período, e com alta prioridade, porque a tarefa não relata transações individuais para o WLM. O uso de recurso depende fortemente das ações do usuário e, por essa razão, vai variar, mas é esperado que seja de mínimo a moderado.

- Daemon de bloqueio

O daemon de bloqueio consulta as tabelas de enfileirar do GRS sobre o pedido do cliente e operador e compara o resultado com os usuários conhecidos Developer para System z. É recomendável que você especifique um objetivo de velocidade de um período e com prioridade alta porque a tarefa não relata transações individuais no WLM. É esperado ser mínimo o uso de recurso.

- Daemon RSE

O daemon RSE trata da criação do logon e autenticação do cliente e gerencia os diferentes conjuntos de encadeamentos RSE. É recomendável que você especifique um objetivo de velocidade de um período e com prioridade alta porque a tarefa não relata transações individuais no WLM. Espera-se ser mínimo o uso de recurso, com um pico no começo do dia útil.



## OMVS

As cargas de trabalho OMVS podem ser divididas em dois grupos, conjuntos de encadeamentos RSE e todo o resto. Isto porque todas as cargas de trabalho, exceto os conjuntos de encadeamentos RSE, usam o ID de usuário cliente como base para o nome do espaço de endereço. (O z/OS UNIX substituirá o "x" na coluna "Nome da Tarefa" por um número aleatório de 1 dígito).

*Tabela 33. Cargas de trabalho WLM OMVS*

Descrição	Nome da tarefa	Carga de trabalho
Conjunto de encadeamento do RSE	RSEDx	OMVS
ISPF Client Gateway (Serviço do TSO Commands e do SCLMDT)	<userid>x	OMVS
CARMA (crastart)	<userid>x	OMVS
CARMA (ISPF Client Gateway)	<userid> e <userid>x	OMVS
Construção do z/OS UNIX build (comandos do shell)	<userid>x	OMVS
Shell z/OS UNIX	<userid>	OMVS
Tarefa do File Manager	<userid>x	OMVS

- Conjunto de encadeamento do RSE

Um conjunto de encadeamentos RSE é como o coração e o cérebro do Developer para System z. Quase todos os dados circulam por aqui e os mineradores (encadeamentos específicos do usuário) dentro do conjunto de encadeamentos controlam as ações da maioria das outras tarefas relacionadas do Developer para System z. É recomendável que você especifique um objetivo de velocidade de um período e com prioridade alta porque a tarefa não relata transações individuais no WLM. O uso de recursos depende fortemente das ações do usuário e, por essa razão, vai variar, mas é esperado ser substancial.

Todas as cargas de trabalho restantes terminarão na mesma classe de serviço devido a uma convenção de nomenclatura do espaço de endereço comum. É recomendável especificar um objetivo de período múltiplo para esta classe de serviço. Os primeiros períodos deveriam ser de objetivos de tempo de resposta percentil e de alto desempenho, enquanto que o último período deveria possuir um objetivo de velocidade de desempenho moderado. Algumas cargas de trabalho, como a ISPF Client Gateway, relatarão transações individuais para o WLM, enquanto outras não.

- ISPF Client Gateway

O ISPF Client Gateway é um serviço ISPF invocado pelo Developer para System z para executar comandos TSO e ISPF não interativos. Isto inclui comandos explícitos emitidos pelo cliente e também os comandos implícitos emitidos pelo Developer para System z, como obter uma lista de membros PDS. O uso de recursos depende fortemente das ações do usuário, e, por essa razão, vai variar, mas é esperado ser o mínimo.

- CARMA

O CARMA é um servidor Developer para System z opcional usado para interagir com Software Configuration Managers (SCMs), baseados em host, tais como o CA Endevor® SCM. O Developer para System z permite diferentes métodos de inicialização para um servidor CARMA, alguns dos quais



transformam-se em uma carga de trabalho OMVS. O uso de recursos depende fortemente das ações do usuário, e, por essa razão, vai variar, mas é esperado ser o mínimo.

- z/OS UNIX build

Quando um cliente inicia uma construção para um projeto z/OS UNIX, o z/OS UNIX REXEC (ou SSH) iniciará uma tarefa que executa um número de comandos do shell z/OS UNIX para executar a construção. O uso dos recursos depende fortemente das ações do usuário e, por essa razão, vai variar, mas é esperado ser de moderado a substancial, dependendo do tamanho do projeto.

- Shell z/OS UNIX

Esta carga de trabalho processa os comandos do shell z/OS UNIX que são emitidos pelo cliente. O uso dos recursos depende fortemente das ações do usuário, e, por essa razão, vai variar, mas é esperado ser o mínimo.

- IBM File Manager

Embora não sejam espaços de endereços Developer para System z, os processos-filho do File Manager gerados são listados aqui porque eles podem ser iniciados por pedido de um cliente Developer para System z e estas tarefas usam a mesma convenção de nomenclatura como tarefas Developer para System z. Estas tarefas do File Manager processam ações do conjunto de dados não triviais do MVS, como edição formatada de um arquivo VSAM. O uso de recursos depende grandemente das ações do usuário e, por essa razão, vai variar, mas é esperado ser de mínimo a moderado.

## JES

Os processos em lote do JES gerenciado são usados de várias maneiras pelo Developer para System z. O uso mais comum é para construções MVS, onde uma tarefa é submetida e monitorada para determinar quando ela termina. Mas o Developer para System z também poderia iniciar um servidor CARMA em lote e comunicar-se com ele usando TCP/IP.

*Tabela 34. Carga de trabalhos WLM JES*

Descrição	Nome da tarefa	Carga de trabalho
CARMA (batch)	CRA<port>	JES
Build MVS (tarefa em lote)	*	JES

- CARMA

O CARMA é um servidor Developer para System z opcional usado para interagir com Software Configuration Managers (SCMs), baseados em host, tais como o CA Endevor<sup>®</sup> SCM. O Developer para System z permite diferentes métodos de inicialização para um servidor CARMA, alguns dos quais transformam-se em uma carga de trabalho JES. É recomendável especificar um objetivo de velocidade de um período, e com alta prioridade, porque a tarefa não relata transações individuais para o WLM. O uso de recursos depende fortemente das ações do usuário, e, por essa razão, vai variar, mas é esperado ser o mínimo.

- Build MVS

Quando o cliente inicia uma construção para um projeto MVS, Developer para System z iniciará um trabalho em lote para executar a construção. O uso dos recursos depende fortemente das ações do usuário e, por essa razão, vai variar, mas é esperado ser de moderado a substancial, dependendo do tamanho do projeto. Estratégias diferentes de objetivos de desempenho moderado é aconselhável, dependendo das circunstâncias locais.

- Você poderia especificar um objetivo de período múltiplo com um período de tempo de resposta percentil e um período de velocidade de final. Neste caso, seus desenvolvedores deveriam estar usando principalmente o mesmo procedimento de construção e arquivos de entrada de tamanho similar para criar tarefas com tempos de resposta uniformes. Também deve haver uma taxa de chegada de tarefas fixa (pelo menos 10 tarefas em 20 minutos) para que o WLM gerencie apropriadamente o objetivo de tempo de resposta.
- Um objetivo de velocidade é mais adequado para a maioria de tarefas em lote porque este objetivo pode lidar com taxas de chegada e tempos de execução altamente variáveis.

## ASCH

Nas versões atuais do Developer para System z, o ISPF Client Gateway é usado para executar comandos TSO e ISPF não interativos. Devido a razões históricas, o Developer para System z também suporta executar estes comandos por meio de uma transação APPC.

*Tabela 35. Cargas de trabalho WLM ASCH*

Descrição	Nome da tarefa	Carga de trabalho
Serviço do TSO Commands (APPC)	FEKFRSRV	ASCH

- serviço TSO Commands  
O serviço TSO Commands pode ser iniciado como uma transação APPC pelo Developer para System z para executar os comandos TSO e ISPF não interativos. Isto inclui os comandos explícitos emitidos pelo cliente e também os comandos implícitos emitidos pelo Developer para System z, como obter uma lista de membros PDS. É recomendável especificar um objetivo de período múltiplo para esta classe de serviço. Para os primeiros períodos, é recomendável especificar objetivos de tempo de resposta percentil e de alto desempenho. Para o período final, é recomendável especificar um objetivo de velocidade de desempenho moderada. O uso dos recursos depende fortemente das ações do usuário, e, por essa razão, vai variar, mas é esperado ser o mínimo.

## CICS

O Application Deployment Manager é um servidor do Developer para System z opcional ativo dentro de uma região do CICS Transaction Server.

*Tabela 36. Cargas de trabalho WLM CICS*

Descrição	Nome da tarefa	Carga de trabalho
Application Deployment Manager	CICSTS	CICS

- Application Deployment Manager  
O servidor opcional do Application Deployment Manager, que é ativo dentro de uma região CICSTS, permite que você transfira com segurança tarefas de gerenciamento CICSTS selecionadas para desenvolvedores. O uso dos recursos depende fortemente das ações do usuário, e, por essa razão, vai variar, mas é esperado ser o mínimo. O tipo de classe de serviço que você deveria usar depende das outras transações ativas nesta região CICS e, por essa razão, isso não é discutido em detalhe.

O WLM suporta múltiplos tipos de gerenciamento que podem ser usado para CICS:

- Gerenciar CICS através de um objetivo de região

O objetivo é configurado para uma classe de serviço que gerencia os espaços de endereço CICS. Só é possível usar um objetivo de velocidade de execução para esta classe de serviço. O WLM usa as regras de classificação JES ou STC para os espaços de endereços, mas não usa as regras de classificação do subsistema CICS para transações.

- Gerenciar CICS através de um objetivo de tempo de resposta de transação

O objetivo de tempo de resposta pode ser configurado em uma classe de serviço designada para uma única transação ou um grupo de transações. O WLM usa as regras de classificações JES ou STC para os espaços de endereços e as regras de classificação de subsistema CICS para transações.



---

## Capítulo 13. Considerações de Ajuste

Conforme explicado no Capítulo 11, “Entendimento do Developer para System z”, na página 189, o RSE (Remote Systems Explorer) é o núcleo do Developer para System z. Para gerenciar as conexões e as cargas de trabalho dos clientes, o RSE é formado por um espaço de endereço do daemon, que controla os espaços de endereços do conjunto de encadeamento. O daemon atua como um ponto focal para fins de conexão e gerenciamento, enquanto os conjuntos de encadeamento processam as cargas de trabalho do cliente.

Isso torna o RSE um alvo principal para o ajuste da configuração do Developer para System z. Entretanto, a manutenção de centenas de usuários, cada um usando 16 ou mais encadeamentos, uma determinada quantidade de armazenamento e possivelmente 1 ou mais espaços de endereço exigem configuração adequada do Developer para System z e do z/OS.

Os seguintes tópicos são abordados neste capítulo:

- “Uso de Recursos”
- “Uso de Armazenamento” na página 222
- “Uso do Espaço do Sistema de Arquivos z/OS UNIX” na página 228
- “Definições de Recursos Principais” na página 231
- “Definições de Vários Recursos” na página 234
- “Monitoramento” na página 236
- “Configuração de Amostra” na página 239

---

### Uso de Recursos

Use as informações desta seção para avaliar o uso normal e máximo de recursos pelo Developer para System z, para que você possa planejar a configuração do sistema de acordo.

Ao usar os números e as fórmula apresentados nesta seção para definir os valores dos limites do sistema, lembre-se de que você está trabalhando com estimativas bastante precisas. Deixe margem suficiente ao configurar os limites do sistema para permitir o uso de recursos por tarefas temporárias e outras tarefas, ou por usuários que se conectam várias vezes ao host simultaneamente. (Por exemplo, por meio de RSE e de TN3270).

#### Nota:

- As informações são limitadas no escopo para serviços acessados por meio do RSE que são fornecidos pelo próprio Developer para System z. Por exemplo, o uso de recursos do TN3270 não é documentado (não acessado por meio do RSE), nem o uso de recursos dos programas chamados durante as construções remotas (baseadas em host) de projetos do MVS ou do z/OS UNIX (não fornecidos pelo Developer para System z).
- Adicionar extensões de terceiros ao Developer para System z pode aumentar os contadores de uso de recursos.

- Todos os serviços possuem tarefas de "manutenção" de curta duração, que usam recursos durante sua execução e que podem executar em sequência ou em paralelo uma em relação a outra. Os recursos usados por essas tarefas não são documentados.
- Onde for útil, o uso de recursos específicos do usuário de software obrigatório, como o ISPF Client Gateway, é documentado.
- Os números apresentados aqui podem ser alterados sem notificação prévia.

## Visão Geral

As tabelas a seguir oferecem uma visão geral do número de espaços de endereços, processos e encadeamentos usados pelo Developer para System z. Mais detalhes sobre os números apresentados aqui podem ser encontrados nas próximas seções:

- "Contagem do Espaço de Endereço" na página 213
- "Contagem de Processos" na página 216
- "Contagem de Encadeamentos" na página 219

A Tabela 37 fornece uma visão geral dos recursos principais usados pelas tarefas iniciadas do Developer para System z. Esses recursos são alocados apenas uma vez. Eles são compartilhados entre todos os clientes do Developer para System z.

*Tabela 37. Uso de recursos comuns*

Tarefa iniciada	Espaços de endereço	Processos	Encadeamentos
JMON	1	1	3
LOCKD	1	3	10
RSED	1	3	11
RSEDx	(a)	2	10

**Nota:** (a) Existe pelo menos 1 espaço de endereço do conjunto de encadeamento do RSE ativo. Consulte "Contagem do Espaço de Endereço" na página 213 para determinar o número real de espaços de endereços do conjunto de encadeamento do RSE.

A Tabela 38 fornece uma visão geral dos recursos principais usados pelo software obrigatório. Esses recursos são alocados para cada cliente do Developer para System z que invoca a função relacionada.

*Tabela 38. Uso de recursos obrigatórios específicos do usuário*

Software obrigatório	Espaços de endereço	Processos	Encadeamentos
ISPF Client Gateway	1	2	4
APPC	1	1	2
File Manager	1	1	2

A Tabela 39 na página 213 fornece uma visão geral dos recursos principais usados por cada cliente do Developer para System z ao executar a função especificada. Valores não-numéricos, como ISPF, são uma referência ao valor correspondente na Tabela 38.

Tabela 39. Uso de recursos específicos do usuário

Ação do usuário	Espaços de endereço	Processos	Encadeamentos		
	ID do usuário	ID do usuário	ID do usuário	RSEDx	JMON
Logon	-	-	-	16	1
Cronômetro para tempo limite inativo	-	-	-	1	-
Expandir PDS(E)	ISPF	ISPF	ISPF	-	-
Abrir conjunto de dados	ISPF	ISPF	ISPF	-	-
Comando TSO	ISPF	ISPF	ISPF	-	-
Shell do z/OS UNIX	1	1	1	6	-
Build MVS	1	-	-	-	-
Build z/OS UNIX	3	3	3	-	-
CARMA (batch)	1	1	2	1	-
CARMA (crastart)	1	1	2	4	-
CARMA (ispf)	4	4	7	5	-
SCLMDT	ISPF	ISPF	ISPF	-	-
Integração do File Manager	ISPF + FM	ISPF + FM	ISPF + FM	-	-
Integração do Analisador de Falha	-	-	-	-	-

**Nota:** O ISPF pode ser substituído pelo APPC, exceto para o SCLM Developer Toolkit.

## Contagem do Espaço de Endereço

A Tabela 40 lista os espaços de endereço que são usados pelo Developer para System z, em que “u” na coluna “Contagem” indica que a quantidade deve ser multiplicada pelo número de usuários ativos simultaneamente que usam a função. O z/OS UNIX substituirá “x” na coluna “Nome da Tarefa” por um número de 1 dígito aleatório.

Tabela 40. Contagem do espaço de endereço

Contador	Descrição	Nome da tarefa	Compar tilhado	Termina após
1	JES Job Monitor	JMON	SIM	Nunca
1	Daemon de bloqueio	LOCKD	SIM	Nunca

Tabela 40. Contagem do espaço de endereço (continuação)

Contador	Descrição	Nome da tarefa	Compar tilhado	Termina após
1	Daemon RSE	RSED	SIM	Nunca
(a)	Conjunto de encadeamento do RSE	RSEDx	SIM	Nunca
1u	ISPF Client Gateway (Serviço do TSO Commands e do SCLMDT)	<userid>x	Não	15 minutos ou no logoff do usuário
1u	Serviço do TSO Commands (APPC)	FEKFRSRV	Não	60 minutos ou efetuar logoff do usuário
1u	CARMA (batch)	CRA<port>	Não	7 minutos ou no logoff do usuário
1u	CARMA (crastart)	<userid>x	Não	7 minutos ou no logoff do usuário
4u	CARMA (ispf)	(1)<userid> ou (3)<userid>x	Não	7 minutos ou no logoff do usuário
(b)	Uso simultâneo do ISPF Client Gateway por 1 usuário	<userid>x	Não	Conclusão da tarefa
1u	Build MVS (tarefa em lote)	*	Não	Conclusão da tarefa
3u	Build z/OS UNIX (comandos do shell)	<userid>x	Não	Conclusão da tarefa
1u	Shell do z/OS UNIX	<userid>	Não	Logoff do usuário
(c)	File Manager	<userid>x	Não	Conclusão da tarefa

**Nota:**

- (a) Existe pelo menos um espaço de endereço do conjunto de encadeamento do RSE ativo. O número real depende de:
  - A diretiva `minimum.threadpool.process` em `rsed.envvars`. O valor padrão é 1.
  - O número de usuários que podem ser atendidos por um conjunto de encadeamento. As configurações padrão são planejadas para 60 usuários por conjunto de encadeamento.
  - A marca do limite máximo de usuários ativos simultaneamente, pois os conjuntos de encadeamento inativos não são interrompidos automaticamente.
- (b) O Developer para System z possui vários encadeamentos ativos por usuário. No caso de o espaço de endereço do ISPF Client Gateway não tiver terminado de atender aos pedido de um encadeamento quando outro encadeamento enviar um pedido, o ISPF inicializará um novo Client Gateway para processar o novo pedido. Esse espaço de endereço termina após a conclusão da tarefa.
- (c) O listener do File Manager inicia um espaço de endereço por objeto que deve ser manipulado, por exemplo, um VSAM. Esse espaço de endereço permanece ativo até que o Developer para System z sinalize que o objeto não é mais necessário, por exemplo, fechando o VSAM.
- O SCLMDT exige um espaço de endereço do ISPF Client Gateway. O SCLMDT compartilha o espaço de endereço com o serviço do TSO Commands.



- A maioria das ações relacionadas ao conjunto de dados do MVS usa o serviço do TSO Commands, que pode estar ativo no ISPF Client Gateway ou em uma transação do APPC, respectivamente.

Use a fórmula da Figura 48 para avaliar o número máximo de espaços de endereço usado pelo Developer para System z.

$$3 + A + N*(x + y + z) + (2 + N*0.01)$$

Figura 48. Número máximo de espaços de endereço

Em que,

- “3” é igual ao número de espaços de endereço do servidor ativo permanente.
- “A” representa o número de espaços de endereço do conjunto de encadeamento do RSE.
- “N” representa o número máximo de usuários simultâneos.
- “x” é um dos seguintes valores, dependendo das opções de configuração selecionadas.

X	SCLMDT	TSO por meio do Client Gateway	TSO por meio do APPC
1	Não	Não	SIM
1	Não	SIM	Não
1	SIM	SIM	Não

- “y” é um dos seguintes valores, dependendo das opções de configuração selecionadas.

S	
0	Sem CARMA
1	CARMA (batch)
1	CARMA (crastart)
4	CARMA (ispf)

- “z” é 0 por padrão, mas pode aumentar dependendo das ações do usuário:
  - Inclua 1 quando um build MVS for executado. Esses espaços de endereço terminam quando a tarefa do build relacionado (uma tarefa em lote) for concluída.
  - Inclua 3 quando um build z/OS UNIX for executado. Observe que o número real pode ser alto, dependendo das necessidades dos programas invocados. Esses espaços de endereço terminam quando a tarefa do build relacionada for concluída.
  - Inclua 1 para cada interação simultânea com o IBM File Manager. Esses espaços de endereço terminam quando o objeto solicitado não é mais necessário.
- “2 + N\*0.01” inclui um buffer para espaços de endereço temporários. O tamanho do buffer necessário pode ser diferente no seu site.

Use a fórmula da Figura 49 na página 216 para avaliar o número máximo de espaços de endereço usados por um cliente do Developer para System z (sem contar os espaços de endereço temporários não documentados).

$$x + y + z$$

Figura 49. Número de espaços de endereços por cliente

Em que,

- "x" depende das opções de configuração selecionadas e é documentado na fórmula para calcular o número máximo de espaços de endereço (Figura 48 na página 215).
- "y" depende das opções de configuração selecionadas e é documentado na fórmula para calcular o número máximo de espaços de endereço (Figura 48 na página 215).
- "z" é 0 por padrão, mas pode aumentar, dependendo das ações do usuário, conforme documentado na fórmula para calcular o número máximo de espaços de endereço (Figura 48 na página 215).

As definições na Tabela 41 podem limitar o número real de espaços de endereço.

Tabela 41. Limites de espaço de endereço

Local	Limite	Recursos afetados
rsed.envvars	maximum.threadpool.process	Limita o número de conjuntos de encadeamento do RSE
IEASYMxx	MAXUSER	Limita o número de espaços de endereço
ASCHPMxx	MAX	Limita o número de iniciadores APPC para o serviço TSO Commands (APPC)

## Contagem de Processos

A Tabela 42 lista o número de processos por espaço de endereço que é usado pelo Developer para System z. "u" nos "Espaços de Endereço" indica que a quantidade deve ser multiplicada pelo número de usuários ativos simultaneamente usando a função.

Tabela 42. Contagem de processos

Processos	Espaços de endereço	Descrição	ID do usuário
1	1	JES Job Monitor	STCJMON
3	1	Daemon de bloqueio	STCLOCK
3	1	Daemon RSE	STCRSE
2	(a)	Conjunto de encadeamento do RSE	STCRSE
2	(b)	ISPF Client Gateway (Serviço do TSO Commands e do SCLMDT)	<userid>
1	1u	Serviço do TSO Commands (APPC)	<userid>
1	1u	CARMA (batch)	<userid>
1	1u	CARMA (crastart)	<userid>
1	1u	CARMA (ispf)	<userid>
1	3u	Build z/OS UNIX (comandos do shell)	<userid>
1	1u	Shell do z/OS UNIX	<userid>
1	(c)	File Manager	<userid>
(5)	(u)	SCLM Developer Toolkit	<userid>

**Nota:**

- (a) Existe pelo menos 1 espaço de endereço do conjunto de encadeamento do RSE ativo. Consulte “Contagem do Espaço de Endereço” na página 213 para determinar o número real de espaços de endereços do conjunto de encadeamento do RSE.
- O daemon RSE e todos os conjuntos de encadeamento do RSE usam o mesmo ID de usuário.
- (b) Em situações normais e ao usar as opções de configuração padrão, existe 1 ISPF Client Gateway ativo por usuário. O número real pode variar, conforme descrito em “Contagem do Espaço de Endereço” na página 213.
- (c) O listener do File Manager usa um processo por objeto que deve ser manipulado, por exemplo, um VSAM. Esse processo permanece ativo até que o Developer para System z sinalize que o objeto não é mais necessário, por exemplo, fechando o VSAM.
- O SCLMDT exige um espaço de endereço do ISPF Client Gateway. O SCLMDT compartilha o espaço de endereço com o serviço do TSO Commands.
- (u) Os processos SCLMDT são executados no espaço de endereço do ISPF Client Gateway e, portanto, não possuem um valor para a contagem do espaço de endereço.
- Os processos SCLMDT são temporários e terminam na conclusão da tarefa, mas vários processos podem estar ativos simultaneamente para um único usuário. A Tabela 42 na página 216 lista o número máximo de processos SCLMDT simultâneos.
- A maioria das ações relacionadas ao conjunto de dados do MVS usa o serviço do TSO Commands, que pode estar ativo no ISPF Client Gateway ou em uma transação do APPC, respectivamente.
- Um build z/OS UNIX usa três processos no total, cada um executando em seu próprio espaço de endereço.
- Todos os processos listados permanecem ativos até que o espaço de endereço relacionado termine, a menos que indicado de outra forma.

Use a fórmula da Figura 50 para avaliar o número máximo de processos usados pelo Developer para System z.

$$7 + 2 * A + N * (x + y + z) + (10 + N * 0.05)$$

*Figura 50. Número máximo de processos*

Em que,

- “7” é igual ao número de processos usados por espaços de endereço do servidor ativo permanente.
- “A” representa o número de espaços de endereço do conjunto de encadeamento do RSE.
- “N” representa o número máximo de usuários simultâneos.
- “x” é um dos seguintes valores, dependendo das opções de configuração selecionadas.

X	SCLMDT	TSO por meio do Client Gateway	TSO por meio do APPC
1	Não	Não	SIM
2	Não	SIM	Não
7	SIM	SIM	Não

- “y” é um dos seguintes valores, dependendo das opções de configuração selecionadas.

S	
0	Sem CARMA
1	CARMA (batch)
1	CARMA (crastart)
4	CARMA (ispf)

- “z” é 0 por padrão, mas pode aumentar dependendo das ações do usuário:
  - Inclua 1 quando um shell z/OS UNIX for aberto. Esse processo permanece ativo até que o usuário efetue logoff.
  - Inclua 3 quando um build z/OS UNIX for executado. Observe que o número real pode ser alto, dependendo das necessidades dos programas invocados. Esses processos terminam quando a tarefa do build relacionada for concluída.
  - Inclua 1 para cada interação simultânea com o IBM File Manager. Esses processos terminam quando o objeto solicitado não é mais necessário.
- “10 + N\*0.05” inclui um buffer para processos temporários. O tamanho do buffer necessário pode ser diferente no seu site.

Use a fórmula da Figura 51 para avaliar o número máximo de processos usados por um cliente do Developer para System z (sem contar os processos temporários não documentados).

$$(x + y + z) + 5*s$$

Figura 51. Número de processos por cliente

Em que,

- “x” depende das opções de configuração selecionadas e é documentado na fórmula para calcular o número máximo de processos (Figura 50 na página 217).
- “y” depende das opções de configuração selecionadas e é documentado na fórmula para calcular o número máximo de processos (Figura 50 na página 217).
- “z” é 0 por padrão, mas pode aumentar dependendo das ações do usuário, conforme documentado na fórmula para calcular o número máximo de processos (Figura 50 na página 217).
- “s” é 1 quando o SCLM Developer Toolkit for usado ou, senão, 0.

As definições na Tabela 43 podem limitar o número real de processos.

Tabela 43. Limites do processo

Local	Limite	Recursos afetados
BPXPRMxx	MAXPROCSYS	Limita o número total de processos

Tabela 43. Limites do processo (continuação)

Local	Limite	Recursos afetados
BPXPRMxx	MAXPROCUSER	Limita o número de processos por UID do z/OS UNIX

Nota:

- O daemon RSE e os conjuntos de encadeamento do RSE usam o mesmo ID do usuário. Como o daemon RSE começa um novo conjunto de encadeamento sempre que for necessário, o número de processos para este ID do usuário pode aumentar. Portanto, MAXPROCUSER deve ser definido para acomodar esse crescimento, que pode ser formulado como  $3 + 2 \cdot A$ .
- O limite MAXPROCUSER é por ID de usuário (UID) exclusivo z/OS UNIX. Multiplique a contagem do processo por usuário estimada pelo número de clientes ativos simultaneamente se os seus usuários compartilharem o mesmo UID.

## Contagem de Encadeamentos

A Tabela 44 lista o número de encadeamentos usados pelas funções selecionadas do Developer para System z. "u" nas colunas "Encadeamentos" indica que a quantidade deve ser multiplicada pelo número de usuários ativos simultaneamente usando a função. A contagem de encadeamento é listada por processo, à medida que os limites são definidos neste nível.

- RSEDx: Esses encadeamentos são criados no conjunto de encadeamento do RSE, que é compartilhado por vários clientes. Todos os encadeamentos que terminam no mesmo conjunto de encadeamento devem ser incluídos juntos para obter a contagem total.
- Ativo: Esses encadeamentos fazem parte do processo que realmente executa a função solicitada. Cada processo é uma unidade independente, de modo que não há necessidade de somar as contagens de encadeamento, mesmo se elas estiverem designadas ao mesmo ID do usuário, a menos que observado de outra forma.
- Autoinicialização: Os processos de autoinicialização são necessários para iniciar o processo real. Cada um possui 1 encadeamento e pode haver várias autoinicializações consecutivas. Não há necessidade de somas as contagens de encadeamentos.

Tabela 44. Contagem de encadeamentos

Encadeamentos			ID do usuário	Descrição
RSEDx	Ativado	Autoinicialização		
-	3 + 1u	-	STCJMON	JES Job Monitor
-	10	2	STCLOCK	Daemon de bloqueio
-	11	2	STCRSE	Daemon RSE
10 (a) + 16u	-	1 (a)	STCRSE	Conjunto de encadeamento do RSE

Tabela 44. Contagem de encadeamentos (continuação)

Encadeamentos			ID do usuário	Descrição
-	4u (b)	1u (b)	<userid>	ISPF Client Gateway (Serviço do TSO Commands e do SCLMDT)
-	2u	-	<userid>	Serviço do TSO Commands (APPC)
1u	2u	-	STCRSE e <userid>	CARMA (batch)
4u	2u	-	STCRSE e <userid>	CARMA (crastart)
5u	4u	3u	STCRSE e <userid>	CARMA (ispf)
-	1u (d)	2u	<userid>	Build z/OS UNIX (comandos do shell)
6u	1u	-	STCRSE e <userid>	Shell do z/OS UNIX
-	2u (c)	-	<userid>	File Manager
-	(5)	-	<userid>	SCLM Developer Toolkit
1u	-	-	STCRSE	Cronômetro para tempo limite inativo

**Nota:**

- (a) Existe pelo menos 1 espaço de endereço do conjunto de encadeamento do RSE ativo. Consulte “Contagem do Espaço de Endereço” na página 213 para determinar o número real de espaços de endereços do conjunto de encadeamento do RSE.
- (b) Em situações normais e ao usar as opções de configuração padrão, existe 1 ISPF Client Gateway ativo por usuário. O número real pode variar, conforme descrito em “Contagem do Espaço de Endereço” na página 213.
- (c) Existe um processo específico do usuário (com a contagem de encadeamento listada) por interação com o IBM File Manager. Esses processos terminam quando o objeto solicitado não é mais necessário.
- O SCLMDT exige um espaço de endereço do ISPF Client Gateway. O SCLMDT compartilha o espaço de endereço com o serviço do TSO Commands.
- Dependendo da ação selecionada, o SCLMDT pode usar vários processos de encadeamento único que terminam na conclusão da tarefa. A Tabela 44 na página 219 lista o número máximo de encadeamentos SCLMDT simultâneos.
- A maioria das ações relacionadas ao conjunto de dados do MVS usa o serviço do TSO Commands, que pode estar ativo no ISPF Client Gateway ou em uma transação do APPC, respectivamente.
- (d) Um build z/OS UNIX invoca utilitários de build diferentes, que podem ser multiencadeados. A Tabela 44 na página 219 lista o número mínimo de encadeamentos de build z/OS UNIX simultâneos.

- Todos os encadeamentos listados permanecem ativos até que o processo relacionado termine, a menos que indicado de outra forma.

Use a fórmula da Figura 52 para avaliar o número máximo de encadeamentos usado por um conjunto de encadeamento do RSE. Use a fórmula da Figura 53 para avaliar o número máximo de encadeamentos usados pelo JES Job Monitor.

$$9 + N*(16 + x + y + z) + (20 + N*0.1)$$

Figura 52. Número máximo de encadeamentos do conjunto de encadeamento do RSE

$$3 + N$$

Figura 53. Número máximo de encadeamentos do JES Job Monitor

Em que,

- "N" representa o número máximo de usuários simultâneos neste conjunto de encadeamentos ou JES Job Monitor. As configurações padrão são planejadas para 60 usuários por conjunto de encadeamento.
- "x" é um dos valores a seguir, dependendo das opções de configuração selecionadas.

X	SCLMDT	TSO por meio do Client Gateway	TSO por meio do APPC	Tempo limite
0	Não	Não	SIM	Não
0	Não	SIM	Não	Não
0	SIM	SIM	Não	Não
1	Não	Não	SIM	SIM
1	Não	SIM	Não	SIM
1	SIM	SIM	Não	SIM

- "y" é um dos seguintes valores, dependendo das opções de configuração selecionadas.

S	
0	Sem CARMA
1	CARMA (batch)
4	CARMA (crastart)
5	CARMA (ispf)

- "z" é 0 por padrão, mas pode aumentar dependendo das ações do usuário:
  - Inclua 6 quando um shell z/OS UNIX for aberto. Esses encadeamentos permanecem ativos até que o usuário efetue logoff.
- "20 + N\*0.1" inclui um buffer para encadeamentos temporários. O tamanho do buffer necessário pode ser diferente no seu site.

As definições na Tabela 45 podem limitar o número real de encadeamentos em um processo, o que é da maior importância para os conjuntos de encadeamentos do RSE.

*Tabela 45. Limites de encadeamento*

Local	Limite	Recursos afetados
BPXPRMxx	MAXTHREADS	Limita o número de encadeamentos em um processo.
BPXPRMxx	MAXTHREADTASKS	Limita o número de tarefas MVS em um processo.
BPXPRMxx	MAXASSIZE	Limita o tamanho do espaço de endereço e, portanto, o armazenamento disponível para blocos de controle relacionados ao encadeamento.
rsed.envvars	Xmx	Define o tamanho máximo do heap Java. Esse armazenamento é reservado e, portanto, não está mais disponível para blocos de controle relacionados ao encadeamento.
rsed.envvars	maximum.clients	Limita o número de clientes (e, portanto, seus encadeamentos) em um conjunto de encadeamento do RSE.
rsed.envvars	maximum.threads	Limita o número de encadeamentos do cliente em um conjunto de encadeamento do RSE.
FEJJCNFG	MAX_THREADS	Limita o número de encadeamentos no JES Job Monitor.

**Nota:** O valor para `maximum.threads` em `rsed.envvars` deve ser inferior ao valor para `MAXTHREADS` e `MAXTHREADTASKS` em `BPXPRMxx`.

---

## Uso de Armazenamento

O RSE é um aplicativo Java, que implica que o planejamento de uso de armazenamento (memória) do Developer para System z deve levar dois limites de alocação de armazenamento em consideração, o tamanho de heap Java e o tamanho do Espaço de Endereço.

### Limite de Tamanho de Heap Java

Java oferece vários serviços para facilitar os esforços de codificação para aplicativos Java. Um desses serviços é o gerenciamento de armazenamento.

O gerenciamento de armazenamento Java aloca grandes blocos de armazenamento e os usa para pedidos de armazenamento pelo aplicativo. Esse armazenamento gerenciado por Java é chamado de heap Java. A coleta de lixo periódica (desfragmentação) reclama o espaço não utilizado no heap e reduz o seu tamanho.

O tamanho de heap máximo Java é definido em `rsed.envvars` com a diretiva `Xmx`. Se essa diretiva não for especificada, Java usará um tamanho padrão de 64 MB.

Cada conjunto de armazenamento do RSE (que atende às ações do cliente) é um aplicativo Java separado e, portanto, tem um heap Java pessoal. Observe que todos os conjuntos de encadeamento usam o mesmo arquivo de configuração `rsed.envvars` e, portanto, têm o mesmo limite de tamanho de heap Java.

O uso do conjunto de encadeamento do heap Java depende muito das ações executadas pelos clientes conectados. O monitoramento regular do uso de heap é



necessário para definir o limite de tamanho de heap ideal. Use o comando do operador **modificar processo de exibição** para monitorar o uso de heap Java por conjuntos de encadeamento do RSE.

## Limite de Tamanho do Espaço de Endereço

Todos os aplicativos z/OS, incluindo aplicativos Java, estão ativos dentro de um espaço de endereço, e estão, portanto, ligados pelas limitações de tamanho do espaço de endereço.

O tamanho do espaço de endereço desejado é especificado durante a inicialização, por exemplo, com o parâmetro REGION em JCL. Entretanto, as configurações do sistema podem limitar o tamanho do espaço de endereço real. Consulte “Tamanho do Espaço de Endereço” na página 150 para saber mais sobre esses limites.

- MAXASSIZE em SYS1.PARMLIB(BPXPRMxx)
- ASSIZEMAX no segmento OMVS do ID do usuário designado à tarefa iniciada
- saídas do sistema IEFUSI e IEALIMIT

Os conjuntos de encadeamento do RSE herdam os limites de tamanho do espaço de endereço do daemon RSE. O tamanho do espaço de endereço deve ser suficiente para abrigar o heap Java, o próprio Java, as áreas de armazenamento comuns e todos os blocos de controle que o sistema cria para suportar a atividade do conjunto de encadeamento, como um TCB (Bloco de Controle de Tarefa) por encadeamento. Observe que algum desse uso de armazenamento está abaixo da linha de 16 MB.

Você deve monitorar o tamanho do espaço de endereço real antes de alterar quaisquer configurações que o influenciem, como alterar o tamanho do heap Java ou a quantidade de usuários suportada por um único conjunto de encadeamento. Use o software de monitoramento regular do sistema para controlar o uso de armazenamento real pelo Developer para system z. Se você não tiver uma ferramenta de monitoramento dedicada, então informações básicas podem ser reunidas com ferramentas como a visualização SDSF DA ou TASID (uma ferramenta de informações do sistema como elas estão armazenadas no banco de dados disponível por meio da página da Web “Support and downloads” do ISPF).

## Diretrizes de Estimativa de Tamanho

Conforme mencionado antes, o uso de armazenamento real pelo Developer para System z é altamente influenciado pela atividade do usuário. Algumas ações usam uma quantidade fixa de armazenamento (por exemplo, logon), enquanto outras são variáveis (por exemplo, listando conjuntos de dados com um qualificador de alto nível especificado).

- Use um espaço de endereço de 2 GB para RSE para permitir espaço para o heap Java e todos os blocos de controle do sistema.
- A configuração rsed.envvars de amostra é planejada para 60 usuários por conjunto de encadeamento.
  - maximum.clients=60
  - maximum.threads=1000 ( $10 + 16 \times 60 = 970$ , portanto 1000 é permitido para 61 clientes)
- A configuração rsed.envvars de amostra permite que o heap Java cresça até 256 MB. Isso permite 60 clientes usando uma média de 4 MB por cliente ( $60 \times 4 = 240$ ).

Observe que o RSE exibe o heap Java atual e o limite de tamanho de espaço de endereço durante a inicialização na mensagem do console FEK004I.

Use um dos seguintes cenários se o monitoramento mostrar que o tamanho de heap Java atual é insuficiente para a carga de trabalho real:

- Aumente o tamanho de heap Java máximo com a diretiva `Xmx` em `rsed.envvars`. Antes de fazer isso, verifique se há espaço no espaço de endereço para o aumento de tamanho.
- Diminua o número máximo de clientes por conjunto de encadeamentos com a diretiva `maximum.clients` em `rsed.envvars`. O RSE ainda suportará o mesmo número de clientes, mas os clientes serão distribuídos entre mais conjuntos de encadeamento.

## **Análise do Uso de Armazenamento de Amostra**

As telas nas figuras a seguir mostram alguns números de amostra do uso de recursos para uma configuração padrão do Developer para System z com uma modificação. O tamanho máximo de heap Java é definido como 10 MB, uma vez que um máximo pequeno resultará em um uso percentil maior e os limites do tamanho de heap serão atingidos com mais rapidez.

Tamanho Máx Heap=10 MB e privado AS Tamanho=1,959 MB

inicialização

BPXM023I (STCRSE)  
ID do Processo(268 ) Uso de Memória(7%) Clientes(0)

Nome Tarefa	Tempo de CPU	Armazen.	EXCP
JMON	0,01	2740	72
LOCKD	1,60	28,7 M	14183
RSED	4,47	32,8 M	15910
RSED8	1,15	27,4 M	12612

logon 1

BPXM023I (STCRSE)  
ID do Processo(268 ) Uso de Memória(13%) Clientes(1)

Nome Tarefa	Tempo de CPU	Armazen.	EXCP
JMON	0,01	2864	81
LOCKD	1,64	28,8 M	14259
RSED	4,55	32,8 M	15980
RSED8	3,72	55,9 M	24128

logon 2

BPXM023I (STCRSE)  
ID do Processo(268 ) Uso de Memória(23%) Clientes(2)

Nome Tarefa	Tempo de CPU	Armazen.	EXCP
JMON	0,02	2944	86
LOCKD	1,66	28,9 M	14268
RSED	4,58	32,9 M	16027
RSED8	4,20	57,8 M	25205

logon 3

BPXM023I (STCRSE)  
ID do Processo(268 ) Uso de Memória(37%) Clientes(3)

Nome Tarefa	Tempo de CPU	Armazen.	EXCP
JMON	0,02	3020	91
LOCKD	1,67	29,0 M	14277
RSED	4,60	32,9 M	16076
RSED8	4,51	59,6 M	26327

logon 4

BPXM023I (STCRSE)  
ID do Processo(268 ) Uso de Memória(41%) Clientes(4)

Nome Tarefa	Tempo de CPU	Armazen.	EXCP
JMON	0,02	3108	96
LOCKD	1,68	29,0 M	14286
RSED	4,61	32,9 M	16125
RSED8	4,77	62,3 M	27404

Figura 54. Uso de recursos com 5 logons

logon 5

BPXM023I (STCRSE)  
ID do Processo(268 ) Uso de Memória(41%) Clientes(4)  
ID do Processo(33554706) Uso de Memória(13%) Clientes(1)

Nome Tarefa	Tempo de CPU	Armazen.	EXCP
JMON	0,03	3184	101
LOCKD	1,69	29,1 M	14295
RSED	4,64	32,9 M	16229
RSED8	4,78	62,4 M	27413
RSED9	4,60	56,6 M	24065

Figura 55. Uso de recursos com 5 logons (continuação)

A Figura 54 na página 225 e a Figura 55 mostram um cenário em que 5 clientes efetuam logon em um daemon RSE com um heap Java de 10 MB.

- Um conjunto de encadeamento (RSED8) está em um estado inativo na inicialização, usando cerca de 27 MB, dos quais 0,7 MB estão no heap Java (7% de 10 MB).
- O conjunto de encadeamento se torna ativo quando o primeiro cliente se conecta, usando outros 27 MB mais 2 MB para cada cliente que se conecta.
- Parte desses 2 MB por conexão estarão no heap Java, como mostra o aumento no uso do heap.
- Entretanto, não há um padrão real no uso do heap, pois ele depende dos mecanismos Java que avaliam o armazenamento necessário e alocam mais do que o necessário. A coleta de lixo intermitente libera o armazenamento, tornando as tendências ainda mais difíceis de serem detectadas.
- Mecanismos internos que limitam o número de conexões por conjunto de encadeamento para garantir tamanho de heap suficiente para os encadeamentos ativos resultam na quinta conexão que está sendo criada em um novo conjunto de encadeamento (RSED9). Essas redes de segurança interna não são normalmente invocadas ao usar uma configuração configurada corretamente, pois outros limites serão atingidos primeiro (a maioria provavelmente `maximum.clients` em `rsed.envvars`).

Tamanho Máx Heap=10 MB e privado AS Tamanho=1,959 MB

inicialização

BPXM023I (STCRSE)  
ID do Processo(212 ) Uso de Memória(7%) Clientes(0)

Nome Tarefa	Tempo de CPU	Armazen.	EXCP
JMON	0,01	2736	71
LOCKD	1,73	30,5 M	14179
RSED	4,35	32,9 M	15117
RSED8	1,43	27,4 M	12609

logon

BPXM023I (STCRSE)  
ID do Processo(212 ) Uso de Memória(13%) Clientes(1)

Nome Tarefa	Tempo de CPU	Armazen.	EXCP
JMON	0,01	2864	80
LOCKD	1,76	30,6 M	14255
RSED	4,48	33,0 M	15187
RSED8	3,53	53,9 M	24125

expandir árvore MVS grande (195 conjuntos de dados)

BPXM023I (STCRSE)  
ID do Processo(212 ) Uso de Memória(13%) Clientes(1)

Nome Tarefa	Tempo de CPU	Armazen.	EXCP
JMON	0,01	2864	80
LOCKD	1,78	30,6 M	14255
RSED	4,58	33,1 M	16094
RSED8	4,28	56,1 M	24740

expandir PDS pequeno (21 membros)

BPXM023I (STCRSE)  
ID do Processo(212 ) Uso de Memória(13%) Clientes(1)

Nome Tarefa	Tempo de CPU	Armazen.	EXCP
IBMUSER2	0,22	2644	870
JMON	0,01	2864	80
LOCKD	1,78	30,6 M	14255
RSED	4,61	33,1 M	16108
RSED8	4,40	56,2 M	24937

abrir membro de tamanho médio (86 linhas)

BPXM023I (STCRSE)  
ID do Processo(212 ) Uso de Memória(13%) Clientes(1)

Nome Tarefa	Tempo de CPU	Armazen.	EXCP
IBMUSER2	0,22	2644	870
JMON	0,01	2864	80
RSED	4,61	33,1 M	16108
RSED8	8,12	62,7 M	27044

Figura 56. Uso de recursos ao editar um membro PDS

A Figura 56 mostra um cenário em que 1 cliente efetua logon em um daemon RSE com um heap Java de 10 MB e edita um membro PDS.

- A procura no catálogo que resulta em 195 nomes de conjuntos de dados usou cerca de 2MB de armazenamento, tudo devido à atividade do sistema, pois o uso do heap Java não aumenta.
- A abertura do PDS de 21 membros dificilmente usa alguma memória do conjunto de encadeamento, mas a tela mostra que o serviço TSO Commands foi invocado. Há um novo espaço de endereço ativo (IBMUUSER2), que usa o tamanho da região designado a esse ID do usuário em TSO. Esse espaço de endereço permanece ativo por uma quantidade de tempo específica, de modo que ele pode ser reutilizado em pedidos futuros pelo serviço TSO Commands.
- A abertura de um membro mostra números semelhantes, enquanto expande um qualificador de alto nível. O uso do heap Java permanece o mesmo, mas há um aumento no armazenamento de 6,5 MB devido à atividade do sistema.

---

## Uso do Espaço do Sistema de Arquivos z/OS UNIX

A maioria dos dados relacionados ao Developer para System z, que não são gravados em uma instrução DD, termina em um arquivo z/OS UNIX. O programador de sistema tem controle sobre os dados que são gravados e para onde eles vão. Entretanto, não há controle sobre a quantidade de dados gravada.

Os dados podem ser agrupados nas seguintes categorias:

- Análise do problema (arquivos de log e de dump do sistema), no qual vários detalhes são documentados no Capítulo 9, “Resolução de Problemas de Configuração”, na página 135
- Auditoria, conforme documentado em “Criação de Log de Auditoria” na página 163
- Dados temporários

Conforme documentado no Capítulo 9, “Resolução de Problemas de Configuração”, na página 135, o Developer for System z grava os logs do host relacionados ao RSE nos seguintes diretórios do z/OS UNIX:

- /var/rdz/logs para os logs de tarefas iniciados no RSE
- /var/rdz/logs/\$LOGNAME para logs do usuário

Por padrão, apenas as mensagens de erro e de aviso são gravadas nos logs. Portanto, se tudo correr conforme o planejado, esses diretórios devem manter apenas os arquivos vazios ou quase vazios(sem contar os logs de auditoria).

Você pode ativar a criação de log de mensagens informativas, preferivelmente na direção do centro de suporte IBM, o que aumenta perceptivelmente o tamanho dos arquivos de log.

inicialização

```
$ ls -l /var/rdz/logs
total 144
-rw-rw-rw- 1 STCRSE STCGRP 33642 Jul 10 12:10 rsedaemon.log
-rw-rw-rw- 1 STCRSE STCGRP 1442 Jul 10 12:10 rseserver.log
```

logon

```
$ ls -l /var/rdz/logs
total 144
drwxrwxrwx 3 IBMUSER SYS1 8192 Jul 10 12:11 IBMUSER
-rw-rw-rw- 1 STCRSE STCGRP 36655 Jul 10 12:11 rsedaemon.log
-rw-rw-rw- 1 STCRSE STCGRP 1893 Jul 10 12:11 rseserver.log
$ ls -l /var/rdz/logs/IBMUSER
total 160
-rw-rw-rw- 1 IBMUSER SYS1 3459 Jul 10 12:11 ffs.log
-rw-rw-rw- 1 IBMUSER SYS1 0 Jul 10 12:11 ffsget.log
-rw-rw-rw- 1 IBMUSER SYS1 0 Jul 10 12:11 ffsput.log
-rw-rw-rw- 1 IBMUSER SYS1 303 Jul 10 12:11 lock.log
-rw-rw-rw- 1 IBMUSER SYS1 126 Jul 10 12:11 rmt_classloader_cache.jar
-rw-rw-rw- 1 IBMUSER SYS1 7266 Jul 10 12:11 rsecomm.log
-rw-rw-rw- 1 IBMUSER SYS1 0 Jul 10 12:11 stderr.log
-rw-rw-rw- 1 IBMUSER SYS1 0 Jul 10 12:11 stdout.log
```

logoff

```
$ ls -l /var/rdz/logs
total 80
drwxrwxrwx 3 IBMUSER SYS1 8192 Jul 10 12:11 IBMUSER
-rw-rw-rw- 1 STCRSE STCGRP 36655 Jul 10 12:11 rsedaemon.log
-rw-rw-rw- 1 STCRSE STCGRP 2208 Jul 10 12:11 rseserver.log
$ ls -l /var/rdz/logs/IBMUSER
total 296
-rw-rw-rw- 1 IBMUSER SYS1 6393 Jul 10 12:11 ffs.log
-rw-rw-rw- 1 IBMUSER SYS1 0 Jul 10 12:11 ffsget.log
-rw-rw-rw- 1 IBMUSER SYS1 0 Jul 10 12:11 ffsput.log
-rw-rw-rw- 1 IBMUSER SYS1 609 Jul 10 12:11 lock.log
-rw-rw-rw- 1 IBMUSER SYS1 126 Jul 10 12:11 rmt_classloader_cache.jar
-rw-rw-rw- 1 IBMUSER SYS1 45157 Jul 10 12:11 rsecomm.log
-rw-rw-rw- 1 IBMUSER SYS1 0 Jul 10 12:11 stderr.log
-rw-rw-rw- 1 IBMUSER SYS1 176 Jul 10 12:11 stdout.log
```

parar

```
$ ls -l /var/rdz/logs
total 80
drwxrwxrwx 3 IBMUSER SYS1 8192 Jul 10 12:11 IBMUSER
-rw-rw-rw- 1 STCRSE STCGRP 36655 Jul 10 12:11 rsedaemon.log
-rw-rw-rw- 1 STCRSE STCGRP 2490 Jul 10 12:12 rseserver.log
```

Figura 57. Uso do espaço do sistema de arquivos z/OS UNIX

A Figura 57 mostra o uso mínimo do espaço do sistema de arquivos z/OS UNIX ao usar o nível de depuração 2 (mensagens informativas).

- Os logs de tarefas iniciadas usam 34 KB após a inicialização e crescem lentamente quando os usuários efetuam logon, efetuam logoff ou os comandos do operador são emitidos.
- Um diretório de log do cliente usa 11 KB após o logon e cresce em um ritmo uniforme quando o usuário começa a trabalhar (não mostrado no exemplo).
- Efetuar logoff incluir outros 40 KB nos logs do usuário, deixando-os com 51 KB.

Exceto para logs de auditoria, os arquivos de log são sobrescritos em cada reinício (para a tarefa iniciada do RSE) ou logon (de um cliente), mantendo o tamanho total em verificação. A diretiva `keep.last.log` em `rsed.envvars` altera isso

ligeiramente, uma vez que ela pode instruir o RSE a manter uma cópia dos logs anteriores. As cópias mais antigas são sempre removidas.

Uma mensagem de aviso é enviada ao console quando o sistema de arquivos que contém os arquivos de log de auditoria estiver em execução com pouco espaço livre e a auditoria estiver ativa. Essa mensagem do console (FEK103E) é repetida regularmente até que o problema de pouco espaço seja resolvido. Consulte “Mensagens do Console” na página 130 para obter uma lista de mensagens do console geradas pelo RSE.

As definições na Tabela 46 controlam os dados que são gravados nos diretórios de log e onde os diretórios estão localizados.

*Tabela 46. Diretivas de saída do log*

Local	Diretriz	Funções
resecomm.properties	debug_level	Definir o nível de detalhes do log padrão.
rsed.envvars	keep.last.log	Manter uma cópia dos logs anteriores antes da inicialização/logon.
rsed.envvars	enable.audit.log	Manter um rastreamento de auditoria de ações do cliente.
rsed.envvars	enable.standard.log	Gravar os fluxos stdout e stderr do conjunto (ou conjuntos) de encadeamento em um arquivo de log.
rsed.envvars	DSTORE_TRACING_ON	Ativar a criação de log de ações do DataStore.
rsed.envvars	DSTORE_MEMLOGGING_ON	Ativar a criação de log do uso de memória do DataStore.
Comando do operador	modificar rsecommlog <level>	Altera dinamicamente o nível de detalhes do log de rsecomm.log
Comando do operador	modificar rsedaemonlog <level>	Altera dinamicamente o nível de detalhes do log de rsedaemon.log
Comando do operador	modificar rseserverlog <level>	Altera dinamicamente o nível de detalhe do log de rseserver.log
Comando do operador	modificar rsestandardlog {on   off}	Altera dinamicamente a atualização de std*.log
rsed.envvars	daemon.log	Caminho inicial para a tarefa iniciada do RSE e os logs de auditoria.
rsed.envvars	user.log	Caminho inicial para logs do usuário.

O Developer para System z, junto com o software obrigatório, como o ISPF Client Gateway, grava também dados temporários em /tmp e em /var/rdz/WORKAREA. A quantidade de dados gravada aqui como resultado de ações do usuário é imprevisível, portanto você deve ter um amplo espaço livre nos sistemas de arquivos que mantêm esses diretórios.



O Developer para System z sempre tenta limpar esses arquivos temporários, mas a limpeza manual, conforme documentada em “(Opcional) Limpeza de WORKAREA” na página 105, pode ser executada a qualquer momento virtualmente.

---

## Definições de Recursos Principais

### /etc/rdz/rsed.envvars

As variáveis de ambiente definidas em rsed.envvars são usadas por RSE, Java e z/OS UNIX. O arquivo de amostra fornecido com o Developer para System z destina-se a instalações de tamanho médio e pequeno que não exigem os componentes opcionais do Developer para System z. “Arquivo de Configuração do RSE rsed.envvars” na página 31 descreve cada variável definida no arquivo de amostra, em que as seguintes variáveis precisam de atenção especial:

**\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Xms128m -Xmx256m"**

Defina o tamanho de heap inicial (Xms) e máximo (Xmx). Os padrões são 128M e 256M, respectivamente. Altere para aplicar os valores de tamanho de heap desejados. Se esta diretiva for comentada a linha, os valores padrão do Java serão usados, que são 4M e 512M, respectivamente (1M e 64M para Java 5.0).

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

Quantidade máxima de clientes atendidos por um conjunto de encadeamentos. O padrão é 60. Remova o comentário e customize para limitar o número de clientes por conjunto de encadeamentos. Observe que outros limites podem impedir que o RSE atinja esse limite.

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

Valor máximo dos encadeamentos ativos em um conjunto de encadeamentos para permitir clientes novos. O padrão é 1000. Remova o comentário e customize para limitar o número de clientes por conjunto de encadeamentos baseado no número de encadeamentos em uso. Observe que cada conexão do cliente utiliza vários encadeamentos (16 ou mais) e que outros limites podem impedir que o RSE atinja esse limite.

**Nota:** Esse valor deve ser inferior ao configurado para MAXTHREADS e MAXTHREADTASKS, em SYS1.PARMLIB(BPXPRMxx).

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Dminimum.threadpool.process=10"**

O número mínimo de conjuntos de encadeamentos ativos. O padrão é 1. Remova o comentário e customize para iniciar pelo menos o número listado de processos do conjunto de encadeamentos. Os processos do conjunto de encadeamentos são usados para balanceamento de carga dos encadeamentos do servidor RSE. Mais novos processos serão iniciados quando forem necessários. Iniciar os novos processos diretamente ajuda a evitar atrasos na conexão, mas usa mais recursos durante os tempos inativos.

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

O número máximo de conjuntos de encadeamentos ativos. O padrão é 100. Remova o comentário e customize para limitar o número de processos do conjunto de encadeamentos. Os processos do conjunto de encadeamentos são usados para balanceamento de carga dos encadeamentos do servidor RSE e a limitação deles limitará a quantidade de conexões do cliente ativas.

### SYS1.PARMLIB(BPXPRMxx)

O RSE é um aplicativo Java, que significa que ele está ativo no ambiente z/OS UNIX. Isso promove o BPXPRMxx a se tornar um membro parmlib crucial, uma

vez que ele contém os parâmetros que controlam o ambiente e os sistemas de arquivos do z/OS UNIX. O BPXPRMxx é descrito no *MVS Initialization and Tuning Reference* (SA22-7592). As seguintes diretivas são conhecidas por impactar o Developer para System z:

#### **MAXPROCSYS(nnnnn)**

Especifica o número máximo de processos que o sistema permite.

Intervalo de Valor: nnnnn é um valor decimal de 5 a 32767.

Padrão: 900

#### **MAXPROCUSER(nnnnn)**

Especifica o número máximo de processos que um único ID de usuário do z/OS UNIX pode ter ativo simultaneamente, independentemente de como os processos foram criados.

Intervalo de Valor: nnnnn é um valor decimal de 3 a 32767.

Padrão: 25

#### **Nota:**

- Todos os processos RSE usam o mesmo ID do usuário do z/OS UNIX (aquele do usuário que está designado ao daemon do RSE), pois todos os clientes são executados como encadeamentos dentro dos processos RSE.
- Esse valor pode ser definido também com a variável PROCUSERMAX no segmento de perfil de segurança OMVS do usuário designado à tarefa iniciada do RSED.

#### **MAXTHREADS(nnnnnn)**

Especifica o número máximo de encadeamentos pthread\_created, incluindo execução, enfileiramento e saída, exceto desconexão, que um único processo pode ter ativado simultaneamente. Especificar um valor de 0 impede que os aplicativos usem pthread\_create.

Intervalo de Valor: nnnnnn é um valor decimal de 0 a 100000.

Padrão: 200

#### **Nota:**

- Cada cliente usa pelo menos 16 encadeamentos dentro do processo do conjunto de encadeamentos do RSE, e vários clientes são ativados dentro do processo.
- Esse valor pode ser configurado também com a variável THREADSMAX no segmento do perfil de segurança OMVS do usuário designado à tarefa iniciada do RSED. Quando configurado, o valor THREADSMAX é usado para MAXTHREADS e MAXTHREADTASKS.

#### **MAXTHREADTASKS(nnnnn)**

Especifica o número máximo de tarefas MVS que um único processo pode ter ativado simultaneamente para encadeamentos pthread\_created.

Intervalo de Valor: nnnnn é um valor decimal de 0 a 32768.

Padrão: 1000

#### **Nota:**

- Cada encadeamento ativo possui uma tarefa MVS (TCB, Bloco de Controle da Tarefa).

- Cada tarefa MVS simultânea exige armazenamento adicional, algumas das quais deve ficar abaixo da linha 16 MB.
- Cada cliente usa pelo menos 16 encadeamentos dentro do processo do conjunto de encadeamentos do RSE, e vários clientes são ativados dentro do processo.
- Esse valor pode ser configurado também com a variável THREADSMAX no segmento do perfil de segurança OMVS do usuário designado à tarefa iniciada do RSED. Quando configurado, o valor THREADSMAX é usado para MAXTHREADS e MAXTHREADTASKS.

#### **MAXUIDS(nnnnn)**

Especifica o número máximo de IDs do usuário (UIDs) do z/OS UNIX que podem funcionar simultaneamente.

Intervalo de Valor: nnnnn é um valor decimal de 1 a 32767.  
Padrão: 200

#### **MAXASSIZE(nnnnn)**

Especifica os valores de recursos RLIMIT\_AS que serão estabelecidos como os valores iniciais para os novos processos. RLIMIT\_AS indica o tamanho da região de espaço de endereço.

Intervalo de Valor: nnnnn é um valor decimal de 10485760 (10 Megabytes) a 2147483647 (2 Gigabytes).  
Padrão: 209715200 (200 Megabytes)

#### **Nota:**

- Esse valor deve ser configurado como 2 G.
- Esse valor pode ser configurado com a variável ASSIZEMAX no segmento de perfil de segurança OMVS do usuário designado à tarefa iniciada do RSED.

#### **MAXFILEPROC(nnnnnn)**

Especifica o número máximo de descritores para arquivos, soquetes, diretórios e quaisquer outros objetos do sistema de arquivos que um único processo pode ter ativado ou alocado simultaneamente.

Intervalo de Valor: nnnnnn é um valor decimal de 3 a 524287.  
Padrão: 64000

#### **Nota:**

- Um conjunto de encadeamentos possui todos os encadeamentos de clientes em um único processo.
- Esse valor pode ser configurado também com a variável FILEPROCMAX no segmento do perfil de segurança OMVS do usuário designado à tarefa iniciada do RSED.

#### **MAXMMAPAREA(nnnnn)**

Especifica a quantidade máxima de espaço de armazenamento do espaço de dados (em páginas) que pode ser alocada para mapeamentos de memória de arquivos z/OS UNIX. O armazenamento não é alocado até que o mapeamento de memória esteja ativado.

Intervalo de Valor: nnnnn é um valor decimal de 1 a 16777216.  
Padrão: 40960

**Nota:** Esse valor pode ser configurado também com a variável MMAPAREAMAX no segmento do perfil de segurança OMVS do usuário designado à tarefa iniciada do RSED.

Use o comando do operador **SETOMVS** ou **SET OMVS** para aumentar ou diminuir dinamicamente (até o próximo IPL) o valor de quaisquer variáveis BPXPRMxx anteriores. Para fazer uma alteração permanente, edite o membro BPXPRMxx que será usado para IPLs. Consulte *MVS System Commands* (SA22-7627) para obter informações adicionais sobre esses comandos do operador.

As definições a seguir são subparâmetros da instrução NETWORK.

#### **MAXSOCKETS(nnnnnnnn)**

Especifica o número máximo de soquetes suportados por esse sistema de arquivos para essa família de endereços. Esse é um parâmetro opcional.

Intervalo de Valor: nnnnnnnn é um valor decimal de 0 a 16777215.  
Padrão: 100

#### **INADDRANYCOUNT(nnnn)**

Especifica o número de portas que o sistema reserva para uso com a PORT 0, as ligações INADDR\_ANY, começando com o número de porta especificado no parâmetro INADDRANYPORT. Esse valor é necessário apenas para CINET (várias pilhas TCP/IP).

Intervalo de Valor: nnnn é um valor decimal de 1 a 4000.  
Padrão: Se nenhum INADDRANYPORT ou INADDRANYCOUNT for especificado, o padrão para INADDRANYCOUNT será 1000.  
Caso contrário, nenhuma porta será reservada (0).

---

## **Definições de Vários Recursos**

### **Placa EXEC na JCL do Servidor**

Recomenda-se que as definições a seguir sejam incluídas na placa JCL dos servidores Developer para System z.

#### **REGION=0M**

REGION=0M é recomendado para o daemon RSE e as tarefas iniciadas do JES Job Monitor, RSED e JMON, respectivamente. Fazendo isso, o tamanho do espaço de endereço fica limitado apenas pelo armazenamento privado disponível, ou pelas saídas do sistema IEFUSI ou IEALIMIT. Observe que é altamente recomendado pela IBM que essas saídas não sejam usadas para espaços de endereços z/OS UNIX, como o daemon RSE.

#### **TIME=NOLIMIT**

Recomenda-se que TIME=NOLIMIT seja usado para todos os servidores Developer para System z. Isso porque o tempo de CPU de todos os clientes do Developer para System z se acumula nos espaços de endereço do servidor.

### **FEK.#CUST.PARMLIB(FEJJCNFG)**

As variáveis de ambiente definidas em FEJJCNFG são usadas pelo JES Job Monitor. O arquivo de amostra fornecido com o Developer para System z destina-se a instalações de porte médio e pequeno. “FEJJCNFG, Arquivo de Configuração do JES Job Monitor” na página 27 descreve cada variável definida no arquivo de amostra, em que as seguintes variáveis exigem atenção especial:

## **MAX\_THREADS**

Número máximo de usuários que podem utilizar um Monitor de Tarefas do JES por vez. O padrão é 200. O valor máximo é 2147483647. Aumentar este número pode exigir o aumento do tamanho do espaço de endereços do JES Job Monitor.

## **SYS1.PARMLIB(IEASYSxx)**

IEASYSxx mantém parâmetros do sistema e é descrito no *MVS Initialization and Tuning Reference* (SA22-7592). As seguintes diretivas são conhecidas por impactar o Developer para System z:

### **MAXUSER=nnnnn**

Esse parâmetro especifica um valor que, na maioria das condições, o sistema usa para limitar o número de tarefas e as tarefas iniciadas que podem ser executadas simultaneamente durante um IPL específico.

Intervalo de Valor: nnnnn é um valor decimal de 0 a 32767. Observe que a soma dos

valores especificados para os parâmetros do sistema  
MAXUSER, RSVSTRT

e RSVNONR não pode exceder 32767.

Valor Padrão: 255

## **SYS1.PARMLIB(IVTPRMxx)**

IVTPRMxx configura parâmetros para o Communication Storage Manager (CSM), e é descrito no *MVS Initialization and Tuning Reference* (SA22-7592). As seguintes diretivas são conhecidas por impactar o Developer para System z:

### **FIXED MAX(maxfix)**

Define a quantidade máxima de armazenamento dedicada a buffers fixos do CSM.

Intervalo de Valor: maxfix é um valor de 1024K a 2048M.

Padrão: 100M

### **ECSA MAX(maxecsa)**

Define a quantidade máxima de armazenamento dedicada a buffers do ECSA CSM.

Intervalo de Valor: maxecsa é um valor de 1024K a 2048M.

Padrão: 100M

## **SYS1.PARMLIB(ASCHPMxx)**

O membro parmlib ASCHPMxx contém informações de planejamento para o planejador de transações ASCH e é descrito no *MVS Initialization and Tuning Reference* (SA22-7592). As seguintes diretivas são conhecidas por impactar o Developer para System z:

### **MAX(nnnnn)**

Um parâmetro opcional da definição CLASSADD que especifica o número máximo de iniciadores de transações APPC que são permitidos em uma determinada classe de iniciadores de transações. Depois que esse limite for atingido, nenhum novo espaço de endereço será criado e os pedidos recebidos serão enfileirados para aguardar até que os espaços de endereço do iniciador existentes se tornem disponíveis. O valor não deve exceder o número máximo

de espaços de endereço permitido por sua instalação e você deve estar ciente dos produtos concorrentes no sistema que também exigirão espaços de endereço.

Intervalo de Valor: nnnnn é um valor decimal de 1 a 64000.

Padrão: 1

**Nota:** Se você usar APPC para iniciar o serviço TSO Commands, então a classe de transações usada deve ter iniciadores de transações suficientes para permitir um iniciador para cada usuário concorrente do Developer para System z.

---

## Monitoramento

Como as cargas de trabalho do usuário podem alterar a necessidade de recursos do sistema, o sistema deve ser monitorado regularmente para medir o uso de recursos de modo que o Rational Developer para System z e as configurações do sistema possam ser ajustadas em resposta aos requisitos do usuário. Os comandos a seguir podem ser usados para ajudar nesse processo de monitoramento.

### Monitoramento de RSE

Os conjuntos de encadeamentos RSE são o ponto focal para a atividade do usuário no Developer para System z e, assim, exigem monitoramento para o uso ideal. O daemon RSE pode ser consultado sobre informações que não podem ser reunidas com as ferramentas de monitoramento comuns do sistema.

- Use as ferramentas de monitoramento comuns do sistema, como RMF, para reunir dados específicos de espaço de endereço, como armazenamento real e tempo de CPU. Se você não tiver uma ferramenta de monitoramento dedicada, então as informações básicas poderão ser reunidas com ferramentas como a visualização SDSF DA ou TASID (uma ferramenta de informações do sistema como elas estão armazenadas no banco de dados disponível por meio da página da Web “Support and Downloads”).
- Durante a inicialização, o daemon RSE relata o tamanho do espaço de endereço disponível e o tamanho de heap Java com a mensagem do console FEK004I.  
FEK004I RseDaemon: Tamanho Máximo de Heap=65MB e Tamanho do AS privado=1,959MB
- O comando do operador **MODIFY RSED,APPL=DISPLAY PROCESS** exibe os processos do conjunto de encadeamentos do RSE. O campo “Uso de Memória” mostra quanto do heap Java definido é realmente usado. Consulte o Capítulo 8, “Comandos do Operador”, na página 123 para obter informações adicionais sobre esse comando.

```
f rsed,appl=d p
BPXM023I (STCRSE)
ID do processo(16777456) Uso de Memória(33%) Clientes(4) Ordem(1)
```

Informações adicionais são fornecidas quando a opção DETAIL do comando de modificação do **DISPLAY PROCESS** é usado:

```
f rsed,appl=d p,detail
BPXM023I (STCRSE)
ID do processo(33555087) ASId(002E) Nome da Tarefa (RSED8) Ordem(1)
PROCESS LIMITS:    CURRENT  HIGHWATER    LIMIT
JAVA HEAP USAGE(%) 10      56           100
CLIENTS             0      25           60
MAXFILEPROC         83     103         64000
MAXPROCUSER         97     99           200
MAXTHREADS          9      14          1500
MAXTHREADTASKS      9      14          1500
```

## Monitorando z/OS UNIX

A maioria dos limites z/OS UNIX que é do interesse do Developer para System z pode ser exibida usando comandos do operador. Alguns comandos mostram ainda o uso atual e a limite máximo de um limite específico. Consulte *MVS System Commands* (SA22-7627) para obter informações adicionais sobre esses comandos.

- A diretiva LIMMSG(ALL) em SYS1.PARMLIB(BPXPRMxx) informa ao z/OS UNIX para exibir mensagens do console (BPXI040I) quando qualquer dos limites parmlib estiver prestes a ser atingido. O valor padrão de LIMMSG é NONE, que desativa a função. Use o comando do operador **SETOMVS LIMMSG=ALL** para ativar dinamicamente essa função (até o próximo IPL). Consulte *MVS Initialization and Tuning Reference* (SA22-7592) para obter informações adicionais sobre essa diretiva.
- O comando do operador **DISPLAY OMVS,OPTIONS** exibe os valores atuais das diretivas z/OS UNIX que podem ser definidas dinamicamente.

```
d omvs,o
BPX0043I 13.10.16 DISPLAY OMVS 066
OMVS 000D ETC/INIT WAIT OMVS=(M7)
CURRENT UNIX CONFIGURATION SETTINGS:
MAXPROCSYS      =      256    MAXPROCUSER      =      16
MAXFILEPROC     =      256    MAXFILESIZE      = NOLIMIT
MAXCPUPTIME     =      1000    MAXUIDS       =      200
MAXPTY          =      256
MAXMMAPAREA     =      256    MAXASSIZE      = 209715200
MAXTHREADS      =      200    MAXTHREADTASKS =      1000
MAXCORESIZE     = 4194304    MAXSHAREPAGES =      4096
IPCMSGQBYTES    = 2147483647  IPCMSGQMNUM  =     10000
IPCMSGNIDS      =      500    IPCSEMNIDS   =      500
IPCSEMNOPS      =      25     IPCSEMNSEMS  =     1000
IPCshmPAGES     =     25600    IPCshMNIDS   =      500
IPCshMNSEGS     =      500    IPCshMSPAGES =    262144
SUPERUSER       = BPXROOT     FORKCOPY       = COW
STEPLIBLIST     =
USERIDALIASTABLE=
SERV_LINKLIB    = POSIX.DYNSERV.LOADLIB  BPXLK1
SERV_LPALIB     = POSIX.DYNSERV.LOADLIB  BPXLK1
PRIORITYPG VALUES: NONE
PRIORITYGOAL VALUES: NONE
MAXQUEUEDSIGS   =      1000    SHRLIBRGNSIZE =    67108864
SHRLIBMAXPAGES  =      4096    VERSION       = /
SYSCALL COUNTS  = NO          TTYGROUP        = TTY
SYSplex        = NO          BRLM SERVER     = N/A
LIMMSG          = NONE       AUTOCVT         = OFF
RESOLVER PROC   = DEFAULT
AUTHPGMLIST     = NONE
SWA             = BELOW
```

- O comando do operador **DISPLAY OMVS,LIMITS** exibe informações sobre limites parmlib atuais dos Serviços do Sistema z/OS UNIX, seus limites máximos e o uso atual do sistema.

```
d omvs,l
BPX0051I 14.05.52 DISPLAY OMVS 904
OMVS 0042 ACTIVE OMVS=(69)
SYSTEM WIDE LIMITS: LIMMSG=SYSTEM
CURRENT HIGHWATER SYSTEM
USAGE USAGE LIMIT
MAXPROCSYS      1         4      256
MAXUIDS          0         0      200
MAXPTY          0         0      256
MAXMMAPAREA     0         0      256
MAXSHAREPAGES   0        10     4096
IPCMSGNIDS      0         0      500
IPCSEMNIDS      0         0      500
IPCshMNIDS      0         0      500
```



```

IPCSHMPAGES          0          0      262144 *
IPCMSGQBYTES         ---          0      262144
IPCMSGQMNUM          ---          0      10000
IPCSHMPAGES          ---          0       256
SHRLIBRGNSIZE        0          0     67108864
SHRLIBMAXPAGES       0          0       4096

```

O comando exibe limites máximos e o uso atual de um processo individual quando a palavra-chave PID=processid também for especificada.

```

d,omvs,l,pid=16777456
BPX0051I 14.06.28 DISPLAY OMVS 645
OMVS      000E ACTIVE          OMVS=(76)
USER      JOBNAME ASID        PID      PPID STATE   START   CT_SECS
STCRSE    RSED8   007E      16777456  67109106 HF---- 20.00.56 113.914
LATCHWAITPID=      0 CMD=java -Ddaemon.log=/var/rdz/logs -
PROCESS LIMITS:    LIMMSG=NONE
                  CURRENT  HIGHWATER  PROCESS
                  USAGE    USAGE      LIMIT

MAXFILEPROC      83      103      256
MAXFILESIZE       ---      ---      NOLIMIT
MAXPROCUSER      97      99      200
MAXQUEUEDSIGS     0        1        1000
MAXTHREADS       9       14      200
MAXTHREADTASKS   9       14      1000
IPCSHMNSEGS       0        0        500
MAXCORESIZE       ---      ---      4194304
MAXMEMLIMIT       0        0        16383P

```

- O comando do operador **DISPLAY OMVS,PFS** exibe informações sobre cada sistema de arquivo físico que faz parte atualmente da configuração do z/OS UNIX, que inclui as pilhas TCP/IP.

```

d omvs,p
BPX0046I 14.35.38 DISPLAY OMVS 092
OMVS      000E ACTIVE          OMVS=(33)
PFS CONFIGURATION INFORMATION
PFS TYPE      DESCRIPTION          ENTRY      MAXSOCK  OPNSOCK  HIGHUSED
TCP          SOCKETS AF_INET        EZBPFINI   50000   244     8146
UDS           SOCKETS AF_UNIX        BPXTUINIT   64        6        10
ZFS           LOCAL FILE SYSTEM      IOEFSCM
              14:32.00 RECYCLING
HFS           LOCAL FILE SYSTEM      GFUAINIT
BPXFTCLN     CLEANUP DAEMON      BPXFTCLN
BPXFTSYN     SYNC DAEMON        BPXFTSYN
BPXFPINT     PIPE                BPXFPINT
BPXFCSIN     CHAR SPECIAL          BPXFCSIN
NFS          REMOTE FILE SYSTEM      GFSCINIT
PFS NAME     DESCRIPTION          ENTRY      STATUS   FLAGS
TCP41        SOCKETS                EZBPFINI   ACT      CD
TCP42        SOCKETS                EZBPFINI   ACT
TCP43        SOCKETS                EZBPFINI   INACT    SD
TCP44        SOCKETS                EZBPFINI   INACT
PFS PARM INFORMATION
HFS          SYNCDEFAULT(60) FIXED(50) VIRTUAL(100)
              CURRENT VALUES: FIXED(55) VIRTUAL(100)
NFS          biod(6)

```

- O comando do operador **DISPLAY OMVS,PID=processid** exibe as informações de encadeamento de um processo específico.

```

d omvs,pid=16777456
BPX0040I 15.30.01 DISPLAY OMVS 637
OMVS      000E ACTIVE          OMVS=(76)
USER      JOBNAME ASID        PID      PPID STATE   START   CT_SECS
STCRSE    RSED8   007E      16777456  67109106 HF---- 20.00.56 113.914
LATCHWAITPID=      0 CMD=java -Ddaemon.log=/var/rdz/logs -
THREAD_ID  TCB@      PRI_JOB  USERNAME  ACC_TIME SC STATE
0E08A00000000000 005E6DF0 OMVS          .927 RCV  FU

```



0E08F00000000001	005E6C58		.001	PTX	JYNV
0E09300000000002	005E6AC0		7.368	PTX	JYNV
0E0CB00000000008	005C2CF0	OMVS	1.872	SEL	JFNV
0E1920000000003CE	005A0B70	OMVS	14.088	POL	JFNV
0E18D0000000003CF	005A1938	IBMUSER	.581	SND	JYNV

## Monitoramento da Rede

Ao suportar um grande número de clientes que se conectam ao host, não apenas o Developer para System z, mas também sua infraestrutura de rede deve ser capaz de tratar a carga de trabalho. O gerenciamento de rede é um assunto amplo e bem documentado que fica fora do escopo da documentação do Developer para System z. Portanto, apenas os seguintes ponteiros são fornecidos.

- O comando do operador **DISPLAY NET,CSM** permite que você monitore o uso de armazenamento gerenciado pelo gerenciador de armazenamento de comunicações (CSM). Você pode usar esse comando para determinar a quantidade de armazenamento do CSM que está em uso no ECSA e os conjuntos de armazenamento de espaço de dados, conforme documentado em *Communications Server SNA Operations* (SC31-8779).

## Monitorando Sistemas de Arquivos z/OS UNIX

O Developer para System z usa sistemas de arquivos z/OS UNIX para armazenar vários tipos de dados, como arquivos de log e temporários. Use o comando z/OS UNIX **df** para verificar quantos descritores de arquivo ainda estão disponíveis e quanto espaço livre foi deixado antes que a próxima extensão do conjunto de dados subjacente HFS ou zFS seja criada.

```
$ df
Mounted on    Filesystem      Avail/Total    Files      Status
/tmp          (OMVS.TMP)      1393432/1396800 4294967248 Available
/u/ibmuser    (OMVS.U.IBMUSER) 1248/1728      4294967281 Available
/usr/lpp/rdz  (OMVS.FEK.HHOP760) 3062/43200     4294967147 Available
/var          (OMVS.VAR)      27264/31680    4294967054 Available
```

## Configuração de Amostra

A configuração de amostra a seguir mostra a configuração necessária para suportar estes requisitos:

- 500 conexões do cliente simultâneas
- 300 builds MVS simultâneos (tarefa em lote)
- 200 conexões CARMA simultâneas (usando o método de inicialização CRASTART)
- tempo limite de inatividade de 3 horas
- uso não autorizado de z/OS UNIX
- SCLM Developer Toolkit e File Manager Integration não são usados
- calcule uma média de uso de heap Java de 5 MB
- os usuários têm UIDs z/OS UNIX exclusivos

## Contagem do Conjunto de Encadeamento

Por padrão, o Developer para System z tenta incluir 60 usuários em um único conjunto de encadeamento. Entretanto, nossos requisitos indicam que o tempo limite de inatividade estará ativo. A Tabela 44 na página 219 mostra que isso incluirá 1 encadeamento por cliente conectado. Esse encadeamento é um encadeamento de cronômetro e portanto ativo constantemente. Isso evitará que o RSE coloque 60 usuários em um único conjunto de encadeamento, uma vez que  $60 \times (16+1) = 1020$  e `maximum.threads` é definido como 1000, por padrão.

Poderíamos aumentar `maximum.threads`, mas devido ao requisito ter uma média de 5 MB de heap Java por usuário, optamos por diminuir o `maximum.clients` para 50. Isso nos mantém dentro do tamanho máximo padrão de heap Java de 256 MB ( $5 \times 50 = 250$ ).

Com 50 clientes por conjunto de encadeamentos e a necessidade de suportar 500 conexões, sabemos agora que precisaremos de 10 espaços de endereço de conjunto de encadeamentos.

## Determinar Limites Mínimos

Usando as fórmulas mostradas anteriormente neste capítulo e os critérios mencionados no início desta seção, podemos determinar o uso do recurso que deve ser adaptado.

- Contagem de espaço de endereço - máximo  
 $3 + A + N \times (x + y + z) + (2 + N \times 0.01)$   
 $3 + 10 + 500 \times 1 + 200 \times 1 + 300 \times 1 + (2 + 500 \times 0.01) = 1020$
- Contagem de espaço de endereço - por usuário  
 $x + y + z$   
 $1 + 1 + 1 = 3$
- Contagem de processos - máximo  
 $7 + 2 \times A + N \times (x + y + z) + (10 + N \times 0.05)$   
 $7 + 2 \times 10 + 500 \times 2 + 200 \times 1 + 300 \times 0 + (10 + 500 \times 0.05) = 1562$
- Contagem de processos - por usuário  
 $(x + y + z) + 5 \times s$   
 $(2 + 1 + 0) + 5 \times 0 = 3$
- Contagem de encadeamentos - Conjunto de encadeamentos do RSE  
 $9 + N \times (16 + x + y + z) + (20 + N \times 0.1)$   
 $9 + 60 \times (16 + 1 + 4 + 0) + (20 + 60 \times 0.1) = 1295$
- Contagem de encadeamentos - JES Job Monitor  
 $3 + N$   
 $3 + 500 = 503$
- IDs do Usuário  
 $500 + 3 = 503$   

Os 3 IDs do usuário extras são para STCJMON, STCLOCK e STCRSE, os IDs do usuário da tarefa iniciada do Developer para System z.

## Definindo Limites

Agora que os números de uso de recursos são conhecidos, podemos customizar a limitação das diretivas com valores apropriados.

- `/etc/rdz/rsed.envvars`
  - `Xmx256m`
  - não alterado
  - `Dmaximum.clients=50`
  - `Dmaximum.threads=1000`
  - não alterado
  - `Dminimum.threadpool.process=10`

Esta mudança é opcional; o RSE iniciará novos conjuntos de encadeamentos, conforme necessário

- DHide\_ZOS\_UNIX=true
- DDSTORE\_IDLE\_SHUTDOWN\_TIMEOUT=10800000
- FEK.#CUST.PARMLIB(FEJJCNFG)
  - MAX\_THREADS=503
- SYS1.PARMLIB(BPXPRMxx)
  - MAXPROCSYS(2500)

1562 mínimo, incluído buffer extra para tarefas diferentes do Developer para System z

- MAXPROCUSER(25)

não alterado, mínimo 3

- MAXTHREADS(1500)

deve ser no mínimo 503 (para JES Job Monitor) se THREADSMAX no segmento OMVS do ID do usuário STCRSE for usado para configurar o limite para RSE (mínimo 1295)

- MAXTHREADTASKS(1500)

deve ser no mínimo 503 (para JES Job Monitor) se THREADSMAX no segmento OMVS do ID do usuário STCRSE for usado para configurar o limite para RSE (mínimo 1295)

- MAXUIDS(700)

503 mínimo, incluído buffer extra para tarefas diferentes do Developer para System z

- MAXASSIZE(209715200)

não alterado (padrão do sistema 200 MB), usamos ASSIZEMAX no segmento OMVS do ID do usuário STCRSE

- SYS1.PARMLIB(IEASYSxx)
  - MAXUSER=2000

1020 mínimo, incluído buffer extra para tarefas diferentes do Developer para System z

- Segmento OMVS do ID do usuário STCRSE
  - ASSIZEMAX(2147483647)

2 GB

## Uso de Recurso de Monitor

Após ativar os limites do sistema como documentado em “Definindo Limites” na página 240, é possível iniciar a monitoração do uso dos recursos pelo Developer para System z para verificar se é necessário o ajuste de algumas variáveis. O Figura 58 na página 242 mostra o uso dos recursos após 495 usuários terem efetuado logon. (O exemplo na figura apenas mostra a criação do logon. Nenhuma

ação do usuário é indicada no exemplo).

```
BPXM023I (STCRSE)
ProcessId(16779764) Memory Usage(10%) Clients(50) Order(1)
ProcessId(67108892) Memory Usage(16%) Clients(50) Order(2)
ProcessId(67108908) Memory Usage(10%) Clients(50) Order(3)
ProcessId(67108898) Memory Usage(16%) Clients(50) Order(4)
ProcessId(67108916) Memory Usage(16%) Clients(50) Order(5)
ProcessId(67108897) Memory Usage(16%) Clients(50) Order(6)
ProcessId(67108921) Memory Usage(16%) Clients(50) Order(7)
ProcessId(83886146) Memory Usage(16%) Clients(50) Order(8)
ProcessId(67108920) Memory Usage(16%) Clients(50) Order(9)
ProcessId(3622      ) Memory Usage(8%) Clients(45) Order(10)
```

Nome	Tarefa	Tempo de CPU	Armazen.	EXCP
-----	-----	-----	-----	-----
JMON		1.74	43.0M	2753
LOCKD		10.05	31.9M	24621
RSED		6.65	40.1M	41780
RSED1		8.17	187.0M	76566
RSED2		13.04	184.9M	78946
RSED3		17.77	181.1M	76347
RSED4		11.63	174.9M	74638
RSED5		15.27	172.9M	72883
RSED6		13.85	180.8M	75031
RSED7		9.79	174.3M	76636
RSED8		21.59	176.1M	70583
RSED8		18.88	184.7M	76953
RSED9		9.52	189.8M	80490

Figura 58. Uso do recurso de configuração de amostra

---

## Capítulo 14. 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 para System z.

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

---

### Usar Sistemas de Arquivos zFS

O zFS (zSeries File System) e o HFS (Hierarchical File System) são sistemas de arquivo UNIX que podem ser usados em um ambiente z/OS UNIX. No entanto, o zFS fornece os seguintes recursos e benefícios:

- Ganhos de desempenho em vários ambientes do cliente ao acessar arquivos com tamanhos aproximados a 8 K, frequentemente acessados e atualizados. O desempenho do acesso de arquivos menores equivale àquele do HFS.
- Clonagem de leitura de um sistema de arquivo no mesmo conjunto de dados. O sistema de arquivo clonado pode ser disponibilizado aos usuários para fornecer uma cópia de leitura em um determinado momento de um sistema de arquivo. Esse é um recurso opcional disponível apenas em um ambiente não-sysplex.
- O zFS é o sistema de arquivo z/OS UNIX estratégico. A funcionalidade do HFS foi estabilizada e os aprimoramentos no sistema de arquivo ocorrerão apenas para o zFS.

Consulte *UNIX System Services Planning* (GA22-7800) para saber mais sobre zFS.

---

### Evite o Uso de STEPLIB

Cada processo do z/OS UNIX que possui um STEPLIB que é propagado de pai para filho ou através de um exec consumirá cerca de 200 bytes de Extended Common Storage Area (ECSA). Se nenhuma variável de ambiente STEPLIB estiver definida, ou quando uma for definida como STEPLIB=CURRENT, o z/OS UNIX propagará todas as alocações TASKLIB, STEPLIB e JOBLIB atualmente ativas durante uma função fork(), spawn() ou exec().

O Developer para System z possui um padrão de arquivo de configuração STEPLIB=NONE codificado em rsed.envvars, conforme descrito em rsed.envvars. 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).

---

### Aprimorar o Acesso às Bibliotecas do Sistema

Determinadas bibliotecas do sistema e módulos de carregamento são intensamente usadas pelo z/OS UNIX e pelas atividades de desenvolvimento do aplicativo. O aprimoramento do acesso, como a inclusão na Área do Pacote de Links (LPA) pode aprimorar o desempenho do sistema. Consulte *MVS Initialization and Tuning Reference* (SA22-7592) para obter informações adicionais sobre a alteração de membros SYS1.PARMLIB descrito a seguir:

## Bibliotecas de Tempo de Execução do Ambiente de Linguagem (LE)

Quando programas C (incluindo o shell do z/OS UNIX) são executados, frequentemente usam 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.

CEE.SCEELPA

O conjunto de dados CEE.SCEELPA contém um subconjunto das rotinas de tempo de execução do LE, intensamente usadas pelo z/OS UNIX. Você deve incluir esse conjunto de dados em 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(*) DSN(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(LNKLISTxx) 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, será necessário configurar a instrução STEPLIB apropriada no arquivo de configuração rsed.envvars, conforme descrito em rsed.envvars. 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 de aplicativos é a principal atividade, o desempenho também pode ser beneficiado se você colocar o editor de ligação em um LPA dinâmico, incluindo as seguintes linhas em SYS1.PARMLIB(PROGxx):

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

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

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

---

## 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 informações adicionais sobre esse perfil.

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

---

## 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 para System z pode ser equilibrado pela configuração das metas corretas para os processos. Algumas diretrizes gerais são listadas abaixo:

- Quando usado, designe a transação APPC para um grupo de desempenho do TSO.
- Designe um grupo de desempenho de tarefa iniciada (SYSSTC) para os espaços de endereço do servidor Developer para System z: JES Job Monitor (JMON), Daemon de bloqueio (LOCKD), daemon RSE (RSED) e conjuntos de encadeamentos do RSE (RSEDx).

Consulte *MVS Planning Workload Management* (SA22-7602) para obter informações adicionais sobre este assunto.

---

## 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 ideia, 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 *Java Diagnostics Guide* (SC34-6650) para obter informações adicionais sobre tamanhos de heap e coleta de lixo.

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

No Developer para System z, as opções da linha de comandos Java são definidas na diretiva `_RSE_JAVA0PTS` de `rsed.envvars`, conforme descrito em “Definindo Parâmetros Extras de Inicialização Java com `_RSE_JAVA0PTS`” na página 42.



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

---

## Opção Java -Xquickstart

**Nota:** -Xquickstart Java é útil apenas se você utilizar o método de inicialização alternativo REXEC/SSH para o servidor RSE. Esse método é documentado em "(Opcional) Usando REXEC (ou SSH)" na página 100.

A opção -Xquickstart pode ser usada para melhorar o tempo de inicialização de alguns aplicativos Java. -Xquickstart faz com que o compilador Just In Time (JIT) seja executado com um subconjunto de otimizações, ou seja, uma compilação rápida. Essa compilação rápida permite melhorar o tempo de inicialização.

-Xquickstart é apropriada para aplicativos de curta execução, principalmente aqueles em que o tempo de execução não está concentrado em um número menor de métodos. -Xquickstart pode prejudicar o desempenho se for usada em aplicativos de longa execução que contêm métodos ativos.

Para ativar a opção -Xquickstart para o servidor RSE, inclua a seguinte diretiva no final de `rsed.envvars`:

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

---

## Compartilhamento de Classe entre JVMs

O IBM Java Virtual Machine (JVM) versão 5 e superior permite compartilhar classes de aplicativos e de autoinicialização entre JVMs armazenando-as em um cache em memória compartilhada. O compartilhamento de classes reduz o consumo total de memória virtual quando mais de uma JVM compartilha um cache. O compartilhamento de classes também reduz o tempo de inicialização de uma JVM após o cache ser criado.

O cache de classe compartilhada é independente de qualquer JVM ativa e persiste além do tempo de vida da JVM que criou o cache. Como o cache de classe compartilhada persiste além do tempo de vida de qualquer JVM, o cache é atualizado dinamicamente para refletir quaisquer modificações que possam ter sido feitas em JARs ou classes no sistema de arquivo.

A sobrecarga para criar e preencher um novo cache é mínima. O custo de tempo de inicialização da JVM para uma única JVM normalmente é entre 0 e 5% menor em comparação com um sistema que não utiliza compartilhamento de classe, dependendo de quantas classes são carregadas. O aprimoramento do tempo de inicialização da JVM com um cache preenchido normalmente é entre 10% e 40% mais rápido em comparação com um sistema que não utiliza compartilhamento de classe, dependendo do sistema operacional e do número de classes carregadas. Várias JVMs em execução simultaneamente mostram maiores benefícios no tempo de inicialização total.

Consulte o *Java SDK and Runtime Environment User Guide* para obter informações adicionais sobre o compartilhamento de classe.

## Ativar Compartilhamento de Classes

Para ativar o compartilhamento de classes para o servidor RSE, inclua a seguinte diretiva no final de `rsed.envvars`. A primeira instrução define um cache denominado RSE com acesso em grupo e permite que o servidor RSE seja iniciado, mesmo se o compartilhamento de classes falhar. A segunda instrução é opcional e



configura o tamanho do cache para 6 megabytes (o padrão do sistema é 16 MB). A terceira instrução inclui os parâmetros de compartilhamento de classes nas opções de inicialização Java.

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

**Nota:** Conforme mencionado em “Segurança do Cache”, todos os usuários que usam a classe compartilhada devem ter o mesmo ID do grupo (GID) 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 em seu segmento OMVS.

## Limites de Tamanho de Cache

O tamanho máximo de cache compartilhado teórico é 2 GB. O tamanho de cache que você pode especificar é limitado pela quantidade de memória física e pelo espaço de troca disponíveis para o sistema. Como o espaço de endereço virtual de um processo é compartilhado entre o cache de 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 de classe compartilhada é limitado por permissões do sistema operacional e permissões de segurança Java.

Por padrão, os caches de classe são criados com a segurança no nível do usuário, portanto, apenas o usuário que criou o cache pode acessá-lo. No z/OS UNIX, há uma opção, `groupAccess`, que fornece acesso a todos os usuários no grupo primário do usuário que criou o cache. Entretanto, independentemente do nível de acesso usado, um cache poderá ser destruído apenas pelo usuário que o criou ou por um usuário root (UID 0).

Consulte o *Java SDK and Runtime Environment User Guide* para obter informações adicionais sobre as opções extras de segurança utilizando um Java SecurityManager.

## SYS1.PARMLIB(BPXPRMxx)

Algumas das configurações de SYS1.PARMLIB(BPXPRMxx) afetam o desempenho das classes compartilhadas. O uso de configurações incorretas pode interromper o funcionamento de classes compartilhadas. Essas configurações também podem ter implicações no desempenho. Para obter informações adicionais sobre implicações no desempenho e sobre o uso desses parâmetros, consulte *MVS Initialization and Tuning Reference* (SA22-7592) e *UNIX System Services Planning* (GA22-7800). Os parâmetros BPXPRMxx mais significativos que afetam a operação de classes compartilhadas são os seguintes:

- MAXSHAREPAGES, IPCSHMSPAGES, IPCSHMPAGES e IPCSHMNSEGS

Essas configurações afetam a quantidade de páginas de memória compartilhada disponíveis para a JVM. O tamanho da página compartilhada para um serviço do sistema z/OS UNIX de 31 bits é fixado em 4 KB. As classes compartilhadas tentam criar um cache de 16 MB por padrão. Entretanto, configure IPCSHMPAGES para maior que 4096.

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

- IPCSEMNIDS e IPCSEMNSEMS

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

## Espaço em Disco

O cache de classe compartilhada requer espaço em disco para armazenar informações de identificação sobre os caches que existem 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 de Cache

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

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

Could not create the Java virtual machine.

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

Current statistics for cache "RSE":

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

```
cache size        = 6291368
free bytes        = 4355696
ROMClass bytes    = 1912272
Metadata bytes    = 23400
Metadata % used   = 1%
```

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

Cache is 30% full

Could not create the Java virtual machine.

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

### Nota:

- Os utilitários de cache executam a operação necessária no cache especificado sem iniciar a JVM, por isso a mensagem "Could not create the Java virtual machine" é normal.

- Um cache pode ser destruído apenas se todas as JVMs que o utilizam estiverem encerradas e o usuário que emite o comando tiver permissões suficientes.



---

## Capítulo 15. Considerações sobre o CICSTS

Tradicionalmente, a função de definição de recursos para o CICS tem sido o domínio do administrador do CICS. Há uma resistência em permitir que o desenvolvedor de aplicativos defina recursos do CICS por vários motivos:

- A maioria das definições de recursos do CICS tem muitos parâmetros que, devido à complexidade, ao inter-relacionamento com outras definições de recurso e aos padrões de compra, requerem conhecimento de administrador do CICS para obter uma definição correta. Definições incorretas podem causar resultados inesperados que podem afetar a região inteira do CICS.
- A maioria das lojas de clientes fornece ambientes de desenvolvimento e de teste do CICS que devem estar disponíveis para uso compartilhado por vários grupos de aplicativos e desenvolvedores. Muitas lojas de cliente têm Acordos de Nível de Serviço em vigor para esses ambientes. Para atender a esses contratos, é necessário controle estrito dos ambientes.

O Developer para System z aborda esses problemas permitindo que administradores do CICS controlem os padrões de definição de recurso do CICS e controlem as propriedades de exibição de um parâmetro de definição de recurso do CICS por meio do servidor CICS Resource Definition (CRD), que faz parte do Application Deployment Manager.

Por exemplo, o administrador do CICS pode fornecer certos parâmetros de definição de recurso do CICS que podem não ser atualizados pelo desenvolvedor de aplicativos. Outros parâmetros de definição de recurso do CICS podem ser atualizados, com ou sem os padrões fornecidos, ou o parâmetro de definição de recurso do CICS pode ser ocultado para evitar uma complexidade desnecessária.

Quando o desenvolvedor de aplicativos estiver satisfeito com as definições de recursos do CICS, elas poderão ser instaladas imediatamente no ambiente de teste do CICS em execução, ou poderão ser exportadas em um manifesto para edição e aprovação adicionais por um administrador do CICS. O administrador do CICS pode utilizar o administrative utility (utilitário em lote) ou a ferramenta Processamento de Manifesto para implementar as alterações de definição de recurso.

**Nota:** A ferramenta Processamento de Manifesto é um plug-in para o IBM CICS Explorer.

Consulte Capítulo 4, “(Opcional) Application Deployment Manager”, na página 71 para obter informações adicionais sobre as tarefas necessárias para configurar o Application Deployment Manager no sistema host.

Customizar o Application Deployment Manager inclui os seguintes serviços no Developer para System z:

- (no cliente) O IBM CICS Explorer fornece uma infraestrutura baseada em Eclipse para visualizar e gerenciar recursos CICS e ativa maior integração entre as ferramentas CICS
- (no cliente) Editor do CICS Resource Definition (CRD)
- (no host) Servidor CICS Resource Definition (CRD), que é executado como um aplicativo do CICS

O servidor CICS Resource Definition (CRD) do Application Deployment Manager consiste no próprio servidor CRD, um repositório CRD, definições de recursos CICS associadas e, ao usar a interface dos Serviços da Web, os arquivos de ligação do Serviço da Web e um manipulador de mensagens do pipeline de amostra. O servidor CRD deve ser executado em uma Web Owning Region (WOR), que é mencionado na documentação do Developer para System z como a região de conexão primária do CICS.

Consulte o Centro de Informações do Developer para System z (<http://publib.boulder.ibm.com/infocenter/ratdevz/v7r6/index.jsp>) para saber mais sobre os serviços disponíveis do Application Deployment Manager no release atual do Developer para System z.

---

## RESTful versus Serviços da Web

O CICS Transaction Server fornece na versão 4.1 e superior suporte para uma interface HTTP projetada usando os princípios do Representational State Transfer (RESTful). Essa interface RESTful é agora a interface CICSTS estratégica para uso por aplicativos clientes. A interface de Serviço da Web mais antiga foi estabilizada e os aprimoramentos serão apenas para a interface RESTful.

O Application Deployment Manager segue essa instrução de direção e exige o servidor RESTful CRD para todos os serviços que são novos no Developer para System versão 7.6 ou superior.

As interfaces RESTful e de Serviço da Web podem ser ativadas simultaneamente em uma única região do CICS, se desejado. Nesse caso, haverá dois servidores CRD ativos na região. Os dois servidores compartilharão o mesmo repositório CRD. Observe que o CICS emitirá alguns avisos sobre definições duplicadas quando a segunda interface for definida para a região.

---

## Regiões de Conexão Primária versus Não-primária

Um ambiente de teste do CICS pode consistir de várias regiões Multi-Region Option (MRO) conectadas. Com o tempo, foram utilizadas designações não oficiais para classificar essas regiões. As designações típicas são Terminal Owning Region (TOR), Web Owning Region (WOR), Application Owning Region (AOR) e Data Owning Region (DOR).

Uma Web Owning Region é usada para implementar suporte aos Serviços da Web do CICS e o servidor CICS Resource Definition (CRD) do Application Deployment Manager deve ser executado nessa região. Essa região é conhecida pelo Application Deployment Manager como a região de conexão primária do CICS. O cliente do CRD implementa uma conexão de serviço da Web na região de conexão primária do CICS.

As regiões de conexão não-primárias do CICS são todas as outras regiões que o servidor CRD pode atender. Esse serviço inclui visualizar recursos utilizando IBM CICS Explorer e definir recursos utilizando o editor de definição de recurso do CICS.

Se o CICSplex SM Business Application Services (BAS) for usado para gerenciar as definições de recurso do CICS da região de conexão primária do CICS, todas as outras regiões do CICS gerenciadas pelo BAS poderão ser atendidas pelo servidor CRD.

As regiões do CICS não gerenciadas pelo BAS requerem alterações adicionais para poderem ser atendidas pelo servidor CRD.

---

## Log de Instalação do Recurso do CICS

As ações feitas pelo servidor CRD em relação aos recursos do CICS são registradas na fila do CICS CSDL TD, que normalmente aponta para a DD MSGUSR de sua região do CICS.

Se CICSplex SM Business Application Services (BAS) for usado para gerenciar suas definições de recursos do CICS, então a diretiva CICSplex SM EYUPARM BASLOGMSG deve ser configurada como (YES) para que os logs sejam criados.

---

## segurança do Application Deployment Manager

### Segurança do Repositório do CRD

O conjunto de dados de VSAM do repositório do servidor CRD contém todas as definições de recurso padrão e deve, portanto, ser protegido contra atualizações, mas os desenvolvedores devem ter permissão para ler os valores armazenados aqui. Consulte “Definir os Perfis do Conjunto de Dados” na página 178 para obter comandos RACF de amostra para proteger o repositório do CRD.

### Segurança de Pipeline

Quando uma mensagem SOAP for recebida pelo CICS por meio da interface de Serviço da Web, ela será processada por um pipeline. Um pipeline é um conjunto de manipuladores de mensagens que são executados em seqüência. O CICS lê o arquivo de configuração do pipeline para determinar quais manipuladores de mensagens devem ser chamados no pipeline. Um manipulador de mensagem é um programa no qual você pode executar processamento especial de pedidos e respostas de serviço da Web.

O Application Deployment Manager fornece um arquivo de configuração de pipeline de amostra que especifica a chamada de um manipulador de mensagens e um programa de processamento de cabeçalho SOAP.

O manipulador de mensagens do pipeline (ADNTMSGH) é usado para segurança através do processamento do ID do usuário e da senha no cabeçalho SOAP. ADNTMSGH é referido pelo arquivo de configuração de pipeline de amostra e, portanto, deve ser colocado na concatenação RPL do CICS.

### Segurança da Transação

CPIH é o ID da transação padrão que um aplicativo chamado por um pipeline executará. Geralmente, o CPIH é configurado para um nível mínimo de autorização.

O Developer para System z fornece várias transações que são usadas pelo servidor CRD durante a definição e a consulta de recursos do CICS. Esses IDs de transação são configurados pelo servidor CRD, dependendo da operação solicitada. Consulte o Capítulo 4, “(Opcional) Application Deployment Manager”, na página 71 para obter informações adicionais sobre a customização de IDs da transação.

Transação	Descrição
ADMS	Para pedidos da ferramenta Processamento de Manifesto para alterar recursos do CICS. Geralmente, isso é destinado aos administradores do CICS. Essa transação requer um alto nível de autorização.
ADMI	Para pedidos que definam, instalem ou desinstalem recursos do CICS. Essa transação pode requerer um nível médio de autorização, dependendo das políticas do site.
ADMR	Para todos os outros pedidos que recuperam as informações de ambiente e de recurso do CICS. Essa transação pode requerer um nível mínimo de autorização, dependendo das políticas do site.

Alguns ou todos esses pedidos de definição de recurso feitos pelas transações do servidor CRD devem ser protegidos. No mínimo, os comandos de atualização (parâmetros de atualização padrão do serviço da Web, parâmetros padrão do descritor e o nome de arquivo para ligação do nome do conjunto de dados) devem ser protegidos para impedir que todos, exceto os administradores do CICS, emitam esses comandos usados para configurar padrões de recurso global.

Quando a transação é conectada, a verificação de segurança de recurso do CICS, se ativada, garante que o ID do usuário esteja autorizado para executar o ID de transação.

A verificação de recursos é controlada pela opção RESSEC na transação que está executando o parâmetro de inicialização do sistema RESSEC e, para o servidor do CRD, o parâmetro de inicialização do sistema XPCT.

A verificação de recursos ocorre apenas se o parâmetro de inicialização do sistema XPCT tiver um valor diferente de NO e a opção RESSEC da definição TRANSACTION for YES ou o parâmetro de inicialização do sistema RESSEC for ALWAYS.

Os seguintes comandos RACF mostram como as transações do servidor CRD podem ser protegidas. Consulte o *RACF Security Guide for CICSTS* para obter informações adicionais sobre a definição da segurança do CICS.

- RALTER GCICSTRN SYSADM UACC(NONE) ADDMEM(ADMS)
- PERMIT SYSADM CLASS(GCICSTRN) ID(#cicsadmin)
- RALTER GCICSTRN DEVELOPER UACC(NONE) ADDMEM(ADMI)
- PERMIT DEVELOPER CLASS(GCICSTRN) ID(#cicsdeveloper)
- RALTER GCICSTRN ALLUSER UACC(READ) ADDMEM(ADMR)
- SETROPTS RACLIST(TCICSTRN) REFRESH

## Comunicação Criptografada por SSL

A criptografia SSL do fluxo de dados é suportada quando o cliente do Application Deployment Manager usa a interface dos Serviços da Web para invocar o servidor CRD. O uso de SSL para essa comunicação é controlado pela palavra-chave SSL(YES) na definição CICSTS TCIPSERVICE, conforme documentado em *RACF Security Guide para CICSTS*.



## Segurança do Recurso

O CICSTS fornece a capacidade de proteger os recursos e os comandos para manipulá-los. Algumas ações do Application Deployment Manager podem falhar se a segurança estiver ativada, mas não configurada completamente (por exemplo, concedendo permissões para manipular novos tipos de recursos).

Em caso de falha da função no Application Deployment Manager, examine os logs do CICS para obter mensagens como a seguir, e executar ações corretivas, conforme documentado no *RACF Security Guide para CICSTS*.

```
DFHXS1111 %date %time %applid %tranid Security violation by user
%userid at netname %portname for resource %resource in class
%classname. SAF codes are (X'safresp',X'safreas'). ESM codes are
(X'esmpresp',X'esmpreas').
```

---

## Administrative Utility

O Developer para System z fornece o administrative utility para permitir que administradores do CICS forneçam os valores padrão para as definições de recurso do CICS. Esses padrões podem ser somente de leitura ou podem ser editados pelo desenvolvedor de aplicativos.

O administrative utility fornece as seguintes funções:

- Nome do CICSplex para ambientes de teste gerenciados pelo CICSplex
- Nome do grupo de migração de dados do CICSplex SM
- Configuração da regra de exportação do manifesto
- Padrões de atributo de recurso do CICS e permissões de exibição
- Ligação lógica para física do CICS usada para definições de conjuntos de dados VSAM

O administrative utility é chamado pela tarefa de amostra ADNJSAPU no conjunto de dados FEK.#CUST.JCL. O uso desse utilitário requer acesso UPDATE ao repositório do CRD.

ADNJSAPU está localizado em FEK.#CUST.JCL, a menos que o programador do sistema z/OS tenha especificado um local diferente quando customizou e enviou a tarefa FEK.SFEKSAMP(FEKSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes.

**Nota:** O repositório CRD deve ser fechado no CICS antes de executar a tarefa ADNJSAPU. O repositório pode ser aberto novamente após a conclusão da tarefa. Por exemplo, após se conectar ao CICS, digite os seguintes comandos para fechar e abrir o arquivo, respectivamente:

- CEMT S FILE(ADNREPF0) CLOSED
- CEMT S FILE(ADNREPF0) OPEN

As instruções de controle de entrada são usadas para atualizar o repositório do CRD de um ambiente de teste do CICS, para o qual as regras gerais de sintaxe a seguir se aplicam:

- Um asterisco na posição 1 indica uma linha de comentário.
- Um comando DEFINE deve iniciar na posição 1, seguido por um único espaço e seguido por uma palavra-chave válida, como TRANSACTION.

- Um valor de palavra-chave deve existir imediatamente após uma palavra-chave. Não são permitidos espaços entre eles. A única exceção é para a exibição das palavras-chave de permissão UPDATE, PROTECT e HIDDEN, que não têm valores.
- Os valores de palavra-chave são colocados entre parênteses.
- Uma palavra-chave e seu valor devem estar contidos em uma única linha.

As definições de amostra a seguir seguem a estrutura dos comandos DFHCSDUP, conforme definido no *CICS Resource Definition Guide para CICSTS*. A única diferença é a inserção das seguintes palavras-chave de permissão de exibição usadas para agrupar valores de atributo em três conjuntos de permissões:

UPDATE	Os atributos que seguem essa palavra-chave serão atualizados por um desenvolvedor de aplicativos utilizando Developer para System z. Esse também é o padrão para atributos omitidos.
PROTECT	Os atributos que seguem essa palavra-chave serão exibidos, mas estarão protegidos contra atualizações por um desenvolvedor de aplicativos utilizando Developer para System z.
HIDDEN	Os atributos que seguem essa palavra-chave não serão exibidos e estarão protegidos contra atualizações por um desenvolvedor de aplicativos utilizando Developer para System z.

Consulte a seguinte amostra de código ADNJSPAU.

```

//ADNJSPAU JOB <JOB PARAMETERS>
//*
//ADNSPAU EXEC PGM=ADNSPAU,REGION=1M
//STEPLIB DD DISP=SHR,DSN=FEK.SFEKLOAD
//ADMREP DD DISP=OLD,DSN=FEK.#CUST.ADNREPF0
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
*
* Parâmetros CICSplex SM
*
DEFINE CPSMNAME( )
*DEFINE STAGINGGROUPNAME(ADMSTAGE)
*
* Regra de exportação de manifesto
*
DEFINE MANIFESTEXPORTRULE(installOnly)
*
* Padrões de definição de recurso do CICS
* Atributos omitidos são padronizados como UPDATE.
*
* Atributos padrão DB2TRAN
*
DEFINE DB2TRAN()
    UPDATE DESCRIPTION()
        ENTRY()
        TRANSID()
*
* Atributos padrão DOCTEMPLATE
*
DEFINE DOCTEMPLATE()
    UPDATE DESCRIPTION()
        TEMPLATENAME()
        FILE() TSQUEUE() TDQUEUE() PROGRAM() EXITPGM()
        DDNAME(DFHHTML) MEMBERNAME()
        HFSFILE()
        APPENDCRLF(YES) TYPE(EBCDIC)
*
* Atributos padrão de arquivo
*
DEFINE FILE()
    UPDATE DESCRIPTION()
        RECORDSIZE() KEYLENGTH()
        RECORDFORMAT(V) ADD(NO)
        BROWSE(NO) DELETE(NO) READ(YES) UPDATE(NO)
        REMOTESYSTEM() REMOTENAME()
    PROTECT DSNAME() RLSACCESS(NO) LSRPOOLID(1) STRINGS(1)
        STATUS(ENABLED) OPENTIME(FIRSTREF)
        DISPOSITION(SHARE) DATABUFFERS(2) INDEXBUFFERS(1)
        TABLE(NO) MAXNUMRECS(NOLIMIT)
        READINTEG(UNCOMMITTED) DSNSHARING(ALLREQS)
        UPDATEMODEL(LOCKING) LOAD(NO)
        JNLREAD(NONE) JOURNAL(NO)
        JNLSYNCREAD(NO) JNLUPDATE(NO)
        JNLADD(NONE) JNLSYNCSWRITE(YES)
        RECOVERY(NONE) FWDRECOVLOG(NO)
        BACKUPTYPE(STATIC)
        PASSWORD() NSRGROUP()
        CFDTPOOL() TABLENAME()

```

Figura 59. ADNJSPAU - Administrative utility do CICSTS (Parte 1 de 3)

```

*
* Atributos padrão Mapset
*
DEFINE MAPSET()
    UPDATE  DESCRIPTION()
    PROTECT RESIDENT(NO) STATUS(ENABLED)
           USAGE(NORMAL) USELPACOPY(NO)
** Atributos padrão de tipo de processo
*
DEFINE PROCESSTYPE()
    UPDATE  DESCRIPTION()
           FILE(BTS)
    PROTECT STATUS(ENABLED)
           AUDITLOG() AUDITLEVEL(OFF)
*
* Atributos padrão de programa
*
DEFINE PROGRAM()
    UPDATE  DESCRIPTION()
           CEDF(YES) LANGUAGE(LE370)
           REMOTESYSTEM() REMOTENAME() TRANSID()
    PROTECT API(CICSAPI) CONCURRENCY(QUASIRENT)
           DATALOCATION(ANY) DYNAMIC(NO)
           EXECCKEY(USER) EXECUTIONSET(FULLAPI)
           RELOAD(NO) RESIDENT(NO)
           STATUS(ENABLED) USAGE(NORMAL) USELPACOPY(NO)
    HIDDEN JVM(NO) JVMCLASS() JVMPROFILE(DFHJVMPR)
*
* Atributos padrão TDQueue
*
DEFINE TDQUEUE()
    UPDATE  DESCRIPTION()
           TYPE(INTRA)
* Parâmetros de partição extra
    DDNAME() DSNAME()
    REMOTENAME() REMOTESYSTEM() REMOTELLENGTH(1)
    RECORDSIZE() BLOCKSIZE(0) RECORDFORMAT(UNDEFINED)
    BLOCKFORMAT() PRINTCONTROL() DISPOSITION(SHR)
* Parâmetros de partição intra
    FACILITYID() TRANSID() TRIGERRLEVEL(1)
    USERID()
* Parâmetros indiretos
    INDIRECTNAME()
    PROTECT WAIT(YES) WAITACTION(REJECT)
* Parâmetros de partição extra
    DATABUFFERS(1)
    SYSOUTCLASS() ERROROPTION(IGNORE)
    OPENTIME(INITIAL) REWIND(LEAVE) TYPEFILE(INPUT)
* Parâmetros de partição intra
    ATIFACILITY(TERMINAL) RECOVSTATUS(NO)

```

*Figura 59. ADNJSPAU - Administrative utility do CICSTS (Parte 2 de 3)*

```

*
* Atributos padrão de transação
*
DEFINE TRANSACTION()
    UPDATE  DESCRIPTION()
            PROGRAM()
            TWASIZE(0)
            REMOTESYSTEM() REMOTENAME() LOCALQ(NO)
    PROTECT PARTITIONSET() PROFILE(DFHCICST)
            DYNAMIC(NO) ROUTABLE(NO)
            ISOLATE(YES) STATUS(ENABLED)
            RUNAWAY(SYSTEM) STORAGECLEAR(NO)
            SHUTDOWN(DISABLED)
            TASKDATAKEY(USER) TASKDATALOC(ANY)
            BREXIT() PRIORITY(1) TRANCLASS(DFHTCL00)
            DTIMOUT(NO) RESTART(NO) SPURGE(NO) TPURGE(NO)
            DUMP(YES) TRACE(YES) CONFDATA(NO)
            OTSTIMEOUT(NO) WAIT(YES) WAITTIME(00,00,00)
            ACTION(BACKOUT) INDOUBT(BACKOUT)
            RESSEC(NO) CMDSEC(NO)
            TRPROF()
            ALIAS() TASKREQ()
            XTRANID() TPNAME() XTPNAME()

|
| *
| * Atributos URDIMAP
| *
| DEFINE URIMAP()
|     UPDATE  USAGE(CLIENT)
|             DESCRIPTION()
|             PATH(/required/path)
|             TCPIPSERVICE()
|             TRANSACTION()
|             PROGRAM()
|     PROTECT ANALYZER(NOANALYZER)
|             ATOMSERVICE()
|             CERTIFICATE()
|             CHARACTERSET()
|             CIPHERS()
|             CONVERTER()
|             HFSFILE()
|             HOST(host.mycompany.com)
|             HOSTCODEPAGE()
|             LOCATION()
|             MEDIATYPE()
|             PIPELINE()
|             PORT(NO)
|             REDIRECTTYPE(NONE)
|             SCHEME(HTTP)
|             STATUS(ENABLED)
|             TEMPLATENAME()
|             USERID()
|             WEBSERVICE()
|
| *
| * Nome de arquivo opcional para ligações de nome do conjunto de dados VSAM
| *
| *DEFINE DSBINDING() DSNAME()
| /*

```

Figura 59. ADNJSPAU - Administrative utility do CICSTS (Parte 3 de 3)

## Notas de Migração do Utilitário Administrativo

Developer para System z versão 7.6.1 incluiu suporte URIMAP no Administrative utility. Para poder usar o suporte URIMAP, o conjunto de dados VSAM do

repositório CRD deve estar alocado com um tamanho de registro máximo de 3000. Até a Developer para System z versão 7.6.1, as tarefas de alocação do repositório CRD da amostra usam um tamanho de registro máximo de 2000.

Siga estas etapas para ativar o suporte URIMAP se você estiver usando um repositório CRD mais antigo:

1. Crie um backup do seu repositório CRD existente, FEK.#CUST.ADNREPF0.
2. Exclua o repositório CRD existente.
3. Customize e submeta a tarefa FEK.SFEKSAMP(ADNVCRD) para alocar e inicializar um novo repositório CRD. Consulte a documentação no membro para obter instruções de customização.
4. Customize e submeta a tarefa FEK.SFEKSAMP(ADNJSPAU) para usar o Administrative utility para preencher o repositório CRD novo.

**Nota:**

- Migrar o repositório CRD existente não é necessário porque o administrative utility substitui os conteúdos completos do repositório CRD cada vez que ele é executado.
- Não há problemas de compatibilidade de versão com o repositório CRD. Todo cliente suportado pelo Developer para System z e código de host trabalharão com tamanho de registro máximo. Mas o suporte URIMAP estará desativado se o tamanho de registro máximo não for 3000.

## Mensagens do Administrative Utility

As mensagens a seguir são emitidas pelo Administrative utility para a SYSPRINT DD. As mensagens CRAZ1803E, CRAZ1891E, CRAZ1892E e CRAZ1893E contêm códigos de status de arquivo, retorno do VSAM, função do VSAM e feedback do VSAM. Os códigos de retorno, função e feedback do VSAM são documentados em *DFSMS Macro Instructions for Data Sets* (SC26-7408). Os códigos de status de arquivo são documentados em *Enterprise COBOL for z/OS Language Reference* (SC27-1408).

### **CRAZ1800I**

**concluído com sucesso na linha <número da última linha de instrução de controle>**

**Explicação:** O administrative utility do programador de sistema foi concluído com sucesso.

**Resposta do usuário:** Nenhuma.

### **CRAZ1801W**

**concluído com avisos na linha <número da última linha de instrução de controle>**

**Explicação:** O administrative utility do programador de sistema foi concluído com um ou mais avisos localizados durante o processamento de instruções de controle.

**Resposta do usuário:** Verifique outras mensagens de aviso.

### **CRAZ1802E**

**encontrado um erro na linha <número da linha>**

**Explicação:** O administrative utility do programador de sistema encontrou um erro grave.

**Resposta do usuário:** Verifique outras mensagens de aviso.

**CRAZ1803E**

**Erro ao abrir o repositório, status=<código de status do arquivo>  
RC=<código de retorno do VSAM> FC=<código de função do VSAM>  
FB=<código de feedback do VSAM>**

**Explicação:** O administrative utility do programador de sistema encontrou um erro grave ao abrir o repositório do CRD.

**Resposta do usuário:** Verifique os códigos de status, retorno, função e feedback do VSAM.

**CRAZ1804E**

**Registro de entrada desconhecido na linha <número da linha>**

**Explicação:** O administrative utility do programador de sistema encontrou uma instrução de controle de entrada desconhecida.

**Resposta do usuário:** Verifique se o comando **DEFINE** foi seguido por um espaço único e depois pela palavra-chave CPSMNAME, STAGINGGROUPNAME, MANIFESTEXPORTRULE, DSBINDING, DB2TRAN, DOCTEMPLATE, FILE, MAPSET, PROCESSTYPE, PROGRAM, TDQUEUE ou TRANSACTION.

**CRAZ1805E**

**Processando palavra-chave <palavra-chave> na linha <número da linha>**

**Explicação:** O administrative utility do programador de sistema está processando a instrução de controle de entrada da palavra-chave DEFINE.

**Resposta do usuário:** Nenhuma.

**CRAZ1806E**

**Regra de exportação de manifesto inválida na linha <número da linha>**

**Explicação:** O administrative utility do programador de sistema encontrou uma regra de exportação de manifesto inválida.

**Resposta do usuário:** Verifique se o valor da palavra-chave MANIFESTEXPORTRULE é "installOnly", "exportOnly" ou "both".

**CRAZ1807E**

**Palavra-chave DSNNAME ausente na linha <número da linha>**

**Explicação:** O administrative utility do programador de sistema estava processando uma instrução de controle DEFINE DSBINDING que não possui a palavra-chave DSNNAME.

**Resposta do usuário:** Verifique se a instrução de controle DEFINE DSBINDING contém a palavra-chave DSNNAME.

**CRAZ1808E**

**Valor da palavra-chave inválido para a palavra-chave <palavra-chave> na linha <número da linha>**

**Explicação:** O administrative utility do programador de sistema estava processando uma instrução de controle DEFINE e encontrou um valor inválido para a palavra-chave nomeada.

**Resposta do usuário:** Verifique se o comprimento e o valor da palavra-chave nomeada estão corretos.

**CRAZ1890W**

**Erro de sintaxe de palavra-chave na linha <número da linha>**

**Explicação:** O administrative utility do programador de sistema estava processando uma instrução de controle DEFINE e encontrou um erro de sintaxe para a palavra-chave ou valor da palavra-chave.

**Resposta do usuário:** Verifique se o valor da palavra-chave está entre parênteses e imediatamente após a palavra-chave. A palavra-chave e o valor da palavra-chave devem estar contidos na mesma linha.

#### **CRAZ1891W**

**Erro ao gravar chave duplicada do repositório, status=<código de status do arquivo> RC=<código de retorno VSAM> FC=<código de função VSAM> FB=<código de feedback VSAM>**

**Explicação:** O administrative utility do programador de sistema encontrou um erro de chave duplicada ao gravar no repositório CRD.

**Resposta do usuário:** Verifique os códigos de status, retorno, função e feedback do VSAM.

#### **CRAZ1892W**

**Erro ao gravar repositório, status=<código de status do arquivo> RC=<código de retorno do VSAM> FC=<código de função do VSAM> FB=<código de feedback do VSAM>**

**Explicação:** O administrative utility do programador de sistema encontrou um erro grave ao gravar no repositório CRD.

**Resposta do usuário:** Verifique os códigos de status, retorno, função e feedback do VSAM.

#### **CRAZ1893W**

**Erro de leitura de repositório, status=<código de status do arquivo> RC=<código de retorno do VSAM> FC=<código de função do VSAM> FB=<código de feedback do VSAM>**

**Explicação:** O administrative utility do programador de sistema encontrou um erro grave ao ler o repositório do CRD.

**Resposta do usuário:** Verifique os códigos de status, retorno, função e feedback do VSAM.



---

## Capítulo 16. Customizando o Ambiente TSO

Este apêndice é fornecido para ajudá-lo a imitar um procedimento de logon do TSO incluindo instruções DD e conjuntos de dados no ambiente TSO no Developer para System z.

---

### O Serviço TSO Commands

O serviço TSO Commands é o componente Developer para System z que executa comandos TSO e ISPF (em lote) e retorna o resultado para o cliente solicitante. Esses comandos podem ser solicitados implicitamente pelo produto ou explicitamente pelo usuário.

Os membros de amostra fornecidos com o Developer para System z criam um ambiente mínimo do TSO/ISPF. Se os desenvolvedores em sua loja precisarem de acesso a bibliotecas customizadas ou de terceiros, o programador de sistema z/OS deve incluir as instruções DD e as bibliotecas necessárias para o ambiente de serviço TSO Commands. Embora a implementação seja diferente no Developer para System z, a lógica por trás disso é idêntica ao procedimento de logon do TSO.

**Nota:** O serviço TSO Commands é uma ferramenta de linha de comandos não interativa, portanto, os comandos ou procedimentos que solicitam dados ou exibem painéis ISPF não funcionarão. Um emulador 3270, como o Host Connect Emulator que faz parte do cliente Developer para System z, será necessário para executá-los.

### Métodos de Acesso

A partir da versão 7.1, o Developer para System z fornece uma opção para acessar o serviço TSO Commands.

- O serviço TSO/ISPF Client Gateway do ISPF, que requer um nível mínimo de serviço do ISPF. Este é o método padrão usado nas amostras fornecidas.
- Uma transação APPC (como em releases pré-versão 7.1).

**Nota:** O serviço TSO/ISPF Client Gateway do ISPF substitui a função do SCLM Developer Toolkit usada na versão 7.1.

Verifique o `rsed.envvars` para determinar qual método de acesso é usado para hosts da versão 7.1 e superior. Se os padrões tiverem sido usados durante o processo de configuração, `rsed.envvars` residirá em `/etc/rdz/`.

- Se a instrução `_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DTSO_SERVER=APPC"` não estiver presente (ou estiver assinalada como comentário, que é o padrão), o serviço TSO/ISPF Client Gateway do ISPF será usado.
- Se a instrução `_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DTSO_SERVER=APPC"` estiver presente (e não estiver assinalada como comentário), o APPC será usado.

---

## Usando o Método de Acesso do TSO/ISPF Client Gateway

### Customização Básica – ISPF.conf

O arquivo de configuração ISPF.conf (localizado por padrão em /etc/rdz/) define o ambiente do TSO/ISPF usado pelo Developer para System z. Existe apenas um arquivo de configuração ISPF.conf ativo, que é usado por todos os usuários do Developer para System z.

A seção principal do arquivo de configuração define os nomes DD e as concatenações relacionadas do conjunto de dados, como no seguinte exemplo:

```
sysproc=ISP.SISPLIB,FEK.SFEKPROC
ispmllib=ISP.SISPMENU
isptlib=ISP.SISPTENU
ispplib=ISP.SISPPENU
ispslib=ISP.SISPSLIB
ispllib=ISP.SISLOAD
myDD=HLQ1.LLQ1,HLQ2.LLQ2
```

- Cada definição de DD utiliza exatamente uma linha (não há suporte para várias linhas) e não existem limites de comprimento de linha.
- As definições não fazem distinção entre maiúsculas e minúsculas e qualquer espaço em branco será ignorado.
- As linhas de comentário iniciam com um asterisco (\*).
- Os nomes DD são seguidos por um sinal de igual (=), que por sua vez é seguido pela concatenação do conjunto de dados. Vários nomes de conjunto de dados são separados por uma vírgula (,).
- As concatenações do conjunto de dados são procuradas na ordem em que são listadas.
- Os conjuntos de dados devem ser completos, sem ser colocados entre aspas (') e sem o uso de variáveis.
- Todos os conjuntos de dados são alocados com DISP=SHR.
- Novos nomes DD podem ser incluídos à vontade, mas devem obedecer as regras (JCL) para nomes DD e não podem ser conflitantes com outros parâmetros de configuração no ISPF.conf. Além disso, o ISPPROF é alocado dinamicamente (DISP=NEW,DELETE) pelo serviço TSO/ISPF Client Gateway.

### Avançado – Usar Perfis do ISPF Existentes

Por padrão, o TSO/ISPF Client Gateway cria um perfil temporário do ISPF para o serviço TSO Commands. Entretanto, você pode instruir o TSO/ISPF Client Gateway a utilizar uma cópia de um perfil existente do ISPF. A chave aqui é a instrução `_RSE_CMDSERV_OPTS` no `rsed.envvars`.

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

Remova o comentário da instrução (remova o sinal de sustenido (#) inicial) e forneça o nome completo do conjunto de dados do perfil existente do ISPF para utilizar esse recurso.

As seguintes variáveis podem ser usadas no nome do conjunto de dados:

- `&SYSUID`. para substituir o ID do usuário do desenvolvedor
- `&SYSPREF`. para substituir o prefixo do TSO do desenvolvedor

**Nota:**

- Se o nome do conjunto de dados transmitido em "ISPPROF" for inválido, um perfil de ISPF vazio temporário será usado.
- O perfil do ISPF (temporário e copiado) é excluído no final da sessão. As alterações feitas no perfil não são mescladas com o perfil existente do ISPF.

## Avançado – Usando um Exec de Alocação

A instrução `allocjob` no `ISPF.conf` (que está assinalada como comentário por padrão) aponta um exec que pode ser usado para fornecer alocações adicionais do conjunto de dados por ID do usuário.

```
*allocjob = FEK.#CUST.CNTL(CRAISPRX)
```

Remova o comentário da instrução (remova o caractere de asterisco (\*) inicial) e forneça a referência completa para o exec de alocação para utilizar esse recurso.

- O exec é executado após a alocação de ISPPROF e os DDs definidos em `ISPF.conf`, mas antes de o ISPF ser inicializado. Assegure-se de que o exec de alocação não desfaça essas definições.
- 1 parâmetro é transmitido ao exec; o ID do usuário do responsável pela chamada.
- Um exec de amostra `CRAISPRX` é fornecido na biblioteca de amostra `FEK.#CUST.CNTL`, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa `FEK.SFEKSAMP(FEKSETUP)`. Consulte “Configuração da Customização” na página 15 para obter mais detalhes.

**Nota:** Como o exec é chamado antes do ISPF ser inicializado, você não pode utilizar `VPUT` e `VGET`. No entanto, você pode criar suas próprias implementações dessas funções utilizando um arquivo VSAM ou PDS(E).

## Avançado – Usar Vários Execs de Alocação

Embora o `ISPF.conf` suporte apenas a chamada de um exec de alocação, não há limites para um exec chamar outro exec. E o ID do usuário do cliente que está sendo transmitido como parâmetro abre os execs de alocação personalizados. Você pode, por exemplo, verificar se o membro `USERID'.EXEC(ALLOC)'` existe e executá-lo.

Uma variação elaborada para esse tema permite o uso de procedimentos de logon existentes do TSO, como a seguir:

- Leia um arquivo de configuração específico do usuário, como `USERID'.FEKPROF'`.
- Consulte qual procedimento de logon é mencionado no arquivo.
- Leia o procedimento mencionado a partir de `SYS1.PROCLIB` e analise-o para localizar as instruções DD e as alocações de conjunto de dados contidas.
- Aloque o conjunto de dados de uma forma semelhante ao procedimento de logon real.

## Avançado – Vários Arquivos ISPF.conf com várias Configurações do Developer para System z

Se os cenários do exec de alocação descritos acima não puderem tratar de suas necessidades específicas, você poderá criar diferentes instâncias do servidor de comunicação RSE do Developer para System z, cada uma utilizando seu próprio arquivo `ISPF.conf`. A desvantagem principal do método descrito a seguir é que os usuários do Developer para System z devem conectar-se a servidores diferentes no mesmo host para obter o ambiente TSO desejado.

**Nota:** A criação de uma segunda instância do servidor RSE requer apenas a duplicação e atualização de arquivos de configuração, JCL de inicialização e definições de tarefa iniciada. Não é necessária uma nova instalação do produto e nenhum código é duplicado.

```
$ cd /etc/rdz
$ mkdir /etc/rdz/tso2
$ cp rsed.envvars /etc/rdz/tso2
$ cp ISPF.conf /etc/rdz/tso2
$ ls /etc/rdz/tso2
ISPF.conf          rsed.envvars
$ oedit /etc/rdz/tso2/rsed.envvars
-> change: _CMDSERV_CONF_HOME=/etc/rdz/tso2
-> uncomment and change: -Ddaemon.log=/var/rdz/logs/tso2
-> add at the END:
# -- NEEDED TO FIND THE REMAINING CONFIGURATION FILES
CFG_BASE=/etc/rdz
CLASSPATH=.:$CFG_BASE:$CLASSPATH
# --
$ oedit /etc/rdz/tso2/ISPF.conf
-> change: change as needed
```

Os comandos no exemplo anterior copiam os arquivos de configuração do Developer para System z que exigem alterações no diretório tso2 recém-criado. A variável `_CMDSERV_CONF_HOME` em `rsed.envvars` deve ser atualizada para definir um novo diretório inicial `ISPF.conf` e o `daemon.log` deve ser atualizado para definir um novo local de log (que é criado automaticamente se ele não existir). A atualização de `CLASSPATH` assegura que o RSE possa localizar os arquivos de configuração que não foram copiados para tso2. O arquivo `ISPF.conf` em si pode ser atualizado para atender às suas necessidades. Observe que a área de trabalho do ISPF (variável `_CMDSERV_WORK_HOME` em `rsed.envvars`) pode ser compartilhada entre as duas instâncias.

Agora resta apenas criar uma nova tarefa iniciada para o RSE que utiliza um novo número de porta e os novos arquivos de configuração `/etc/rdz/tso2`.

Consulte as seções relacionadas nesta publicação para obter informações adicionais sobre as ações mostradas acima.

---

## Usando o Método de Acesso do APPC

### Customização Básica – JCL de Transação APPC

A definição de uma transação APPC consiste em parâmetros APPC e um JCL de transação. A JCL de amostra para criar uma transação APPC do Developer para System z, `FEK.#CUST.JCL(FEKAPPCC)`, contém duas opções para definir a JCL da transação, com e sem suporte ISPF.

```
//SYSIN DD DDNAME=SYSINISP * use SYSINTSO or SYSINISP
```

O cliente obtém o ambiente TSO/ISPF definido na JCL de transação, portanto, customizando esta seção seguindo regras DD regulares, você pode customizar o ambiente para o cliente.

```
...
//CMDSERV EXEC PGM=IKJEFT01,DYNAMNBR=50,
// PARM='ISPSTART CMD(%FEKFRSRV TIMEOUT=60) NEWAPPL(ISR) NESTMACS'
//SYSPROC DD DISP=SHR,DSN=FEK.SFEKPROC
//ISPLIB DD DISP=SHR,DSN=ISP.SISPPENU
//ISPLIB DD DISP=SHR,DSN=ISP.SISPMENU
//ISPTLIB DD DISP=SHR,DSN=ISP.SISPTENU
//ISPSLIB DD DISP=SHR,DSN=ISP.SISPSENU
```

```
//ISPPROF DD DISP=(NEW,DELETE,DELETE),UNIT=SYSALLDA,
//          SPACE=(TRK,(1,1,5)),LRECL=80,RECFM=FB
//SYSTSPRT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSTSIN DD DUMMY
//MYDD DD DISP=SHR,DSN=HLQ1.LLQ1
//          DISP=SHR,DSN=HLQ2.LLQ2
```

**Nota:** Uma transação APPC existente pode ser modificada utilizando painéis ISPF APPC.

## Avançado – Usar Perfis do ISPF Existentes

Se o suporte do ISPF for selecionado, o Developer para System z criará por padrão um perfil temporário do ISPF para o serviço TSO Commands. Entretanto, você pode instruir o Developer para System z a utilizar uma cópia de um perfil existente do ISPF. Conforme descrito na tarefa de amostra FEK.SFEKSAMP(FEKAPPC), você deve fazer o seguinte:

- Remover o comentário da etapa COPY na JCL de transação (EXEC e placas DD relacionadas).
- Alterar o &SYSUID..ISPPROF para corresponder ao nome do conjunto de dados do perfil do ISPF do usuário.
- Assinalar como comentário a primeira instrução DD ISPPROF na etapa CMDSERV e remover o comentário da segunda.

```
...
//COPY EXEC PGM=IEBCOPY * (optional) clone existing ISPF profile
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DISP=SHR,DSN=&SYSUID..ISPPROF//SYSUT2 DD DISP=(MOD,PASS),DSN=&&PROF,
//          UNIT=SYSALLDA,
//          LIKE=&SYSUID..ISPPROF
//*
//CMDSERV EXEC PGM=IKJEFT01,DYNAMNBR=50,
// PARM='ISPSTART CMD(%FEKFRSRV TIMEOUT=60) NEWAPPL(ISR) NESTMACS'
//*ISPPROF DD DISP=(NEW,DELETE,DELETE),UNIT=SYSALLDA,
//*          SPACE=(TRK,(1,1,5)),LRECL=80,RECFM=FB
//ISPPROF DD DISP=(OLD,DELETE,DELETE),DSN=&&PROF
```

**Nota:** Se for usado um nome de conjunto de dados inválido, a inicialização da transação APPC (e, portanto, o serviço TSO Commands) falhará.

## Avançado – Usando um Exec de Alocação

A transação da amostra JCL chama o serviço do TSO Commands diretamente ao passar seu nome (FEKFRSRV) como parâmetro para programar IKJEFT01. É possível alterar isto para chamar um outro executável. Este executável pode fazer alocações baseadas no ID do usuário atual e, em seguida, chamar o serviço do TSO Commands.

Ao contrário do método de acesso do TSO/ISPF Client Gateway, as variáveis armazenadas no perfil do ISPF do usuário podem ser usadas por esse exec para ajudá-lo a customizar o ambiente. Mas lembre-se de que as atualizações no perfil serão perdidas no final da sessão, pois você está utilizando uma cópia temporária, não o perfil real.

Observe, no entanto, que o uso de um exec de alocação na transação APPC não é suportado e a descrição acima está no estado em que se encontra.

## Avançado – Várias Transações APPC com Várias Configurações do Developer para System z

Se for necessário vários ambientes TSO exclusivos, você pode criar instâncias diferentes do servidor de comunicação RSE do Developer para System z', cada uma utilizando sua própria transação APPC. A desvantagem principal do método descrito a seguir é que os usuários do Developer para System z devem conectar-se a servidores diferentes no mesmo host para obter o ambiente TSO desejado.

**Nota:** A criação de uma segunda instância do servidor RSE requer apenas a duplicação e atualização de arquivos de configuração, JCL de inicialização e definições de tarefa iniciada. Não é necessária uma nova instalação do produto e nenhum código é duplicado.

```
$ cd /etc/rdz
$ mkdir /etc/rdz/tso2
$ cp rsed.envvars /etc/rdz/tso2
$ ls /etc/rdz/tso2/
rsed.envvars
$ oedit /etc/rdz/tso2/rsed.envvars
-> uncomment and change: _FEKFSCMD_TP_NAME_=FEKFTS02
-> uncomment and change: -Ddaemon.log=/var/rdz/logs/tso2
-> add at the END:
# -- NEEDED TO FIND THE REMAINING CONFIGURATION FILES
CFG_BASE=/etc/rdz
CLASSPATH=.:$CFG_BASE:$CLASSPATH
# --
```

Os comandos acima criam um novo diretório tso2 e copiam os arquivos de configuração do Developer para System z que precisam de mudanças para o novo local. A variável `_FEKFSCMD_TP_NAME_` em `rsed.envvars` deve ser atualizada para definir o novo nome da transação APPC, e o `daemon.log` deve ser atualizado para definir um novo local de log do daemon (que é criado automaticamente se ele ainda não existir). A atualização de `CLASSPATH` assegura que o RSE possa localizar os arquivos de configuração que não foram copiados para tso2.

```

//FEKAPPCC JOB CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1),NOTIFY=&SYSUID
//*
//TPADD EXEC PGM=ATBSDFMU
//SYSSDLIB DD DISP=SHR,DSN=SYS1.APPCTP
//SYSSDOUT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSIN DD DDNAME=SYSINISP * use SYSINTSO or SYSINISP
//SYSINISP DD DATA,DLM='QT'
TPADD
TPNAME(FEKFTS02)
ACTIVE(YES)
TPSCHED_DELIMITER(DLM1)
KEEP_MESSAGE_LOG(ERROR)
MESSAGE_DATA_SET(&SYSUID..FEKFTS02.&TPDATE..&TPTIME..LOG)
DATASET_STATUS(MOD)
CLASS(A)
JCL_DELIMITER(DLM2)
//FEKFTS02 JOB CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1)
//*
//CMDSERV EXEC PGM=IKJEFT01,DYNAMNBR=50,
// PARM='ISPSTART CMD(%FEKFRSRV TIMEOUT=60) NEWAPPL(ISR) NESTMACS'
//SYSPROC DD DISP=SHR,DSN=FEK.SFEKPROC
//ISPPLIB DD DISP=SHR,DSN=ISP.SISPPENU
//ISPMLIB DD DISP=SHR,DSN=ISP.SISPMENU
//ISPTLIB DD DISP=SHR,DSN=ISP.SISPTENU
//ISPSLIB DD DISP=SHR,DSN=ISP.SISPSENU
//ISPPROF DD DISP=(NEW,DELETE,DELETE),UNIT=SYSALLDA,
// SPACE=(TRK,(1,1,5)),LRECL=80,RECFM=FB
//SYSTSPRT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSTSIN DD DUMMY
//MYDD DD DISP=SHR,DSN=HLQ1.LLQ1
// DISP=SHR,DSN=HLQ2.LLQ2
DLM2
DLM1
QT

```

Figura 60. FEKAPPCC - criar uma segunda transação APPC

Em seguida, crie uma nova transação APPC customizando e enviando a tarefa de amostra FEK.#CUST.JCL(FEKAPPCC), conforme exibido na amostra acima. Além da customização normal (descrita na JCL), você também deve alterar o TPNAME para TPNAME(FEKFTS02) para corresponder à definição \_FEKFSCMD\_TP\_NAME\_ no novo rsed.envvars. Você deve alterar também o nome na variável MESSAGE\_DATA\_SET e o nome do JOB da JCL da transação.

Agora resta apenas criar uma nova tarefa iniciada para o RSE que utiliza um novo número de porta e os novos arquivos de configuração /etc/rdz/tso2.

Consulte as seções relacionadas nesta publicação para obter informações adicionais sobre as ações mostradas acima.





---

## Capítulo 17. Executando Várias Instâncias

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

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

### Nota:

- FEK e /usr/lpp/rdz são o qualificador e o caminho de alto nível usados durante a instalação do produto. FEK.#CUST, /etc/rdz e /var/rdz são os locais padrão usados durante a customização do produto (consulte “Configuração da Customização” na página 15 para obter informações adicionais).
- Você deve instalar o Developer para System z em um sistema de arquivos privado (HFS ou zFS) para facilitar a implementação das partes do produto z/OS UNIX.
- Se você não puder usar um sistema de arquivos privado, deverá usar uma ferramenta de arquivamento, como o comando z/OS UNIX tar para transportar os diretórios do z/OS UNIX de sistema para sistema. Isso para preservar os atributos (como controle de programas) para os arquivos e diretórios do Developer para System z.  
Consulte *UNIX System Services Command Reference* (SA22-7802) para obter informações adicionais sobre os seguintes comandos de amostra para arquivar e restaurar o diretório de instalação do Developer para System z.
  - Archive: `cd /SYS1/usr/lpp/rdz; tar -cSf /u/userid/rdz.tar`
  - Restore: `cd /SYS2/usr/lpp/rdz; tar -xSf /u/userid/rdz.tar`

---

## Configuração Idêntica em um Sysplex

Os arquivos de configuração do Developer para System z (e código) podem ser compartilhados através de sistemas diferentes em um sysplex, com cada sistema executando a sua própria cópia idêntica do Developer para System z, se algumas recomendações forem obedecidas.

- Os arquivos de log devem terminam em locais únicos para impedir que um sistema sobrescreva as informações de outro. Roteando os logs do z/OS UNIX para locais específicos com as diretivas `daemon.log` e `user.log` em `rsed.envvars`, é possível compartilhar os arquivos de configuração se você montar um sistema de arquivos z/OS UNIX específico no caminho especificado. Dessa maneira, todos os logs são gravados no mesmo local lógico, mas devido ao sistema de arquivos não compartilhado abaixo, eles terminam em locais físicos diferentes.

- Diretórios do tipo configuração como /etc/rdz/ e /var/rdz/projects/ podem ser compartilhados através do sysplex, uma vez que o Developer para System z os usa no modo somente leitura.
- Diretórios de dados temporários como /tmp/ e /var/rdz/WORKAREA/ devem ser exclusivos por sistema, uma vez que os nomes de arquivos temporários não seguem as especificações do sysplex.
- Se você compartilhar o código, deverá compartilhar também os arquivos de configuração para garantir que não tenha alguns sistemas fora de sincronização depois de aplicar manutenção.

## 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:** A configuração compartilhada NÃO PODE ser usada com segurança para manutenção de teste, visualização técnica ou novo release.

Para configurar outra instância de uma instalação ativa do Developer para System z, refaça as etapas de customização das partes que são diferentes, usando conjuntos de dados, diretórios e portas diferentes para evitar sobreposição da configuração atual.

Na amostra SSL mencionada acima, a configuração do daemon RSE atual pode ser fechada e depois a configuração clonada pode ser atualizada. Em seguida, a JCL de inicialização do daemon RSE pode ser clonada e customizada com uma nova porta TCP/IP e o local dos arquivos de configuração atualizado. As customizações MVS (JES Job Monitor, entre outras) podem ser compartilhadas entre instâncias SSL e não-SSL. Isso resultaria nas seguintes ações:

```
$ cd /etc/rdz
$ mkdir /etc/rdz/ssl
$ cp rsed.envvars /etc/rdz/ssl
$ cp ssl.properties /etc/rdz/ssl
$ ls /etc/rdz/ssl/
rsed.envvars    ssl.properties
$ oedit /etc/rdz/ssl/rsed.envvars
-> uncomment and change: -Daemon.log=/var/rdz/logs/ssl
-> add at the END:
# -- NEEDED TO FIND THE REMAINING CONFIGURATION FILES
CFG_BASE=/etc/rdz
CLASSPATH=.:$CFG_BASE:$CLASSPATH
# --
$ oedit /etc/rdz/ssl/ssl.properties
-> change: change as needed
```

Os comandos acima copiam os arquivos de configuração do Developer para System z que exigem alterações para um diretório ssl recém-criado. A variável daemon.log em rsed.envvars deve ser atualizada para definir um novo local de log (que é criado automaticamente se ele ainda não existir). A atualização de CLASSPATH assegura que o RSE possa localizar os arquivos de configuração que não foram copiados para ssl. O próprio arquivo ssl.properties pode ser atualizado para corresponder às suas necessidades.

Agora resta criar uma nova tarefa iniciada para o RSE que usa um novo número de porta e os novos arquivos de configuração /etc/rdz/ssl.

Consulte as seções relacionadas nesta publicação para obter informações adicionais sobre as ações mostradas acima.

---

## Todas as Outras Situações

Quando alterações de código estiverem envolvidas (manutenção, visualizações técnicas, novo release), ou suas alterações forem razoavelmente complexas, é aconselhável fazer outra instalação do Developer para 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 para System z:

- SMP/E CSI
- Bibliotecas de instalação
- Porta TCP/IP do JES Job Monitor e seu arquivo de configuração FEJJCNFG
- JCL de inicialização do JES Job Monitor
- Nome da transação APPC
- Arquivos de configuração RSE, rsed.envvars, \*.properties e \*.settings
- Porta TCP/IP do RSE
- JCL de inicialização do RSE

Uma visão geral mais detalhada é listada a seguir:

- SMP/E CSI
  1. Instale cada instância do Developer para System z em um CSI separado. O SMP/E evitará uma segunda instalação do mesmo FMID em um CSI, mas aceitará a instalação de outro FMID. Se o segundo FMID for de uma versão mais recente, ele excluirá a versão existente do produto. Se o segundo FMID for de uma versão mais antiga, a instalação falhará devido a nomes de partes duplicados.
- Bibliotecas de instalação
  1. Instale cada instância do Developer para System z em conjuntos de dados e diretórios separados. Lembre-se de que você só pode alterar o caminho do z/OS UNIX prefixando o padrão /usr/lpp/rdz fornecido pela IBM. Uma amostra válida seria /service/usr/lpp/rdz.
  2. A tarefa de configuração de customização FEK.SFEKSAMP(FEKSETUP) cria os conjuntos de dados e diretórios usados para armazenar arquivos de configuração. Como os arquivos de configuração devem ser exclusivos, e para evitar sobrescrever customizações existentes, você deve utilizar nomes de conjuntos de dados e de diretórios exclusivos ao enviar essa tarefa.
- Partes obrigatórias
  1. O arquivo de configuração do JES Job Monitor FEK.#CUST.PARMLIB(FEJJCNFG) contém o número de porta TCP/IP do JES Job Monitor e, portanto, não pode ser compartilhado. O membro pode ser renomeado (se a JCL também for atualizada), 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. A JCL de inicialização do JES Job Monitor FEK.#CUST.PROCLIB(JMON) refere-se a FEJJCNFG e, portanto, também não pode ser compartilhada. Depois de renomear o membro (e o cartão JOB, se você iniciá-lo como uma tarefa do usuário), todas as JCL's poderão ser colocadas no mesmo conjunto de dados.
  3. O arquivo de configuração RSE /etc/rdz/rsed.envvars contém referências para o caminho de instalação e, opcionalmente, para o local do log do

servidor, que requer que ela seja exclusivo. O nome do arquivo é obrigatório, portanto você não pode manter as cópias diferentes no mesmo diretório.

4. O arquivo de configuração ISPF.conf tem uma referência a FEK.SFEKPROC(FEKFRSRV), o servidor TSO Commands. Isto é específico do nível do software, portanto, você deve criar um arquivo ISPF.conf por instância.
  5. Todos os outros arquivos de configuração baseados em z/OS UNIX (como \*.properties) devem residir no mesmo diretório que o rsed.envvars e, portanto, não podem ser compartilhados, uma vez que o rsed.envvars deve estar em um local não compartilhado.
  6. A JCL de inicialização do RSE FEK.#CUST.PROCLIB(RSED) não pode ser compartilhada, já que ela define o número da porta TCP/IP e tem uma referência para os diretórios de instalação e configuração, que devem ser exclusivos. Depois de renomear o membro (e o cartão JOB, se você iniciá-lo como uma tarefa do usuário), todas as JCLs poderão ser colocadas no mesmo conjunto de dados.
  7. A JCL de inicialização do daemon de bloqueio FEK.#CUST.PROCLIB(LOCKD) não pode ser compartilhada, uma vez que possui uma referência para os diretórios de instalação e configuração, que devem ser exclusivos. Depois de renomear o membro (e o cartão de tarefa, se você iniciá-lo como uma tarefa do usuário) poderá colocar todas as JCLs no mesmo conjunto de dados.
- Partes opcionais
    1. As portas TCP/IP do REXEC e do SSH podem ser compartilhadas sem qualquer restrição.
    2. A transação APPC tem uma referência a FEK.SFEKPROC(FEKFRSRV), o servidor TSO Commands. Isto é específico ao nível do software, portanto, você deve criar uma transação APPC por instância. Lembre-se de que, como o nome da transação APPC é alterado, a variável \_FEKFSCMD\_TP\_NAME\_ deve ser definida em rsed.envvars.
    3. Alguns procedimentos ELAXF\* possuem uma referência a hlq.SFEKLOAD, a biblioteca de carregamento do Developer para System z. Consulte a nota na JCLLIB em "Procedimentos de Construção Remota ELAXF\*" na página 25 para uma possível solução ao disponibilizar conjuntos diferentes aos usuários.
    4. Para ativar duas instâncias do procedimento armazenado do DB2, as tarefas a seguir devem ser realizadas. No entanto, observe que essa é uma descrição no estado em que se encontra que não é suportada:
      - 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) Procedimento Armazenado do DB2” na página 87, mas com os novos membros.
  - e. O novo nome do ambiente WLM (por exemplo, ELAXMWDZ) deve ser usado no assistente do procedimento armazenado DB2 no cliente.
5. O suporte bidirecional em regiões do CICS depende de um membro da biblioteca de carregamento e, assim, não pode ser compartilhado entre os 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 nome do módulo de carregamento for alterado.
  6. Os módulos de carregamento do Application Deployment Manager que são incluídos nas regiões do CICS têm compatibilidade com versões anteriores e, portanto, a versão mais recente poderá ser compartilhada entre os releases.
  7. O Application Deployment Manager CRD VSAM é compatível com versões anteriores e, portanto, a versão mais recente pode ser compartilhada entre os releases.
  8. As definições de recursos CICS do Application Deployment Manager são compatíveis com versões anteriores e, portanto, a versão mais recente pode ser compartilhada entre os releases.
  9. Os CARMA VSAMs poderiam ser alterados entre os níveis de software, portanto, não é aconselhável compartilhá-los.



---

## Capítulo 18. Guia de Migração

---

### Considerações de Migração

Esta seção destaca as alterações na instalação e na configuração em comparação a releases anteriores do produto. Também fornece algumas diretrizes gerais para migrar para este release. Consulte as seções relacionadas neste manual para obter informações adicionais.

- Se você for um usuário anterior do IBM Rational Developer para System z, do IBM WebSphere Developer para System z, do IBM WebSphere Developer para zSeries ou do IBM WebSphere Studio Enterprise Developer, recomenda-se salvar os arquivos customizados relacionados ANTES de instalar o upgrade para esta versão do IBM Rational Developer para System z Versão 7.6.1.
- Leia o Capítulo 17, “Executando Várias Instâncias”, na página 271 se você planeja executar várias instâncias do Developer para System z.

**Nota:** As informações de migração listadas aqui são para as versões do Developer para System z ainda suportadas na hora da publicação.

### Fazendo Backup de Arquivos Configurados Anteriormente

Se você for um usuário anterior do Developer para System z, recomenda-se salvar os arquivos customizados relacionados antes de instalar esta versão do IBM Developer para System z.

Os arquivos customizáveis do Developer para system z podem ser localizados nos seguintes locais:

- Versão 7.5
  - FEK.SFEKSAMP, alguns membros são copiados para FEK.#CUST.\* pela tarefa de amostra FEKSETUP, em que \* equivale a PARMLIB, PROCLIB, JCL, CNTL, ASM e COBOL
  - FEK.SFEKSAMV
  - /usr/lpp/rdz/samples/, alguns arquivos são copiados para /etc/rdz/ e /etc/rdz/sclmdt/\* pela tarefa de amostra FEKSETUP, em que \* é igual a CONFIG/, CONFIG/PROJECT/ e CONFIG/script/
- Versão 7.1
  - FEK.SFEKSAMP
  - CRA.SCRASAMP
  - /usr/lpp/wd4z/rse/lib/, aconselha-se que os arquivos customizáveis sejam copiados para /etc/wd4z/
- Versão 7.0
  - FEK.SFEKSAMP
  - CRA.SCRASAMP
  - /usr/lpp/wd4z/rse/lib/, aconselha-se que os arquivos customizáveis sejam copiados para /etc/wd4z/

As configurações anteriores do Developer para system z também documentam as mudanças nos arquivos de configuração pertencentes por outros produtos.

- Versão 7.5
  - SYS1.PARMLIB(ASCHPMxx)

- definir uma classe de transação APPC para o serviço TSO Commands
  - SYS1.PARMLIB(BPXPRMxx)
    - configurar padrões do sistema z/OS UNIX
  - SYS1.PARMLIB(COMMNDxx)
    - iniciar servidores na hora do IPL
  - SYS1.PARMLIB(LPALSTxx)
    - incluir FEK.SFEKLPA no LPA
  - SYS1.PARMLIB(PROGxx)
    - FEK.SFEKAUTH autorizado por APF
    - incluir FEK.SFEKAUTH e FEK.SFEKLOAD em LINKLIST
  - (APPC)
    - definir uma transação APPC para o serviço TSO Commands
  - (WLM)
    - associar um programa de transação APPC a um grupo de desempenho do TSO
  - (WLM)
    - designar um ambiente de aplicativos para um procedimento armazenado do DB2
- Versão 7.1
  - SYS1.PARMLIB(ASCHPMxx)
    - definir uma classe de transação APPC para o serviço TSO Commands
  - SYS1.PARMLIB(BPXPRMxx)
    - configurar padrões do sistema z/OS UNIX
  - SYS1.PARMLIB(PROGxx)
    - FEK.SFEKLOAD autorizado por APF
  - /etc/services
    - definir a porta do daemon RSE
  - /etc/inetd.conf
    - definir o serviço do daemon RSE
  - /etc/SCLMDT/COMFIG/ISPF.conf
    - definir o local do servidor TSO Commands
  - (APPC)
    - definir uma transação APPC para o serviço TSO Commands
  - (WLM)
    - associar um programa de transação APPC a um grupo de desempenho do TSO
  - (WLM)
    - designar um ambiente de aplicativos para um procedimento armazenado do DB2
- Versão 7.0
  - SYS1.PARMLIB(ASCHPMxx)
    - definir uma classe de transação APPC para o serviço TSO Commands
  - SYS1.PARMLIB(BPXPRMxx)
    - configurar padrões do sistema z/OS UNIX
  - SYS1.PARMLIB(PROGxx)
    - FEK.SFEKLOAD autorizado por APF



- /etc/services  
definir a porta do daemon RSE
- /etc/inetd.conf  
definir o serviço do daemon RSE
- (APPC)  
definir uma transação APPC para o serviço TSO Commands
- (WLM)  
associar um programa de transação APPC a um grupo de desempenho do TSO
- (WLM)  
designar um ambiente de aplicativos para um procedimento armazenado do DB2

---

## Notas de Migração da Versão 7.6.1

As seguintes notas de migração são específicas para a versão 7.6.1. Elas são válidas para migração a partir da versão 7.6 ou são adições às notas de migração da versão existente 7.6.

- Application Deployment Manager - Os módulos ADN\* existentes na concatenação de RPL do CICS devem ser atualizados.
- Application Deployment Manager - os seguintes membros da amostra foram atualizados para incluir suporte URIMAP no utilitário administrativo:
  - ADNJSPAU
  - ADNVCRD
- Application Deployment Manager - um VSAM existente do repositório CRD deve ser substituído para ativar suporte URIMAP.
- CARMA - Suporte incluído para um layout de comprimento variável para o conjunto de dados VSAM de informações customizadas do CARMA, CRASTRS
- CARMA - Novos membros da amostra foram incluídos:
  - CRA#VS2 - migrar CRASTRS para formato de comprimento variável
- JES Job Monitor - Uso de \_CEE\_ENVFILE\_S no JCL da tarefa iniciada.
- JES Job Monitor - As seguintes diretivas FEJJCNGF tornaram-se opcionais :
  - HOST\_CODEPAGE
- PROCLIB - Novos membros do PROCLIB foram incluídos
  - ELAXFDCL
- RSE - Uso de Java 64 bits é suportado agora.
- RSE - Novos comandos do operador foram incluídos (desde a versão 7.6.1.0):
  - MODIFY DISPLAY PROCESS,DETAIL
- RSE - As seguintes diretivas não customizáveis mudaram ou são novas em rsed.envvars (desde versão 7.6.0.0):
  - (\_RSE\_JAVAOPTS) -DDSTORE\_KEEPALIVE\_RESPONSE\_TIMEOUT
  - (\_RSE\_JAVAOPTS) -DDSTORE\_IO\_SOCKET\_READ\_TIMEOUT
  - (\_RSE\_JAVAOPTS) -DRSECOMM\_LOGFILE\_MAX
- RSE - Novas diretivas opcionais foram incluídas no rsed.envvars (desde as versões 7.6.0.0 e 7.6.0.1):
  - (\_RSE\_JAVAOPTS) -Denable.automount
  - (\_RSE\_JAVAOPTS) -Ddeny.nozero.port
  - (\_RSE\_JAVAOPTS) -Dsingle.logon

- (\_RSE\_JAVAOPTS) -Dprocess.cleanup.interval
- RSE - As seguintes mensagens do console mudaram ou são novas (desde as versões 7.6.0.1 e 7.6.1.0):
  - FEK001I
  - FEK210I

---

## Migrar da Versão 7.5 para a Versão 7.6

### IBM Rational Developer para System z, FMID HHOP760

- O local de instalação SMP/E padrão dos componentes MVS and z/OS UNIX não foi alterado e, portanto, permanece FEK.\* e /usr/lpp/rdz/\*.
- Application Deployment Manager - Os módulos ADN\* existentes na concatenação de RPL do CICS devem ser atualizados.
- Application Deployment Manager - Novos módulos de carregamento, que devem fazer parte da concatenação RPL do CICS, foram incluídos para suportar a interface RESTful do CICS.
  - ADNANAL
  - ADNCRD41
  - ADNREST
- Application Deployment Manager - Novos membros de amostra foram incluídos para suportar a interface RESTful do CICS.
  - ADNCSDRS
  - ADNCSDTX
  - ADNTXNC
- Application Deployment Manager - Os membros de amostra existentes são renomeados.
  - ADNARCSD -> ADNCSDAR
  - ADNCMSGH -> ADNMSGHC
  - ADNMFEST -> ADNVMFST
  - ADNPCCSD -> ADNCSDWS
  - ADNSMSGH -> ADNMSGHS
  - ADNVSAM -> ADNVCRD
- Um novo tipo de produção RAM é fornecido para acessar CA Endevor® SCM.
  - CRARNDVR
- CARMA - Novos membros de amostra foram fornecidos para suportar o CA Endevor® SCM RAM.
  - FEK.#CUST.JCL(CRA#VCAD)
  - FEK.#CUST.JCL(CRA#VCAS)
  - FEK.#CUST.CNTL(CRASUBCA)
  - FEK.#CUST.PARMLIB(CRASHOW)
  - FEK.#CUST.PARMLIB(CRATMAP)
  - FEK.SFEKPROC(CRANDVRA)
  - /etc/rdz/crastart.endevor.conf
- CARMA - Novos membros de amostra foram fornecidos para suportar a mesclagem de definições da RAM.
  - CRA#UADD
  - CRA#UQRY

- File Manager Integration - A interface em lote para acessar o File Manager não é mais suportada.
- File Manager Integration - O arquivo de configuração FMIEXT.properties foi completamente alterado e deve ser substituído.
- JES Job Monitor - As opções LE são integradas no módulo de carregamento FEJMON (desde a versão 7.5.0.1), que pode exigir mudanças na definição da tarefa iniciada. Consulte a JCL de amostra FEK.SFEKSAMP(FEJJJCL) para obter mais detalhes.
- JES Job Monitor - Novas diretivas opcionais foram incluídas no FEJCNFG (na versão 7.5.0.1 e 7.5.1.0).
  - APPLID
  - CONSOLE\_NAME
  - GEN\_CONSOLE\_NAME
- JES Job Monitor - Um novo comando, Mostrar JCL, é suportado (desde a versão 7.5.1.0), podendo exigir atualizações ao software de segurança.
- Daemon de bloqueio – O daemon de bloqueio (LOCKD) é uma nova tarefa iniciada (desde a versão 7.5.0.1). Essa tarefa iniciada pode ser consultada para identificar qual cliente Developer para z está realizando um bloqueio de conjunto de dados. (Os comandos do sistema param no nível do espaço de endereço, que é o conjunto de encadeamentos do RSE.)
- SCLMDT - o local padrão dos arquivos de configuração do projeto SCLMDT foi alterado.
  - /var/rdz/sclmdt
- RSE - Novos comandos do operador foram incluídos.
  - MODIFY RSESTANDARDLOG
- RSE - Novas diretivas necessárias foram incluídas em rsed.envvars (na versão 7.5.0.1 e 7.6.0.0).
  - \_RSE\_LOCKD\_PORT
  - (\_RSE\_JAVAOPTS) -Dlock.daemon.port
  - (\_RSE\_JAVAOPTS) -Dlock.daemon.cleanup.interval
  - \_RSE\_LOCKD\_CLASS
  - \_RSE\_HOST\_CODEPAGE
  - (\_RSE\_JAVAOPTS) -Dfile.encoding
  - (\_RSE\_JAVAOPTS) -Dconsole.encoding
- RSE - Novas diretivas opcionais foram incluídas em rsed.envvars (desde as versões 7.5.0.1, 7.5.1.0 e 7.6.0.0).
  - (\_RSE\_JAVAOPTS) -Duser.log
  - (\_RSE\_JAVAOPTS) -Dkeep.last.log
  - (\_RSE\_JAVAOPTS) -Denable.standard.log
  - (\_RSE\_JAVAOPTS) -DDSTORE\_LOG\_DIRECTORY
  - (\_RSE\_JAVAOPTS) -DHIDE\_ZOS\_UNIX
  - (\_RSE\_JAVAOPTS) -Denable.certificate.mapping
  - GSK\_CRL\_SECURITY\_LEVEL
  - GSK\_LDAP\_SERVER
  - GSK\_LDAP\_PORT
  - GSK\_LDAP\_USER
  - GSK\_LDAP\_PASSWORD
- RSE - Algumas diretivas opcionais foram alteradas em rsed.envvars.

- (\_RSE\_JAVAOPTS) -Ddaemon.log
- (\_RSE\_JAVAOPTS) -Xmx
- SCLMDT\_CONF\_HOME
- RSE - Novas diretivas opcionais foram incluídas em `ssl.properties` (desde a versão 7.5.1.0 e 7.6.0.0).
  - `server_keystore_label`
  - `server_keystore_type`
- RSE - O daemon RSE suporta a autenticação por certificado de cliente X.509 (desde a versão 7.5.1.0), que requer atualizações no certificado atual e na configuração de segurança, quando usada.
- RSE - A segurança foi reforçada, causando falhas nos pedidos de conexão com erros `PassTicket` e `FEKAPPL`.
- RSE - O local padrão de todos os arquivos de log (logs de daemon e do usuário) foi alterado.
  - `/var/rdz/logs`
  - `/var/rdz/logs/$LOGNAME`
- RSE - Uma nova JCL de amostra foi fornecida para reunir informações de log e de configuração do Developer para System z.
  - `FEKLOGS`

## Arquivos Configuráveis

A Tabela 47 fornece uma visão geral de arquivos que foram customizados na versão 7.6. Observe que as bibliotecas de amostra do Developer para System z, `FEK.SFEKSAMP`, `FEK.SFEKSAMV` e `/usr/lpp/rdz/samples/`, são fornecidas com mais membros customizáveis que os listados aqui, como o código de origem de amostra do CARMA e as tarefas para compilá-las.

**Nota:** A tarefa de amostra `FEKSETUP` copia todos os membros listados em conjuntos de dados e diretórios diferentes, padrão `FEK.#CUST.*` e `/etc/rdz/*`.

Tabela 47. Customizações da Versão 7.6

Membro/Arquivo	Local padrão	Propósito	Notas de migração
FEKSETUP	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para criar conjuntos de dados e diretórios e preenchê-los com arquivos customizáveis	Atualizado para incluir novos membros customizáveis
JMON	FEK.SFEKSAMP(FEJJJCL) [FEK.#CUST.PROCLIB]	JCL para JES Job Monitor	Opção incluída para alterar opções LE
FEJJJCL	FEK.SFEKSAMP [FEK.#CUST.PROCLIB(JMON)]	Nome de remessa para o membro de JMON	Consulte o membro de JMON
RSED	FEK.SFEKSAMP(FEKRSED) [FEK.#CUST.PROCLIB]	JCL para daemon RSE	none
FEKRSED	FEK.SFEKSAMP [FEK.#CUST.PROCLIB(RSED)]	Nome de remessa para membro de RSED	Consulte o membro de RSED
LOCKD	FEK.SFEKSAMP(FEKLCKD) [FEK.#CUST.PROCLIB]	JCL para daemon de bloqueio	NOVO, customização é necessária
FEKLCKD	FEK.SFEKSAMP [FEK.#CUST.PROCLIB(LOCKD)]	Nome de remessa para o membro LOCKD	Consulte o membro LOCKD

Tabela 47. Customizações da Versão 7.6 (continuação)

Membro/Arquivo	Local padrão	Propósito	Notas de migração
ELAXF*	FEK.SFEKSAMP [FEK.#CUST.PROCLIB]	JCL para construções de projetos remotos, etc.	none
FEKRACF	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para definições de segurança	Atualizações menores
FEJJCNFG	FEK.SFEKSAMP [FEK.#CUST.PARMLIB]	Arquivo de configuração do JES Job Monitor	Novas diretivas opcionais foram incluídas
FEJTSO	FEK.SFEKSAMP [FEK.#CUST.CNTL]	JCL para emissões TSO	none
CRA\$VMSG	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para criação do VSAM de mensagens do CARMA	none
CRA\$VDEF	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para criação do VSAM de configuração do CARMA	none
CRA\$VSTR	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para criação do VSAM de informações customizadas do CARMA	none
CRASUBMT	FEK.SFEKSAMP [FEK.#CUST.CNTL]	CLIST de inicialização em lote do CARMA	none
CRASUBCA	FEK.SFEKSAMP [FEK.#CUST.CNTL]	CLIST de inicialização em lote do CARMA para CA Endevor® SCM RAM	NOVO, customização é opcional
CRASHOW	FEK.SFEKSAMP [FEK.#CUST.PARMLIB]	Configuração do CARMA para CA Endevor® SCM RAM	NOVO, customização é opcional
CRATMAP	FEK.SFEKSAMP [FEK.#CUST.PARMLIB]	Configuração do CARMA para CA Endevor® SCM RAM	NOVO, customização é opcional
CRANDVRA	FEK.SFEKPROC	REXX de alocação do CARMA para CA Endevor® SCM RAM	NOVO, customização é opcional
CRAISPRX	FEK.SFEKSAMP [FEK.#CUST.CNTL]	Exec de alocação de DD de amostra para CARMA utilizando TSO/ISPF Client Gateway	none
CRA#VSLM	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para criar VSAM de mensagem do SCLM RAM	none
CRA#ASLM	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para criar conjuntos de dados do SCLM RAM	none
CRA#VPDS	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para criar VSAM de mensagem do PDS RAM	none
CRA#CRAM	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para compilar RAM da estrutura	none
CRA#VCAD	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para a criação do VSAM de configuração do CARMA VSAM para CA Endevor® SCM RAM	NOVO, customização é opcional

Tabela 47. Customizações da Versão 7.6 (continuação)

Membro/Arquivo	Local padrão	Propósito	Notas de migração
CRA#VCAS	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para a criação do VSAM de informações customizadas do CARMA para CA Endevor® SCM RAM	NOVO, customização é opcional
CRA#UADD	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para mesclagem de definições da RAM	NOVO, customização é opcional
CRA#UQRY	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para extração de definições da RAM	NOVO, customização é opcional
CRAXJCL	FEK.SFEKSAMP [FEK.#CUST.ASM]	Código de origem de amostra para substituição de IRXJCL	none
CRA#CIRX	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para compilar CRAXJCL	none
ADNCSDRS	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para definir o servidor CRD RESTful como a região primária do CICS	NOVO, customização é opcional
ADNCSDTX	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para definir os IDs de transação alternativa como a região do CICS	NOVO, customização é opcional
ADNTXNC	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para criação de IDs de transação alternativa	NOVO, customização é opcional
ADNMSGHC	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para compilar ADNMSGHS	Renomeado, era ADNCMSGH
ADNMSGHS	FEK.SFEKSAMP [FEK.#CUST.COBOL]	Código de origem de amostra para o Manipulador de Mensagem de Pipeline	Renomeado, era ADNSMSGH
ADNVCRD	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para criar o repositório do CRD	Renomeado, era ADNVSAM
ADNCSDWS	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para definir o servidor CRD de Serviço da Web como a região primária do CICS	Renomeado, era ADNPCCSD
ADNCSDAR	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para definir o servidor CRD para regiões não-primárias do CICS	Renomeado, era ADNARCSD
ADNJSPAU	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para atualizar os padrões do CRD	Foram incluídas definições para o serviço RESTful, é necessário refazer as customizações
ADNVMFST	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para criar e definir o repositório do Manifesto	Renomeado, era ADNMFEST

Tabela 47. Customizações da Versão 7.6 (continuação)

Membro/Arquivo	Local padrão	Propósito	Notas de migração
ELAXMSAM	FEK.SFEKSAMP [FEK.#CUST.PROCLIB]	Procedimento JCL do espaço de endereços WLM para o Stored Procedure Builder para PL/I e COBOL	none
ELAXMJCL	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para definir Stored Procedure Builder para PL/I e COBOL para o DB2	none
FEKAPPCC	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para criação de uma transação APPC	none
FEKAPPCL	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para exibição de uma transação APPC	none
FEKAPPX	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para exclusão de uma transação APPC	none
FEKLOGS	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para coletar arquivos de log	NOVO, customização é opcional
rsed.envvars	/usr/lpp/rdz/samples/ [/etc/rdz/]	Variáveis de ambiente do RSE	Cópias mais antigas devem ser substituídas por esta (as customizações devem ser refeitas).
ISPF.conf	/usr/lpp/rdz/samples/ [/etc/rdz/]	arquivo de configuração do TSO/ISPF Client Gateway	ISPSISPCLIB incluído em SYSPROC para SCLMDT
CRASRV.properties	/usr/lpp/rdz/samples/ [/etc/rdz/]	Arquivo de configuração CARMA	none
crastart.conf	/usr/lpp/rdz/samples/ [/etc/rdz/]	Arquivo de configuração do CARMA para uso de CRASTART	none
crastart.endevor.conf	/usr/lpp/rdz/samples/ [/etc/rdz/]	Arquivo de configuração do CARMA para uso do CRASTART para CA Endevor® SCM RAM	NOVO, customização é opcional
ssl.properties	/usr/lpp/rdz/samples/ [/etc/rdz/]	Arquivo de configuração SSL RSE	Novas diretivas opcionais foram incluídas
rsecomm.properties	/usr/lpp/rdz/samples/ [/etc/rdz/]	Arquivo de configuração de rastreamento RSE	none
propertiescfg.properties	/usr/lpp/rdz/samples/ [/etc/rdz/]	Arquivo de configuração de grupos de propriedades baseados em host	none
projectcfg.properties	/usr/lpp/rdz/samples/ [/etc/rdz/]	Arquivo de configuração de projetos baseados em host	none



Tabela 47. Customizações da Versão 7.6 (continuação)

Membro/Arquivo	Local padrão	Propósito	Notas de migração
FMEXT.properties	/usr/lpp/rdz/samples/ [/etc/rdz/]	Arquivo de configuração de Integração do File Manager	Cópias mais antigas devem ser substituídas por esta (as customizações devem ser refeitas).
uchars.settings	/usr/lpp/rdz/samples/ [/etc/rdz/]	Arquivo de configuração de caracteres não editáveis	none

## Migrar da Versão 7.1 para a Versão 7.5

### IBM Rational Developer para System z, FMID HHOP750

- O local de instalação SMP/E padrão dos componentes MVS não foi alterado e, portanto, permanece FEK.\*.
- O local de instalação SMP/E padrão dos componentes z/OS UNIX foi alterado para /usr/lpp/rdz/\*.
- O Common Access Repository Manager (CARMA) foi integrado ao Developer para System z Versão 7.5, eliminando a necessidade de instalá-lo como um produto separado.
- O SCLM Developer Toolkit foi integrado ao Developer para System z Versão 7.5, eliminando a necessidade de instalá-lo como um produto separado.
- Na versão 7.5, o Gateway do Cliente TSO/ISPF do ISPF substitui a função SCLM Developer Toolkit usada na versão 7.1 para conectar-se ao serviço de Comandos TSO. O método de conexão APPC ainda é suportado.
- Na versão 7.5, o servidor RSE não é mais um processo gerenciado pelo INETD, mas uma tarefa iniciada. O servidor RSE agora utiliza também um único modelo de servidor, enquanto que, com versões anteriores, cada conexão cliente-host possuía um servidor RSE privado.
- Todos os módulos requerem a autorização do APF (JES Job Monitor e SCLM Developer Toolkit) movidos para FEK.SFEKAUTH na versão 7.5, que requer uma atualização para as definições existentes de APF.
- O módulo de carregamento do JES Job Monitor movido para o FEK.SFEKAUTH na versão 7.5, que requer uma atualização para o procedimento de tarefa iniciada existente.
- Os módulos de carregamento CARMA foram movidos para novas bibliotecas, exigindo uma atualização no script de inicialização do servidor CRASUBMT existente.
- Os módulos de carregamento do SCLM Developer Toolkit mudaram para novas bibliotecas, exigindo uma atualização nas definições LINKLIST existentes.
- O ELAXFTS0 é um novo procedimento de construção de amostra desde a versão 7.1.1, e ELAXFCP1 e ELAXFPP1 são novos na versão 7.5.
- O uchars.settings é um novo arquivo de configuração para os caracteres não-editáveis.
- O propertiescfg.properties é um novo arquivo de configuração para os grupos de propriedades padrão.



- FEJCNFG, CRASRV.properties e FMIEXT.properties possuem novas diretivas opcionais.
- O rsed.envvars foi alterado na versão 7.5 e deve ser substituído.
- O arquivo de amostra ISPF.conf fornecido com a versão 7.5 é semelhante ao usado pelo SCLM Developer Toolkit na versão 7.1.
- Algumas das customizações do Application Deployment Manager existentes devem ser refeitas.
- O Application Deployment Manager possui novas funções que requerem customização.
- As configurações de segurança para o servidor RSE foram alteradas drasticamente na versão 7.5.
- O perfil de segurança MVS.MCSOPER.JMON é novo para o JES Job Monitor na versão 7.5.
- O script de inicialização do CARMA mudou de nome e foi movido para um novo local, requerendo uma atualização no arquivo de configuração CRASRV.properties existente.
- O script de inicialização do FMI mudou de nome e foi movido para um novo local, requerendo uma atualização no arquivo de configuração FMIEXT.properties existente.
- Um novo módulo de carregamento foi incluído para suporte bidirecional, exigindo uma atualização na concatenação CICS DFHRPL existente se você não estiver usando a biblioteca FEK.SFEKLOAD.
- As alterações no parâmetro MAXPROCUSER de SYS1.PARMLIB(BPXPRMxx) agora também são documentadas.

## Arquivos Configuráveis

A Tabela 48 fornece uma visão geral de arquivos que foram customizados na versão 7.5. Observe que as bibliotecas de amostra do Developer para System z, FEK.SFEKSAMP, FEK.SFEKSAMV e /usr/lpp/rdz/samples/, são fornecidas com mais membros customizáveis que os listados aqui, como o código de origem do CARMA de amostra e as tarefas para compilá-las.

**Nota:** A tarefa de amostra FEKSETUP copia todos os membros listados em conjuntos de dados e diretórios diferentes, padrão FEK.#CUST.\* e /etc/rdz/\*.

Tabela 48. Customizações da Versão 7.5

Membro/Arquivo	Local padrão	Propósito	Notas de migração
FEKSETUP	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para criar conjuntos de dados e diretórios e preenchê-los com arquivos customizáveis	NOVO, customização é necessária
JMON	FEK.SFEKSAMP(FEJJJCL) [FEK.#CUST.PROCLIB]	JCL para JES Job Monitor	STEPLIB alterado para SFEKAUTH
RSED	FEK.SFEKSAMP(FEKRSED) [FEK.#CUST.PROCLIB]	JCL para daemon RSE	NOVO, customização é necessária
ELAXF*	FEK.SFEKSAMP [FEK.#CUST.PROCLIB]	JCL para construções de projetos remotos, etc.	ELAXFTSO, ELAXFCP1 e ELAXFPP1 são novos

Tabela 48. Customizações da Versão 7.5 (continuação)

Membro/Arquivo	Local padrão	Propósito	Notas de migração
FEKRACF	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para definições de segurança	NOVO, necessário
FEJJCNG	FEK.SFEKSAMP [FEK.#CUST.PARMLIB]	Arquivo de configuração do JES Job Monitor	<ul style="list-style-type: none"> <li>• Algumas diretivas se tornaram opcionais</li> <li>• Novas diretivas opcionais foram incluídas</li> </ul>
FEJTSO	FEK.SFEKSAMP [FEK.#CUST.CNTL]	JCL para emissões TSO	O nome da tarefa agora pode ser uma variável
CRAISPRX	FEK.SFEKSAMP [FEK.#CUST.CNTL]	Exec de alocação de DD de amostra para CARMA utilizando TSO/ISPF Client Gateway	NOVO, customização é opcional
CRAXJCL	FEK.SFEKSAMP [FEK.#CUST.ASM]	Código de origem de amostra para substituição de IRXJCL	NOVO, customização é opcional
CRA#CIRX	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para compilar CRAXJCL	NOVO, customização é opcional
ADNSMSGH	FEK.SFEKSAMP [FEK.#CUST.COBOLE]	Código de origem de amostra para o Manipulador de Mensagem de Pipeline	As cópias antigas devem ser substituídas por esta (as customizações devem ser refeitas)
ADNPCCSD	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para definir o servidor CRD para a região primária do CICS	As cópias antigas devem ser substituídas por esta (as customizações devem ser refeitas)
ADNJSPAU	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para atualizar os padrões do CRD	NOVO, customização é opcional
ADNMFEST	FEK.SFEKSAMP [FEK.#CUST.JCL]	JCL para criar e definir o repositório do Manifesto	NOVO, customização é opcional
rsed.envvars	/usr/lpp/rdz/samples/ [/etc/rdz/]	Variáveis de ambiente do RSE	As cópias antigas devem ser substituídas por esta (as customizações devem ser refeitas)

Tabela 48. Customizações da Versão 7.5 (continuação)

Membro/Arquivo	Local padrão	Propósito	Notas de migração
ISPF.conf	/usr/lpp/rdz/samples/ [/etc/rdz/]	arquivo de configuração do TSO/ISPF Client Gateway	Idêntico ao ISPF.conf fornecido com SCLMDT na v7.1
CRASRV.properties	/usr/lpp/rdz/samples/ [/etc/rdz/]	Arquivo de configuração CARMA	<ul style="list-style-type: none"> <li>Local e nome do script de inicialização alterados</li> <li>Novas diretivas opcionais foram incluídas</li> </ul>
crastart.conf	/usr/lpp/rdz/samples/ [/etc/rdz/]	Arquivo de configuração do CARMA para uso de CRASTART	NOVO, customização é opcional
FMEXT.properties	/usr/lpp/rdz/samples/ [/etc/rdz/]	Arquivo de configuração de Integração do File Manager	<ul style="list-style-type: none"> <li>Local e nome do script de inicialização alterados</li> <li>Novas diretivas opcionais foram incluídas</li> </ul>
uchars.settings	/usr/lpp/rdz/samples/ [/etc/rdz/]	Arquivo de configuração de caracteres não editáveis	NOVO, customização é opcional

## Migrar da Versão 7.0 para a Versão 7.1

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

- O local de instalação SMP/E padrão dos componentes MVS e z/OS UNIX não foi alterado e, portanto, permanece FEK.\* e /usr/lpp/wd4z/\*.
- **Incluído:** Opção de configuração - comandos TSO/ISPF através de uma transação APPC ou do SCLM Developer Toolkit
- **Alterado:** A transação APPC explora um novo recurso ISPF
- **Incluído:** Os seguintes membros customizáveis são novos:
  - samplib ELAXFADT
  - samplib ADNCMSGH
  - /usr/lpp/wd4z/rse/lib/FMEXT.properties
- **Alterado:** Os seguintes membros mudaram:
  - SFEKDLL(FEJBDTRX) -> SFEKLOAD(FEJBDTRX)
- **Alterado:** Os seguintes membros customizáveis mudaram:
  - 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

- O local de instalação SMP/E padrão dos componentes MVS não foi alterado e, portanto, permanece CRA.\*.
- **Alterado:** A criação de log foi gravada na instrução CARMALOG DD
- **Alterado:** O VSAM de mensagens CARMA (CRAMSG) e o VSAM de configuração (CRADEF) foram atualizados
- **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\$VMSG
  - samplib CRASREPR -> CRA\$VSTR
  - samplib CRASALX -> CRA#ASLM
  - samplib CRACOBJ1 -> CRA#CCB1
  - samplib CRACOBJ2 -> CRA#CCB2
  - samplib CRACLICM -> CRA#CCLT
  - samplib CRARAMCS -> CRA#CPDS
  - samplib CRARAMCM -> CRA#CRAM
  - samplib CRATREPR -> CRA#VPDS
  - samplib CRALREPR -> CRA#VSLM
  - samplib CRACLIRN -> CRA#XCLT
- **Alterado:** Os seguintes membros customizáveis mudaram:
  - clist CRASUBMT

## Arquivos Configuráveis

A tabela 23 fornece uma visão geral de arquivos que foram customizados na versão 7.1. Observe que as bibliotecas de amostra do CARMA e do Developer para System z, CRA.SCRASAMP, FEK.SFEKSAMP e /usr/lpp/wd4z/rse/lib/, são fornecidas com mais membros customizáveis que os listados aqui, como o código de origem do CARMA de amostra e as tarefas para compilá-las.

Tabela 49. Customizações da Versão 7.1

Membro/Arquivo	Local padrão	Propósito	Notas de migração
ELAXF*	FEK.SFEKSAMP	JCL para construções de projetos remotos, e outras tarefas	ELAXFADT é novo
CRA\$VMSG	CRA.SCRASAMP	JCL para criação do VSAM de mensagens do CARMA	<ul style="list-style-type: none"> <li>• renomeado, era CRAMREPR</li> <li>• o VSAM criado por essa tarefa foi atualizado</li> </ul>

Tabela 49. Customizações da Versão 7.1 (continuação)

Membro/Arquivo	Local padrão	Propósito	Notas de migração
CRA\$VDEF	CRA.SCRASAMP	JCL para criação do VSAM de configuração do CARMA	<ul style="list-style-type: none"> <li>renomeado, era CRAREPR</li> <li>o VSAM criado por essa tarefa foi atualizado</li> </ul>
CRA\$VSTR	CRA.SCRASAMP	JCL para criação do VSAM de informações customizadas do CARMA	renomeado, era CRASREPR
CRASUBMT	CRA.SCRASAMP	CLIST de inicialização em lote do CARMA	incluir DD CARMALOG
CRA#VSLM	CRA.SCRASAMP	JCL para criar VSAM de mensagem do SCLM RAM	renomeado, era CRALREPR
CRA#ASLM	CRA.SCRASAMP	JCL para criar conjuntos de dados do SCLM RAM	renomeado, era CRASALX
CRA#VPDS	CRA.SCRASAMP	JCL para criar VSAM de mensagem do PDS RAM	renomeado, era CRATREPR
CRA#CRAM	CRA.SCRASAMP	JCL para compilar RAM da estrutura	renomeado, era CRARAMCM
FEKAPPCC	FEK.SFEKSAMP	JCL para criação de uma transação APPC	explorar o suporte ISPF NEST
r sed.envvars	/usr/lpp/wd4z/rse/lib/ [/etc/wd4z/]	Variáveis de ambiente do RSE	As cópias antigas devem ser substituídas por esta (as customizações devem ser refeitas)
FMEXT.properties	/usr/lpp/wd4z/rse/lib/ [/etc/wd4z/]	Arquivo de configuração de Integração do File Manager	NOVO, customização é necessária, quando usada



---

## Apêndice A. Configurando o SSL e a Autenticação X.509

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. Este apêndice fornece também uma configuração de amostra para suportar usuários que se autenticam com um certificado X.509.

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. O SSL fornece 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 *Security Server RACF Security Administrator's Guide* (SA22-7683) para obter informações adicionais sobre certificados digitais e o protocolo de chave pública usados pelo SSL.

As ações necessárias para configurar as comunicações SSL para o Developer para System z variam de site para site, dependendo das necessidades exatas, do método de comunicação RSE usado e do que já está disponível no site.

Neste apêndice, clonaremos as definições de RSE atuais para que tenhamos uma 2ª conexão do daemon RSE que utilizará SSL. Também criaremos nossos próprios certificados de segurança a serem usados pelas diferentes partes da conexão RSE.

- “Decidir Onde Armazenar Chaves Privadas e Certificados”
- “Criar um Anel de Chave com o RACF” na página 295
- “Clonar a Configuração RSE Existente” na página 296
- “Atualizar rsed.envvars para Ativar a Coexistência” na página 297
- “Atualizar ssl.properties para Ativar SSL” na página 297
- “Ativar SSL Criando um Novo Daemon RSE” na página 297
- “Testar a Conexão” na página 298
- “(Opcional) Incluir Suporte de Autenticação de Cliente X.509” na página 301
- “(Opcional) Criar um Banco de Dados de Chaves com gskkyman” na página 301
- “(Opcional) Criar um Keystore com keytool” na página 304

Todo este apêndice apresenta uma convenção de nomenclatura uniforme:

- Certificado: rdzrse
- Armazenamento de chaves e certificados: rdzssl.\*
- Senha: rsessl
- ID de usuário do Daemon: stcrse

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

---

### Decidir Onde Armazenar Chaves Privadas e Certificados

Os certificados de identidade e as chaves de criptografia/descriptografia usadas pelo SSL são armazenados em um arquivo de chaves. Existem diferentes implementações deste arquivo de chaves, dependendo do tipo de aplicativo.

No entanto, todas as implementações seguem o mesmo princípio. Um comando gera um par de chaves (uma chave pública e uma chave privada associada). O comando agrupa então a chave pública em um certificado X.509 auto-assinado, que é 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-chave.

O daemon RSE é um aplicativo SSL do Sistema e utiliza um arquivo de banco de dados de chaves. Esse banco de dados de chaves pode ser um arquivo físico criado por gskkyman ou um conjunto de chaves gerenciado pelo seu software de segurança compatível com SAF (por exemplo, RACF). O servidor RSE (que é iniciado pelo daemon) é um aplicativo Java SSL e usa um arquivo keystore criado por keytool ou um conjunto de chaves gerenciado pelo seu software de segurança.

*Tabela 50. Mecanismos de armazenamento de certificado SSL*

Armazenamento de certificado	Criado e gerenciado por	Daemon RSE	servidor RSE
conjunto de chaves	Produto de segurança compatível com SAF	suportados	suportados
banco de dados de chaves	gskkyman do z/OS UNIX	suportados	/
keystore	keytool Java	/	suportados

Para conectar por meio de SSL, é necessário o keystore e o banco de dados principal, como um arquivo z/OS UNIX ou um conjunto de chaves compatível com SAF:

- keystore (RACF ou keytool)
- banco de dados de chaves (RACF ou gskkyman)

**Nota:**

- Conjuntos de chaves compatíveis com SAF é o método preferido para gerenciar certificados.
- Um certificado compartilhado poderá ser usado se o daemon RSE e o servidor RSE usarem o mesmo método de gerenciamento de certificado.
- O daemon RSE deve ser executado controlado pelo programa. O uso do SSL do Sistema implica que SYS1.SIEALNKE deve se tornar controlado pelo programa pelo software de segurança.
- 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.  
STEPLIB=\$STEPLIB:SYS1.SIEALNKE  
Porém, lembre-se de que:
  - A utilização de STEPLIB no z/OS UNIX tem um impacto de desempenho negativo.
  - Se uma biblioteca STEPLIB for autorizada pelo APF, todas deverão ser autorizadas. As bibliotecas perderão sua autorização do APF se forem combinadas com bibliotecas não-autorizadas pelo STEPLIB.
- O SSL do Sistema utiliza o ICSF (Integrated Cryptographic Service Facility), se estiver disponível. O ICSF oferece suporte criptográfico a hardware que será usado, em vez de algoritmos de software do SSL do Sistema. Consulte *System SSL Programming* (SC24-5901) para obter informações adicionais.



Consulte *Security Server RACF Security Administrator's Guide* (SA22-7683) para obter informações sobre RACF e caracteres digitais. A documentação de gskkyman pode ser localizada em *System SSL Programming* (SC24-5901) e a documentação de keytool está disponível em <http://java.sun.com/j2se/1.5.0/docs/tooldocs/solaris/keytool.html>.

---

## Criar um Anel de Chave com o RACF

Não execute esta etapa se utilizar o gskkyman para criar o banco de dados de chaves do daemon RSE e o keytool para criar o keystore do servidor RSE.

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(stcrse)
PERMIT IRR.DIGTCERT.LISTRING CLASS(FACILITY) ACCESS(READ) ID(stcrse)
SETROPTS RACLIST(FACILITY) REFRESH
```

```
RACDCERT ID(stcrse) GENCERT SUBJECTSDN(CN('rdz rse ssl') +
OU('rdz') O('IBM') L('Raleigh') SP('NC') C('US')) +
NOTAFTER(DATE(2017-05-21)) WITHLABEL('rdzrse') KEYUSAGE(HANDSHAKE)
```

```
RACDCERT ID(stcrse) ADDRING(rdzssl.racf)
RACDCERT ID(stcrse) CONNECT(LABEL('rdzrse') RING(rdzssl.racf) +
DEFAULT USAGE(PERSONAL))
```

A amostra acima é iniciada criando-se os perfis necessários e permitindo que o ID de usuário STCRSE acesse os conjuntos de chaves e os certificados pertencentes por esse ID de usuário. O ID do usuário usado deve corresponder ao ID do usuário usado para executar o daemon do RSE SSL. A próxima etapa é criar um novo certificado auto-assinado com rótulo rdzrse. Não é necessária senha. Esse certificado é incluído em um anel de chaves recém-criado (rdzssl.racf). Assim como com o certificado, não é necessária senha para o anel de chave.

O resultado pode ser verificado com a seguinte opção list:

```
RACDCERT ID(stcrse) LIST
```

Informações do certificado digital para o usuário STCRSE:

```
Rótulo: rdzrse
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=rdz rse ssl.OU=rdz.O=IBM.L=Raleigh.SP=NC.C=US<
Nome do Assunto:
>CN=rdz rse ssl.OU=rdz.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: STCRSE
Anel:
>rdzssl.racf<
```

## (Opcional) Usando um Certificado Assinado

Os certificados podem ser auto-assinados ou assinados por uma Autoridade de Certificação (CA). Um certificado assinado por uma CA significa que a CA garante que o proprietário do certificado é quem afirma ser. O processo de assinatura inclui as credenciais de CA (também um certificado) no certificado, tornando-o uma cadeia de certificados com vários elementos.

Ao usar um certificado assinado por uma CA, você pode evitar perguntas sobre validação confiável pelo cliente do Developer para System z, se o cliente já confiar na CA.

Siga estas etapas para criar e utilizar um certificado assinado pela CA:

1. Crie um certificado autoassinado.  
`RACDCERT ID(stcrse) GENCERT WITHLABEL('rdzrse') . . .`
  2. Crie um pedido de assinatura para este certificado.  
`RACDCERT ID(stcrse) GENREQ (LABEL('rdzrse')) DSN(dsn)`
  3. Envie o pedido de assinatura para a CA de sua escolha.
  4. Verifique se as credenciais de CA (também um certificado) já são conhecidas.  
`RACDCERT CERTAUTH LIST`
  5. Marque o certificado de CA como confiável.  
`RACDCERT CERTAUTH ALTER(LABEL('CA cert')) TRUST`  
Ou inclua o certificado de CA no banco de dados.  
`RACDCERT CERTAUTH ADD(dsn) WITHLABEL('CA cert') TRUST`
  6. Inclua o certificado assinado no banco de dados; ele substituirá o auto-assinado.  
`RACDCERT ID(stcrse) ADD(dsn) WITHLABEL('rdzrse') TRUST`
- Nota:** NÃO exclua o certificado auto-assinado antes de substituí-lo. Se você fizer isso, perderá a chave privada fornecida com o certificado, inutilizando-o.
7. Crie um conjunto de chaves.  
`RACDCERT ID(stcrse) ADDRING(rdzssl.racf)`
  8. Inclua o certificado assinado no conjunto de chaves.  
`RACDCERT ID(stcrse) CONNECT(ID(stcrse) LABEL('rdzrse'))  
RING(rdzssl.racf))`
  9. Inclua o certificado de CA no conjunto de chaves.  
`RACDCERT ID(stcrse) CONNECT(CERTAUTH LABEL('CA cert'))  
RING(rdzssl.racf))`

---

## Clonar a Configuração RSE Existente

Nesta etapa, uma nova instância dos arquivos de configuração do RSE é criada, para que a configuração SSL possa ser executada paralelamente com a(s) existente(s). Os comandos de amostra a seguir esperam que os arquivos de configuração estejam em `/etc/rdz/`, que é o local padrão usado em “Configuração da Customização” na página 15.

```
$ cd /etc/rdz
$ mkdir ssl
$ cp rsed.envvars ssl
$ cp ssl.properties ssl
$ ls ssl
rsed.envvars    ssl.properties
```

Os comandos do z/OS UNIX listados acima criam um subdiretório chamado `ssl` e o preenchem com os arquivos de configuração que precisam de mudanças. Podemos compartilhar os outros arquivos de configuração, o diretório de instalação e os componentes do MVS, porque não são específicos do SSL.

Ao reutilizar a maioria dos arquivos de configuração existentes, podemos nos concentrar nas alterações realmente necessárias para configurar o SSL e evitar fazer a configuração completa do RSE novamente. (Por exemplo, podemos evitar a definição de um novo local para `ISPF.conf`.)

---

## Atualizar `rsed.envvars` para Ativar a Coexistência

Até agora, as definições são uma cópia exata da configuração atual, o que implica que os logs do novo daemon RSE sobreporão os arquivos de log atuais do servidor. O RSE também precisa saber onde localizar os arquivos de configuração que não foram copiados para o diretório `ssl`. Os dois problemas podem ser tratados por mudanças secundárias em `rsed.envvars`.

```
$ oedit /etc/rdz/ssl/rsed.envvars
-> uncomment and change: -Ddaemon.log=/var/rdz/logs/ssl
-> add at the END:
# -- NEEDED TO FIND THE REMAINING CONFIGURATION FILES
CFG_BASE=/etc/rdz
CLASSPATH=.:$CFG_BASE:$CLASSPATH
# --
```

As mudanças acima definem um novo local de log (que será criado pelo daemon RSE se o local do log não existir). As mudanças também atualizam o `CLASSPATH` para que os processos RSE do SSL procurem arquivos de configuração primeiramente no diretório atual (`/etc/rdz/ssl`) e, em seguida, procurem no diretório original (`/etc/rdz`).

---

## Atualizar `ssl.properties` para Ativar SSL

Atualizando `ssl.properties`, o RSE é orientado a iniciar o uso da comunicação criptografada por SSL.

```
$ oedit /etc/rdz/ssl/ssl.properties
-> change: enable_ssl=true
-> uncomment and change: daemon_keydb_file=rdzssl.racf
-> uncomment and change: daemon_key_label=rdzrse
-> uncomment and change: server_keystore_file=rdzssl.racf
-> uncomment and change: server_keystore_label=rdzrse
-> uncomment and change: server_keystore_type=JCERACFKS
```

As mudanças acima ativam o SSL e informam ao daemon RSE e ao servidor RSE que seus certificados (compartilhados) estão armazenados com o rótulo `rdzrse` no conjunto de chaves `rdzssl.racf`. A palavra-chave `JCERACFKS` informa ao servidor RSE que um conjunto de chaves compatível com SAF é usado como keystore.

---

## Ativar SSL Criando um Novo Daemon RSE

Conforme já mencionado, criaremos uma segunda conexão que utilizará SSL, o que significa a criação de um novo daemon do RSE. O daemon do RSE pode ser uma tarefa iniciada ou uma tarefa do usuário. O método de tarefa do usuário para configuração (teste) inicial será usado. As instruções a seguir esperam que a JCL de amostra esteja em `FEK.#CUST.PROCLIB(RSED)`, que é o local padrão usado em “Configuração da Customização” na página 15:

1. Crie um novo membro FEK.#CUST.PROCLIB(RSEDSSL) e copie na JCL de amostra FEK.#CUST.PROCLIB(RSED).
2. Customize RSEDSSL incluindo uma placa de tarefa na parte superior e uma instrução exec na parte inferior. Fornece também um novo número de porta (4047) e o local dos arquivos de configuração relacionados ao SSL (/etc/rdz/ssl), conforme mostrado na amostra de código a seguir. Observe que forçamos o uso do ID de usuário STCRSE, pois esse ID de usuário recebeu a autoridade de acesso apropriada para certificados e conjuntos de chaves em uma etapa anterior.

```
//RSEDSSL JOB CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1),USER=STCRSE
//*
//* RSE DAEMON - SSL
//*
//RSED      PROC IVP='',                                * 'IVP' to do an IVP test
//          PORT=4047,
//          HOME='/usr/lpp/rdz',
//          CNFG='/etc/rdz/ssl'
//*
//RSE       EXEC PGM=BPXBATSL,REGION=0M,TIME=NOLIMIT,
//          PARM='PGM &HOME./bin/rsed.sh &IVP &PORT &CNFG'
//STDOUT    DD SYSOUT=*
//STDERR    DD SYSOUT=*
//PEND
//*
//RSED      EXEC RSED
//*
```

Figura 61. RSEDSSL - Tarefa do usuário do daemon RSE para SSL

**Nota:** O ID de usuário designado à tarefa RSEDSSL deve ter as mesmas autorizações que o daemon RSE original. O perfil FACILITY BPX.SERVER e o perfil PTKTDATA IRRPTAUTH.FEKAPPL.\* são os elementos-chave aqui.

---

## Testar a Conexão

A configuração do host SSL é completa e o daemon RSE para SSL pode ser iniciado ao enviar a tarefa FEK.#CUST.PROCLIB(RSEDSSL) criada anteriormente.

A nova configuração pode agora ser testada conectando-se com o cliente Developer para System z. Como criamos uma nova configuração para ser usada pelo SSL (clonando uma existente), uma nova conexão deverá ser definida no cliente utilizando-se a porta 4047 para o daemon RSE.

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 Developer para System z não reconhecer o certificado do host ou a CA que o assinou, o cliente Developer para System z perguntará ao usuário se é possível confiar nesse certificado.

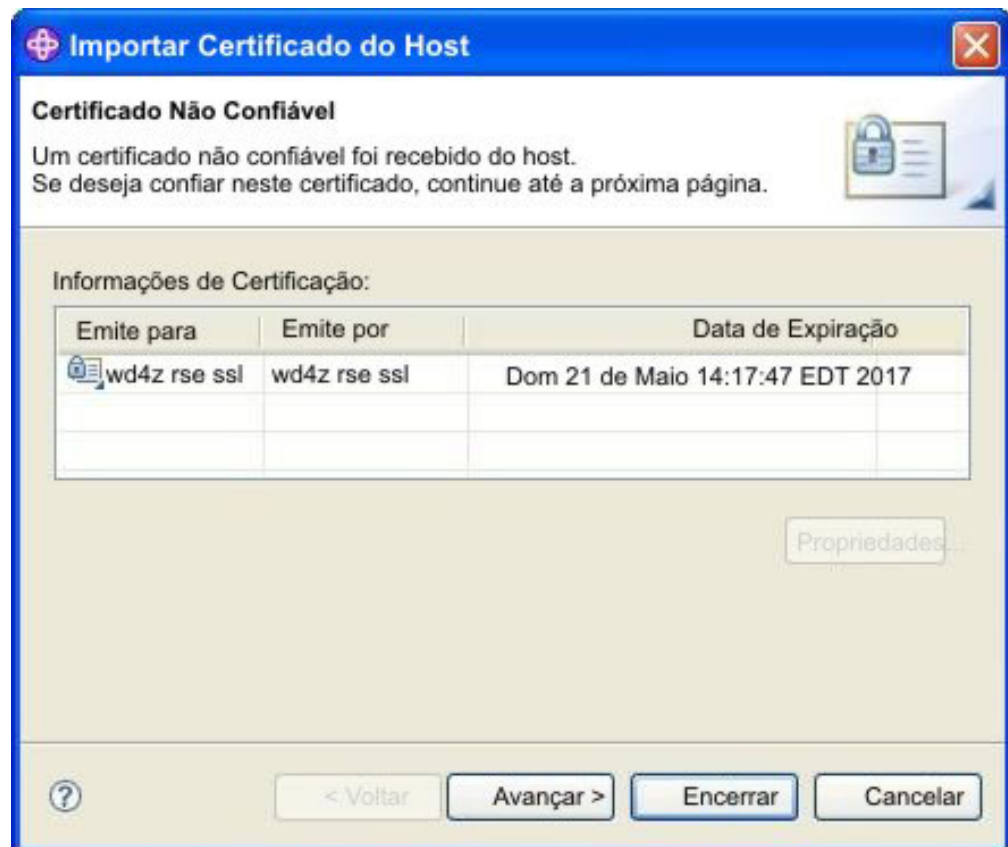


Figura 62. Diálogo Importar Certificado do Host

Clicando no botão Concluir, o usuário pode aceitar esse certificado como confiável; depois disso, a inicialização da conexão continuará.

**Nota:** O daemon RSE e o servidor RSE podem utilizar dois locais de certificados diferentes, resultando em dois certificados diferentes e, portanto, em duas 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:

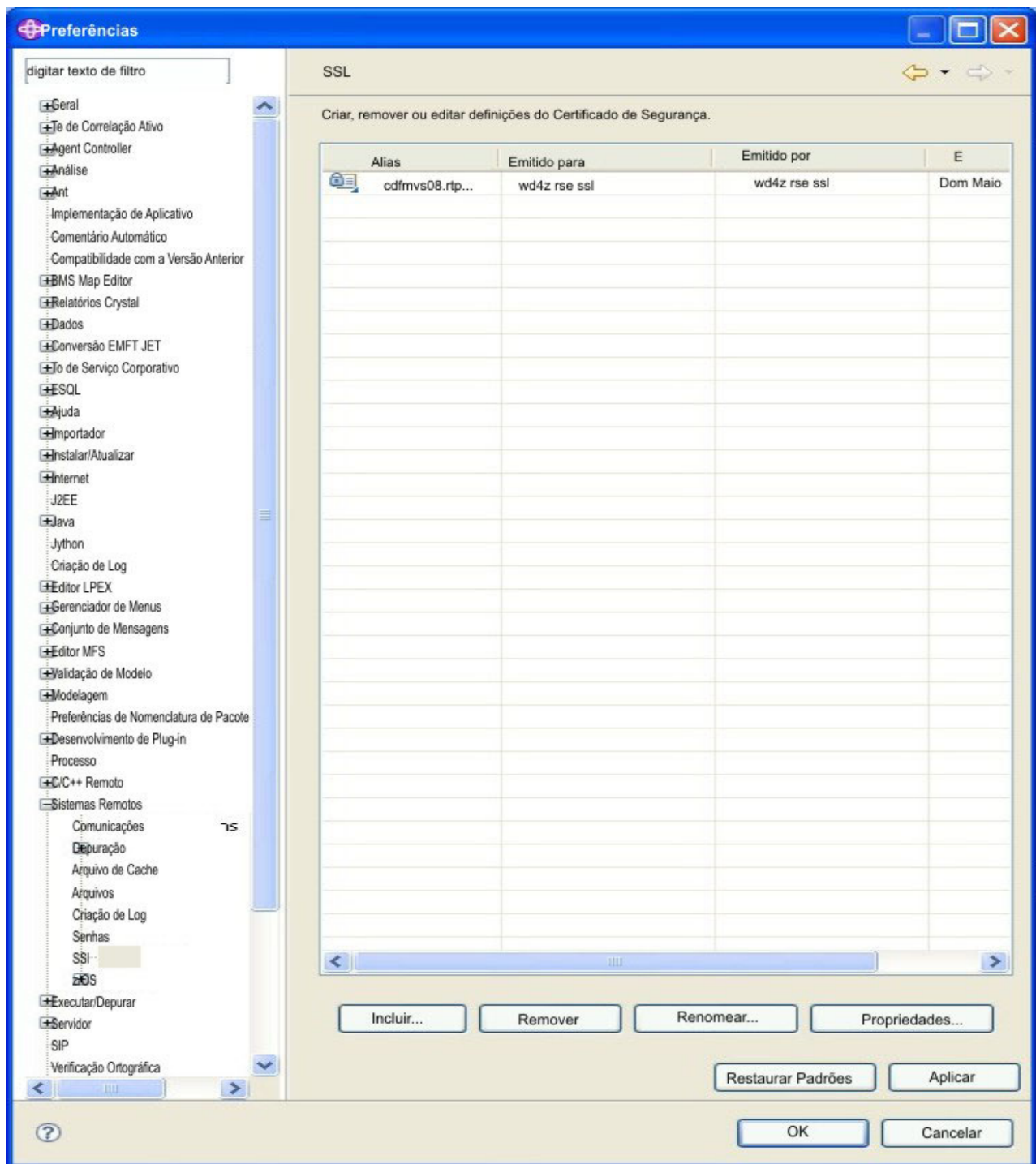


Figura 63. Diálogo Preferências - SSL

Se a comunicação SSL falhar, o cliente retornará uma mensagem de erro. Informações adicionais estão disponíveis nos diferentes arquivos de log do servidor e do usuário, conforme descrito em “Criação de Log de Daemon RSE e de Conjunto de Encadeamento” na página 137 e “Criação de Log de Usuário do RSE” na página 138.

---

## (Opcional) Incluir Suporte de Autenticação de Cliente X.509

O daemon RSE suporta que os próprios usuários se autenticuem com um certificado X.509. Usar a comunicação criptografada SSL é um pré-requisito para essa função por ser uma extensão para a autenticação de host com um certificado usado no SSL.

Há várias maneiras de fazer a autenticação de certificado para um usuário, conforme descrito em “Autenticação de Cliente Usando Certificados X.509” na página 168. As próximas etapas documentam a configuração necessária para suportar o método em que o software de segurança autentica o certificado utilizando a extensão de certificado HostIdMappings.

1. Altere o certificado que identifica a Autoridade de Certificação (CA) usado para assinar o certificado do cliente para um certificado de CA altamente confiável. Embora o status TRUST seja suficiente para a validação de certificado, é feita uma mudança para HIGHTRUST por ser usado para a parte de autenticação de certificado do processo de login.

```
RACDCERT CERTAUTH ALTER(LABEL('HighTrust CA')) HIGHTRUST
```

2. Inclua o certificado de CA no conjunto de chaves `rdzssl.racf` para que esteja disponível para validar os certificados de cliente.

```
RACDCERT ID(stcrse) CONNECT(CERTAUTH LABEL('HighTrust CA') +  
RING(rdzssl.racf))
```

Isso conclui a configuração do software de segurança para o certificado de CA.

3. Defina um recurso (formato `IRR.HOST.hostname`) na classe `SERVAUTH` para o nome do host, `CDFMVS08.RALEIGH.IBM.COM`, definido na extensão `HostIdMappings` do certificado do cliente.

```
RDEFINE SERVAUTH IRR.HOST.CDFMVS08.RALEIGH.IBM.COM UACC(NONE)
```

4. Conceda ao ID do usuário da tarefa iniciada do RSE, `STCRSE`, acesso a esse recurso com autoridade `READ`.

```
PERMIT IRR.HOST.CDFMVS08.RALEIGH.IBM.COM CLASS(SERVAUTH) +  
ACCESS(READ) ID(stcrse)
```

5. Ative suas mudanças para a classe `SERVAUTH`. Use o primeiro comando se a classe `SERVAUTH` ainda não estiver ativa. Use o segundo para atualizar uma configuração ativa.

```
SETOPTS CLASSACT(SERVAUTH) RACLIST(SERVAUTH)  
ou  
SETOPTS RACLIST(SERVAUTH) REFRESH
```

Isso conclui a configuração do software de segurança para a extensão `HostIdMappings`.

6. Reinicie a tarefa iniciada do RSE para começar a aceitar logons de clientes usando os certificados X.509.

---

## (Opcional) Criar um Banco de Dados de Chaves com gskkyman

Não execute esta etapa se você usar um conjunto de chaves compatível com SAF para o banco de dados principal do daemon RSE.

`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 informações adicionais 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
```

```
$ cd /etc/rdz/ssl
```

```
$ gskkyman          Menu do Banco de Dados
```

```
1 - Criar novo banco de dados
```

Digite o número da opção: **1**

Digite o nome do banco de dados de chaves (pressione ENTER para retornar ao menu):

```
rdzssl.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 de chaves /etc/rdz/ssl/rdzssl.kdb criado.

Pressione ENTER para continuar.

```
Menu do Key Management
```

```
6 - Criar um certificado auto-assinado
```

Digite o número da opção (pressione ENTER para retornar ao menu anterior): **6**

```
Tipo de Certificado
```

```
5 - Certificado do usuário ou do servidor com chave RSA de 1024 bits
```

Selecione o tipo de certificado (pressione ENTER para retornar ao menu): **5**

Digite o rótulo (pressione ENTER para retornar ao menu): **rdzrse**

Digite o nome do assunto para o certificado

Nome comum (necessário): **rdz rse ssl**

Unidade organizacional (opcional): **rdz**

Organização (necessário): **IBM**

Cidade/Localidade (opcional): **Raleigh**

Estado/Província (opcional): **NC**

País/Região (2 caracteres - necessário): **US**

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
```

```
0 - Sair do programa
```

Digite o número da opção (pressione ENTER para retornar ao menu anterior): **0**

```
$ ls -l rdzssl.*
```

```
total 152
```

```
-rw----- 1 IBMUSER SYS1      35080 May 24 14:24 rdzssl.kdb
```

```
-rw----- 1 IBMUSER SYS1       80 May 24 14:24 rdzssl.rdb
```

```
$ chmod 644 rdzssl.*
```



```
$ ls -l rdzssl.*
-rw-r--r-- 1 IBMUSER SYS1      35080 May 24 14:24 rdzssl.kdb
-rw-r--r-- 1 IBMUSER SYS1       80 May 24 14:24 rdzssl.rdb
```

A amostra acima inicia criando um banco de dados de chaves chamado `rdzssl.kdb` com a senha `rsessl`. Quando o banco de dados existir, ele será preenchido através da criação de um novo certificado auto-assinado, válido durante 10 anos (sem contar os dias extras). O certificado é armazenado com o rótulo `rdzrse` e com a mesma senha (`rsessl`) usada 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. 644 (o proprietário tem leitura/gravação, todos têm leitura) é uma máscara útil para o comando **chmod**.

O resultado pode ser verificado ao selecionar a opção **Mostrar Informações do Certificado** no submenu **Gerenciar Chaves e Certificados**, como a seguir:

```
$ gskkyman
```

```
Menu do Banco de Dados
```

```
2 - Abrir banco de dados
```

```
Digite o número da opção: 2
```

```
Digite o nome do banco de dados de chaves (pressione ENTER para retornar ao menu):
```

```
rdzssl.kdb
```

```
Digite a senha do banco de dados (pressione ENTER para retornar ao menu): rsessl
```

```
Menu do Key Management
```

```
1 - Gerenciar chaves e certificados
```

```
Digite o número da opção (pressione ENTER para retornar ao menu anterior): 1
```

```
Lista de Chaves e Certificados
```

```
1 - rdzrse
```

```
Digite o número do rótulo (ENTER para retornar ao menu
de seleção, p para lista anterior: 1
```

```
Menu Chave e Certificado
```

```
1 - Mostrar informações do certificado
```

```
Digite o número da opção (pressione ENTER para retornar ao menu anterior): 1
```

```
Informações do Certificado
```

```
Rótulo: rdzrse
```

```
ID do Registro: 14
```

```
ID do Registro do Emissor: 14
```

```
Confiável: Sim
```

```
Versão: 3
```

```
Número serial: 45356379000ac997
```

```
Nome do emissor: rdz rse ssl
```

```
rdz
```

```
IBM Brasil - Centro de Traduções
```

```
Rodovia SP 101 km 09
```

```
CEP 13185-900
```

```
Hortolândia, SP
```

```
NC
```

```
US
Nome do assunto: rdz rse ssl
rdz
IBM Brasil - Centro de Traduções
Rodovia SP 101 km 09
CEP 13185-900
Hortolândia, SP
```

```
NC
US
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

0 - Sair do programa

Digite o número da opção (pressione ENTER para retornar ao menu anterior): 0

A amostra `ssl.properties` a seguir mostra que as diretivas `daemon_*` são diferentes da amostra do conjunto de chaves SAF anterior.

```
$ oedit /etc/rdz/ssl/ssl.properties
-> change: enable_ssl=true
-> uncomment and change: daemon_keydb_file=rdzssl.kdb
-> uncomment and change: daemon_keydb_password=rsessl
-> uncomment and change: daemon_key_label=rdzrse
-> uncomment and change: server_keystore_file=rdzssl.racf
-> uncomment and change: server_keystore_label=rdzrse
-> uncomment and change: server_keystore_type=JCERACFKS
```

As mudanças acima ativam o SSL e informam ao daemon RSE que o certificado está armazenado com o rótulo `rdzrse` no banco de dados de chaves `rdzssl.kdb` com a senha `rsessl`. O servidor RSE ainda está usando um conjunto de chaves compatível com SAF.

---

## (Opcional) Criar um Keystore com keytool

Não execute esta etapa se você utilizar um conjunto de chaves compatível com SAF para o keystore do servidor RSE.

"keytool -genkey" gera um par de chaves privadas e um certificado auto-assinado correspondente que são armazenados como uma entrada (identificada por um alias) em um arquivo keystore (novo).

**Nota:** Java deve ser incluído nos diretórios de procura de comando. A instrução a seguir pode ser necessária para executar o `keytool`, em que `/usr/lpp/java/J5.0` é o diretório em que Java está instalado:  
`PATH=$PATH:/usr/lpp/java/J5.0/bin`

Todas as informações podem ser transmitidas como um parâmetro, mas devido às limitações no comprimento da linha de comandos, será necessária uma certa interatividade, como a seguir:

```

$ cd /etc/rdz/ssl
$ keytool -genkey -alias rdzrse -validity 3650 -keystore rdzssl.jks -storepass
rsessl -keypass rsessl
Qual é o seu nome e sobrenome?
[Desconhecido]: rdz rse ssl
Qual é o nome de sua unidade organizacional?
[Desconhecido]: rdz
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
CN=rdz rse ssl, OU=rdz, O=IBM, L=Raleigh, ST=NC, C=US está correto? (digite "sim"
ou "não")
[não]: sim
$ ls -l rdzssl.*
-rw-r--r-- 1 IBMUSER SYS1          1224 May 24 14:17 rdzssl.jks

```

O certificado auto-assinado criado acima é válido durante aproximadamente 10 anos (não contando o dia 29 de fevereiro). Ele é armazenado em /etc/rdz/ssl/rdzssl.jks utilizando o alias rdzrse. 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, como a seguir:

```

$ keytool -list -alias rdzrse -keystore rdzssl.jks -storepass rsessl -v
Nome do alias: rdzrse
Data de criação: 24 de maio de 2007
Tipo de entrada: keyEntry
Comprimento da cadeia de certificados: 1
Certificado 1:
Proprietário: CN=rdz rse ssl, OU=rdz, O=IBM, L=Raleigh, ST=NC, C=US
Emissor: CN=rdz rse ssl, OU=rdz, 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

```

A amostra ssl.properties a seguir mostra que as diretivas server\_\* são diferentes da amostra do conjunto de chaves SAF anterior.

```

$ oedit /etc/rdz/ssl/ssl.properties
-> change: enable_ssl=true
-> uncomment and change: daemon_keydb_file=rdzssl.racf
-> uncomment and change: daemon_key_label=rdzrse
-> uncomment and change: server_keystore_file=rdzssl.jks
-> uncomment and change: server_keystore_password=rsessl
-> uncomment and change: server_keystore_label=rdzrse
-> optionally uncomment and change: server_keystore_type=JKS

```

As mudanças acima ativam o SSL e informam ao servidor RSE que o certificado está armazenado com o rótulo rdzrse no keystore rdzssl.jks com a senha rsessl. O daemon RSE ainda está usando um conjunto de chaves compatível com SAF.



---

## Apêndice B. Configurando o TCP/IP

Este apêndice é fornecido para ajudá-lo com alguns problemas comuns que você pode encontrar ao configurar o TCP/IP ou durante a verificação ou a 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

Com o uso de APPC para o serviço TSO Commands, o Developer para 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 Installation Verification Program (IVP) do fekfivpt. O comando deve retornar uma saída como nesta amostra (\$ é o prompt do z/OS UNIX):

```
$ fekfivpt
```

```
Quarta-feira, 2 de julho de 2008, 13h11min54s EDT
uid=1(USERID) gid=0(GROUP)
utilizando /etc/rdz/rsed.envvars
```

```
-----
Configuração do resolvedor TCP/IP (ordem de procura do z/OS UNIX):
-----
```

```
Resolver Trace Initialization Complete -> 2008/07/02 13:11:54.745964
```

```
res_init Resolver values:
Global Tcp/Ip Dataset = None
Default Tcp/Ip Dataset = None
Local Tcp/Ip Dataset = /etc/resolv.conf
Translation Table = Default
UserId/JobName = USERID
Caller API = LE C Sockets
Caller Mode = EBCDIC
(L) DataSetPrefix = TCPIP
(L) HostName = CDFMVS08
(L) TcpIpJobName = TCPIP
(L) DomainOrigin = RALEIGH.IBM.COM
(L) NameServer = 9.42.206.2
                  9.42.206.3
(L) NsPortAddr = 53 (L) ResolverTimeout = 10
(L) ResolveVia = UDP (L) ResolverUdpRetries = 1
(*) Options NDots = 1
(*) SockNoTestStor
(*) AlwaysWto = NO (L) MessageCase = MIXED
(*) LookUp = DNS LOCAL
```

```
res_init Succeeded
```

```
res_init Started: 2008/07/02 13:11:54.755363
```

```
res_init Ended: 2008/07/02 13:11:54.755371
```

```
*****
```

```
MVS TCP/IP NETSTAT CS V1R9 TCPIP Name: TCPIP 13:11:54
```

```
Tcpip started at 01:28:36 on 06/23/2008 with IPv6 enabled
```

```
-----
```

endereço IP do host:

```
-----  
hostName=CDFMVS08  
hostAddr=9.42.112.75  
bindAddr=9.42.112.75  
localAddr=9.42.112.75
```

Êxito, os endereços correspondem

---

## 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 usará a ordem de procura nativa aplicável do MVS ou do z/OS UNIX sem nenhuma 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 usados 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 deixar o TCP/IP determinar dinamicamente o local dos arquivos de configuração, com base nas ordens de procura documentadas em *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 usada.

**Nota:** Use o recurso do resolvedor de rastreo para determinar que os valores de TCPIP.DATA estão sendo usados pelo resolvedor e de onde eles foram lidos. Para obter informações sobre como iniciar o rastreo dinamicamente, consulte *Communications Server: IP Diagnosis Guide* (GC31-8782). Depois que

o rastreamento for ativado, emita um comando TSO **NETSTAT HOME** e um comando shell **netstat -h** 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 Usadas no Ambiente do 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 usado ao tentar acessar alguns arquivos de configuração especificados nesta seção.

A ordem de procura usada para acessar o arquivo de configuração do resolvidor de base é a seguinte:

1. **GLOBALTCPIPDATA**

Se definido, o valor da instrução de configuração GLOBALTCPIPDATA do resolvidor é usado (consulte também “Compreendendo os Resolvedores” na página 308). 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 usado. 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 usado. 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, tarefa ou encadeamento).

6. **jobname.TCPIP.DATA**

jobname é o nome especificado na instrução JOB da 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 usado (consulte também “Compreendendo os Resolvedores” na página 308).

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 usados. A ordem de

procura usada para acessar o arquivo de configuração é a seguinte. A ordem de procura termina quando o primeiro arquivo é 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** da 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 resolvedor (se localizado); caso contrário, hlq é **TCPIP** por padrão.
5. Se nenhuma tabela for encontrada, o resolvedor usará uma tabela padrão codificada permanentemente, idêntica à tabela listada no membro do conjunto de dados **SEZATCPX(STANDARD)**.

## Tabelas do Host Local

Por padrão, o resolvedor 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 usadas. 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 usados. A instrução **LOOKUP TCPIP.DATA** também pode ser usada para controlar como os servidores de nomes de domínios e as tabelas do host local são usadas. Para obter mais informações sobre 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 de **Ipv4** para informações de nome de site é a seguinte. A procura termina quando o primeiro arquivo é 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** da 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 estas Informações de Configuração no Developer para System z

Conforme informado anteriormente, o Developer para System z depende de o TCP/IP ter o nome de host correto quando inicializado, ao utilizar APPC. Isto significa que o TCP/IP diferente e os arquivos de configuração do Resolver devem ser configurados corretamente.

No exemplo a seguir, nós concentraremos em algumas tarefas de configuração para o TCP/IP e o Resolver. Observe que isso não abrange uma configuração completa do TCP/IP ou do Resolver, ela destaca alguns aspectos principais que podem ser aplicáveis no seu site:

1. Na 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=EZBTCP,REGION=0M,TIME=1440,PARM=&PARMS
//PROFILE  DD   DISP=SHR,DSN=SYS1.TCPPARMS(&PROF)
//SYSTCPD  DD   DISP=SHR,DSN=SYS1.TCPPARMS(&DATA)
//SYSPRINT DD   SYSOUT=*,DCB=(RECFM=VB,LRECL=132,BLKSIZE=136)
//ALGPRINT DD   SYSOUT=*,DCB=(RECFM=VB,LRECL=132,BLKSIZE=136)
//CFGPRINT DD   SYSOUT=*,DCB=(RECFM=VB,LRECL=132,BLKSIZE=136)
//SYSOUT   DD   SYSOUT=*,DCB=(RECFM=VB,LRECL=132,BLKSIZE=136)
//CEEDUMP  DD   SYSOUT=*,DCB=(RECFM=VB,LRECL=132,BLKSIZE=136)
//SYSERROR DD   SYSOUT=*
```

2. SYS1.TCPPARMS(TCPDATA) informa que o nome do sistema deve ser o nome do host e que um servidor de nomes de domínio (DNS) não é usado; todos os nomes serão resolvidos através da consulta de tabela do site.

```
; 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 usado, 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. Na JCL do Resolver vemos que a instrução SETUP DD não é usada. Conforme mencionado em “Compreendendo os Resolvedores” na página 308, isto significa que GLOBALTCPIPDATA e outras variáveis não serão usadas.

```
//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 51 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 Usadas no Ambiente do z/OS UNIX” na página 309, 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 usado 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 usada.

5. A Tabela 51 também informa que não é necessário fazer nada para utilizar a tabela de conversão ASCII-EBCDIC padrão.
6. Assumindo que o comando **MAKESITE** do TSO não seja usado (pode criar as variáveis X\_SITE e X\_ADDR), /etc/hosts será a tabela de sites usada 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, e aconselhável manter os arquivos de configuração do TCP/IP e do Resolver em sincronismo um com o outro.

*Tabela 51. 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"> <li>1. GLOBALTCPIPDATA</li> <li>2. variável de ambiente RESOLVER_CONFIG</li> <li>3. /etc/resolv.conf</li> <li>4. SYSTCPD DD-name</li> <li>5. userid.TCPIP.DATA</li> <li>6. jobname.TCPIP.DATA</li> <li>7. SYS1.TCPPARMS(TCPDATA)</li> <li>8. DEFAULTTCPIPDATA</li> <li>9. TCPIP.TCPIP.DATA</li> </ol>

Tabela 51. Definições locais disponíveis para o resolvedor (continuação)

Descrição do Tipo do Arquivo	APIs afetadas	Arquivos do Candidato
Tabelas de Conversão	Todas as APIs	<ol style="list-style-type: none"> <li>1. variável de ambiente X_XLATE</li> <li>2. userid.STANDARD.TCPXLBIN</li> <li>3. jobname.STANDARD.TCPXLBIN</li> <li>4. hlq.STANDARD.TCPXLBIN</li> <li>5. Tabela de conversão fornecida pelo resolvedor, membro STANDARD em SEZATCPX</li> </ol>
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"> <li>1. variável de ambiente X_SITE</li> <li>2. variável de ambiente X_ADDR</li> <li>3. /etc/hosts</li> <li>4. userid.HOSTS.xxxxINFO</li> <li>5. jobname.HOSTS.xxxxINFO</li> <li>6. hlq.HOSTS.xxxxINFO</li> </ol> IPv6 <ol style="list-style-type: none"> <li>1. GLOBALIPNODES</li> <li>2. variável de ambiente RESOLVER_IPNODES</li> <li>3. userid.ETC.IPNODES</li> <li>4. jobname.ETC.IPNODES</li> <li>5. hlq.ETC.IPNODES</li> <li>6. DEFAULTIPNODES</li> <li>7. /etc/ipnodes</li> </ol>

**Nota:** A Tabela 51 na página 312 é uma cópia parcial de uma tabela em *Communications Server: IP Configuration Guide* (SC31-8775). Consulte esse manual para obter a tabela completa.

## O Endereço do Host Não É Resolvido Corretamente

Quando ocorrerem problemas nos quais o TCP/IP Resolver não pode resolver o endereço do host corretamente, muitas vezes isso é devido a um arquivo de configuração do resolvedor ausente ou incompleto. Uma indicação clara desse problema é a seguinte mensagem em `lock.log`:

```
clientip(0.0.0.0) <> callerip(<endereço IP do host>)
```

Para verificar isso, execute o `fekfivpt` TCP/IP IVP, conforme descrito em Capítulo 7, “Verificação de Instalação”, na página 107. A seção de configuração do resolvedor da saída será semelhante à seguinte amostra:

```
Resolver Trace Initialization Complete -> 2008/07/02 13:11:54.745964
```

```
res_init Resolver values:
Global Tcp/Ip Dataset = None
Default Tcp/Ip Dataset = None
Local Tcp/Ip Dataset = /etc/resolv.conf
Translation Table = Default
UserId/JobName = USERID
Caller API = LE C Sockets
Caller Mode = EBCDIC
```

Certifique-se de que as definições no arquivo (ou conjunto de dados) referido pelo “Local Tcp/Ip Dataset” estejam corretas.

Esse campo ficará em branco se você não utilizar um nome padrão para o arquivo resolvidor de IP (utilizando a ordem de procura do z/OS UNIX). Nesse caso, inclua a seguinte instrução no `rsed.envvars`, em que <arquivo resolvidor> ou <dados do resolvidor> representam o nome do arquivo resolvidor de IP:

```
RESOLVER_CONFIG=<arquivo resolvidor>
```

ou

```
RESOLVER_CONFIG='<conjunto de dados do resolvidor>'
```

---

## Apêndice C. Configurando o INETD

Este apêndice é fornecido para ajudá-lo com alguns problemas comuns que você pode encontrar ao configurar o INETD ou durante a verificação ou a modificação de uma configuração existente. INETD é usado pelo Developer para System z para a funcionalidade REXEC/SSH.

O daemon do INETD fornece gerenciamento de serviço para uma rede IP. Ele reduz o carregamento 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 usado 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 transpor várias linhas, seguindo essas regras de sintaxe adicionais:
  - 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 usado, 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 usado para o serviço.

**protocol[,sndbuf=n[,rcvbuf=n]]**

protocol pode ser tcp[4|6] ou udp[4|6], e é usado 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 usado 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. Ele pode ser um conjunto de dados 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 de quatro 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 usado para o serviço

**protocol**

Especifica o protocolo de transporte usado para o serviço. Os valores válidos são tcp e udp

**aliases**

Especifica uma lista de nomes de serviços não-oficiais

## Ordem de Procura Usada no Ambiente z/OS UNIX

A ordem de procura usada para acessar ETC.SERVICES no z/OS UNIX é a seguinte. A procura termina quando o primeiro arquivo é localizado:

1. **/etc/services**
2. **userid.ETC.SERVICES**  
userid é o ID do usuário usado para iniciar INETD.
3. **hlq.ETC.SERVICES**  
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.

## Ordem de Procura Usada no Ambiente MVS Ambiente

A ordem de procura usada para acessar ETC.SERVICES no MVS nativo é a seguinte. A procura termina quando o primeiro conjunto de dados é localizado:

1. **//Placa DD SERVICES**  
O conjunto de dados alocado para a instrução DD SERVICES é usado
2. **userid.ETC.SERVICES**  
userid é o ID do usuário usado para iniciar INETD.
3. **jobname.ETC.SERVICES**  
jobname é o nome especificado na instrução JOB 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 resolvedor (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 é usada apenas ao iniciar um módulo de carregamento MVS, como o 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 usadas pelo TCPIP para ver se a porta está reservada para um determinado serviço. ETC.SERVICES é usado 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 usar o nome do procedimento JCL iniciado para o Espaço de Endereço de Kernel do z/OS UNIX para permitir que quase todos os processos sejam ligados à porta. Normalmente, esse nome é OMVS, a menos que um nome diferente seja especificado explicitamente no parâmetro STARTUP\_PROC do membro BPXPRMxx parmlib.

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 usam o nome da tarefa em lote.
- Os aplicativos iniciados do console do operador do MVS usam o nome do procedimento iniciado (STC) como o nome da tarefa.
- Os aplicativos executados de um ID de usuário do TSO usam 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 usam 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 de JCL iniciada para o Espaço de Endereço de Kernel dos Serviços do Sistema UNIX pode ser usado para permitir que quase todos os responsáveis pela chamada da API do soquete `bind()` (exceto usuários da API Pascal) se liguem à porta. Normalmente, esse nome é OMVS, a menos que um nome diferente seja especificado explicitamente no parâmetro STARTUP\_PROC do membro BPXPRMxx parmlib.
- Aplicativos z/OS UNIX iniciados pelo INETD usam 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 usado para identificar registros syslog originados do processo INETD e para fornecer o valor PID para comandos que requerem um, como `kill`. Também é usado como um mecanismo de bloqueio para impedir que mais de um 1 processo INETD esteja ativo.

---

## Inicialização

A implementação de INETD do z/OS UNIX está localizada por padrão em `/usr/sbin/inetd` e suporta dois parâmetros de inicialização opcionais 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 *Communications Server IP Configuration Guide* (SC31-8775) para obter informações adicionais 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`

Você deve 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 usados 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 ele é encerrado (um WTOR é enviado ao 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 usado para iniciar o INETD:

```
# Start INETD
inetd::respfrk:/usr/sbin/inetd /etc/inetd.conf
```

**Nota:** Lembre-se de que o parâmetro `respfrk` usado 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 saber mais sobre inittab.

## BPXBATCH

O método de inicialização BPXBATCH funciona para tarefas iniciadas 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. A JCL listada na seguinte amostra de código é um procedimento de amostra para iniciar INETD (a etapa KILL remove um processo INETD ativo, se houver algum):

```
//INETD    PROC PRM=
//*
//KILL      EXEC PGM=BPXBATCH,REGION=0M,
//          PARM='SH ps -e | grep inetd | cut -c 1-10 | xargs -n 1 kill'
//*
//INETD     EXEC PGM=BPXBATCH,REGION=0M,
//          PARM='PGM /usr/sbin/inetd &PRM'
//STDERR    DD SYSOUT=*
//* STDIN, STDOUT e STDENV são padronizados como /dev/null
//*
```

Figura 64. 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. Desde o z/OS 1.7, SYSOUT pode ser designado a STDOUT e STDERR. Consulte *UNIX System Services Command Reference* (SA22-7802) para saber mais sobre BPXBATCH.
- 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 shell, o INETD utiliza a procura do z/OS UNIX para localizar ETC.SERVICES. Os seguintes comandos de amostra podem ser usados (por uma pessoa com autoridade suficiente) para parar e iniciar o INETD (# é o prompt do z/OS UNIX):

```
# ps -e | grep inetd
7 ?          0:00 /usr/sbin/inetd
# kill 7
# _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 configurados para o ID do usuário, junto 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'))
```

INETD é um daemon que exige acesso a funções como `setuid()`. Portanto, o ID do usuário usado para iniciar o INETD requer acesso READ 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 EXECUTE para o programa `inetd` (/usr/sbin/inetd), acesso READ para os arquivos `inetd.conf` e `ETC.SERVICES` e acesso WRITE para `/etc/inetd.pid`. Se você deseja executar o INETD sem o UID 0, pode conceder acesso CONTROL 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. Use o comando **extattr +p** para tornar um arquivo z/OS UNIX controlado por programa. Use 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 READ 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 aos perfis individuais que abrangem as permissões necessárias

- Acesso READ a SUPERUSER.PROCESS.GETPSENT na classe UNIXPRIV (para o comando **ps**)

- Acesso READ a SUPERUSER.PROCESS.KILL na classe UNIXPRIV (para o comando **kill**)

- Acesso READ a BPX.JOBNAME na classe FACILITY (para a variável de ambiente `_BPX_JOBNAME`)

Consulte *UNIX System Services Command Reference* (SA22-7802) para saber mais sobre os comandos **extattr** e **su**. Consulte *UNIX System Services Planning* (GA22-7800) para saber mais sobre a classe UNIXPRIV e perfis BPX.\* na classe

FACILITY. Consulte *Security Server RACF Security Administrator's Guide* (SA22-7683) para obter informações adicionais sobre as definições de segmento OMVS e a classe PROGRAM.

---

## Requisitos do Developer para System z

O Developer para System z depende do INETD para gerenciar REXEC e/ou SSH. Ele também pode impor requisitos adicionais para a configuração do INETD descrita acima.

REXEC (ou SSH) é usado para os dois seguintes propósitos, conforme descrito em “(Opcional) Usando REXEC (ou SSH)” na página 100.

- ações remotas (baseadas em host) em subprojetos do z/OS UNIX
- método alternativo de inicialização do servidor RSE

As ações remotas em subprojetos z/OS UNIX não requerem configurações especiais. O método de inicialização RSE alternativo, no entanto, requer configurações especiais.

### 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 pela 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 usado, sendo 0 como padrão.

### REXEC (ou SSH)

O daemon REXEC (ou SSH) que é iniciado pelo INETD quando um cliente se conecta à porta 512 (ou 22, respectivamente) é usado para fazer autenticação, iniciar o servidor RSE e retornar o número da porta para favorecer a comunicação com o cliente. Para isso, o ID do usuário designado ao daemon do REXEC (ou SSH) (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 READ ao perfil `BPX.DAEMON` na classe FACILITY
- Acesso READ e EXECUTE aos diretórios de instalação do Developer para System z, padrão `/usr/lpp/rdz/*`
- Acesso READ e EXECUTE aos diretórios de configuração do Developer para System z, padrão `/etc/rdz/*`

---

## Apêndice D. Configurando o APPC

Este apêndice é fornecido para ajudá-lo com alguns problemas comuns que você pode encontrar ao configurar o APPC (Advanced Program-to-Program Communication) ou durante a verificação e a 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 de APPC e sobre 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

APPC/MVS armazena seus dados de configuração nos seguintes membros SYS1.PARMLIB e 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 usados 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 para System z ativa um novo TP chamado FEKFRSRV, mencionado como o serviço TSO Commands.

A tarefa a seguir é uma concatenação dos membros de amostra SYS1.SAMPLIB(ATBTPVSM) e SYS1.SAMPLIB(ATBSIVSM), e pode ser usada para definir os VSAMs do APPC.

```

//APPCVSAM JOB <parâmetros da tarefa>
//*
/* CUIDADO: Isso não é um procedimento JCL nem uma tarefa completa.
/* Antes de usar 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 conterà 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 65. 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 Unidade Lógica (LU). 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 usadas.

A parte específica do APPC da configuração VTAM consiste em três etapas:

1. Defina o nome de modo usado (por exemplo, APPCHOST) como VTAM usando SYS1.SAMPLIB(ATBLJOB) para fazer a montagem e a edição de link de 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. Use o comando do console **v net,act,id=atbappl** para ativar o aplicativo recém-definido (em que net é igual ao nome da tarefa iniciada do VTAM). Use

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 usado no membro de amostra SYS1.SAMPLIB(ATBAPPL) pode ser usado 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             C

```

Figura 66. SYS1.SAMPLIB(ATBAPPL)

Consulte *Communications Server IP SNA Network Implementation Guide* (SC31-8777) para obter informações adicionais sobre a configuração de VTAM.

---

## SYS1.PARMLIB(APPCPMxx)

Para ativar e suportar o fluxo de comunicação entre os sistemas, os sites devem definir Unidades Lógicas (LUs) entre as sessões que podem ser ligadas. 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 TSO Commands requer que o APPC seja configurado para ter uma LU de base que possa manipular 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 usados.

A amostra de código a seguir é um membro SYS1.PARMLIB(APPCPMxx) que pode ser usado para o serviço TSO Commands.

```

LUADD
  ACBNAME(MVSLU01)
  BASE
  SCHED(ASCH)
  TPDATA(SYS1.APPCTP)
  SIDEINFO DATASET(SYS1.APPCSI)

```

Figura 67. 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 da seguinte forma:



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).

**Nota:** O comando do operador **D APPC,LU,ALL** mostrará todas as definições de LU ativas e marcará a LU de base.

Se seu sistema possui uma LU com parâmetros BASE e NOSCHED, esta LU seria usada como a LU BASE mas o serviço TSO Commands não funcionará porque esta LU não possui um planejador de transações para manipular 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 “Arquivo de Configuração do RSE `rsed.envvars`” na página 31 para obter informações adicionais 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 usada para o serviço TSO Commands deve ter iniciadores APPC suficientes para permitir um iniciador para cada usuário do Developer para System z.

O serviço TSO Commands também precisa das especificações padrão para ser especificado nas seções `OPTIONS` e `TPDEFAULT`.

A amostra de código a seguir é um membro `SYS1.PARMLIB(ASCHPMxx)` que pode ser usado para o serviço TSO Commands.

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

OPTIONS
  DEFAULT(A)

TPDEFAULT
  REGION(2M)
  TIME(5)
  MSGLEVEL(1,1)
  OUTCLASS(X)
```

*Figura 68. SYS1.PARMLIB(ASCHPMxx)*



**Nota:**

- 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.
- O comando do operador **D ASCH,ALL** mostrará todas as classes de planejador de transação APPC ativas.

---

## 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. Insira os seguintes comandos do console para iniciar APPC/MVS (em que xx é igual aos dois últimos caracteres dos membros SYS1.PARMLIB relacionados):
  1. S APPC,SUB=MSTR,APPC=xx
  2. S ASCH,SUB=MSTR,ASCH=xxInclua 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 o seguinte comando de console **SET** (em que xx é igual aos dois últimos 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 usados para verificar a configuração do APPC. Consulte *MVS System Commands* (GC28-1781) para obter informações adicionais sobre os comandos do console mencionados.

---

## Definindo a Transação de Serviço Comandos do TSO

Quando o APPC/MVS está ativo, o serviço TSO Commands do Developer para System z pode ser definido, conforme descrito em “(Opcional) Transação APPC para o Serviço de Comandos TSO” na página 102.

A maneira documentada para definir a transação do APPC é customizar e enviar a FEK.#CUST.JCL(FEKAPPC).

A transação APPC também pode ser definida interativamente por meio da interface ISPF do APPC, documentada em um white paper. Esse White Paper descreve também como configurar a transação APPC para coletar informações de contabilidade específicas ao usuário.

O White Paper do *APPC e do WebSphere Developer para System z* (SC23-5885-00) está disponível na biblioteca da Internet em Developer para System z, <http://www-306.ibm.com/software/awdtools/rdz/library/>.

**Nota:** A JCL do TP, usada pelo APPC para iniciar o serviço Comandos do TSO, foi alterada no Developer para System z versão 7.1. Se você seguir as orientações no White Paper para definir o TP, deverá incluir a palavra-chave NESTMACS na linha PARM, por exemplo:

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

---

## (Opcional) Opções de Configuração Alternativas

O Developer para System z suporta opções de configuração APPC e VTAM alternativas, sendo que algumas estão documentadas nesta seção.

### Nome de Transação Alternativo

O nome da transação padrão para o serviço TSO Commands é FEKFRSRV, conforme descrito em “(Opcional) Transação APPC para o Serviço de Comandos TSO” na página 102. Conforme descrito na mesma seção, esse nome pode ser alterado ao definir a transação para APPC.

Observe que a alteração do nome da transação no APPC significa que o novo nome deverá ser designado a `_FEKFSCMD_TP_NAME_` em `rsed.envvars`, conforme descrito em “Arquivo de Configuração do RSE `rsed.envvars`” na página 31.

### Várias LUs

APPC é um protocolo de comunicação que permite que um programa (o nó parceiro) interaja com um programa no host (o nó local). Com o Developer para System z, o nó parceiro (servidor TSO Commands) e o nó local (servidor RSE) ficam ativos no mesmo sistema z/OS. Por padrão, ambos usam a mesma definição de LU (BASE) para estabelecer comunicação.

Você pode especificar um nome de LU parceira alternativo para o serviço TSO Commands na diretiva `_FEKFSCMD_PARTNER_LU_` de `rsed.envvars`, conforme descrito em “Arquivo de Configuração do RSE `rsed.envvars`” na página 31. Observe que não é possível alterar a LU local, que deve ser sempre uma LU BASE válida (com as palavras-chave BASE e SCHED).

### Segurança da LU

O VTAM suporta uma configuração de APPC segura, na qual a comunicação entre a LU parceira e a local deve ser definida para o software de segurança.

Ele é ativado ao incluir `VERIFY=REQUIRED` na definição do VTAM da LU (BASE) local. As definições de segurança devem ser feitas na classe APPCLU, conforme descrito em *MVS Planning: APPC/MVS Management* (SA22-7599).

Observe que se essa configuração estiver ativa no VTAM e a configuração de seu software de segurança não estiver concluída, a comunicação com o serviço TSO Commands falhará na inicialização sem qualquer mensagem no log do sistema indicando que o VTAM recusou a configuração da conexão. O teste IVP do APPC (`fekfivpa`) falhará com a mensagem “Return code 1 - Allocate Failure no retry”.

---

## Apêndice E. Requisitos

Este apêndice lista os pré-requisitos e correquisitos do host para esta versão do Developer para System z.

Consulte *Rational Developer para System z Prerequisites* (SC23-7659) na biblioteca on-line do Developer para System z em <http://www-01.ibm.com/software/awdtools/rdz/library/> para obter uma lista atualizada de requisitos obrigatórios e opcionais.

Os produtos listados nesta seção estão todos disponíveis no momento da publicação deste manual. Consulte o Web site do IBM Software Support Lifecycle <http://www.ibm.com/software/support/lifecycle/>, para ver se um produto selecionado ainda está disponível no momento em que você deseja usar a função relacionada do Developer para System z.

---

### Pré-requisitos de Host do z/OS

O uso do Developer para System z exige que você tenha o seguinte ambiente com os pré-requisitos apropriados:

#### z/OS

Um dos seguintes níveis deve estar instalado no host:

Número do Programa	Nome do Produto	PTFs ou Níveis de Serviço Necessários
5694-A01	z/OS v 1.11	ISPF: <ul style="list-style-type: none"><li>• APAR OA29489 (TSO/ISPF Client Gateway) PTF UA51713</li></ul> TCP/IP: <ul style="list-style-type: none"><li>• Nenhum PTF ou Nível de Serviço necessário</li></ul>
5694-A01	z/OS v 1.10	ISPF: <ul style="list-style-type: none"><li>• APAR OA29489 (TSO/ISPF Client Gateway) PTF UA51712</li></ul> TCP/IP: <ul style="list-style-type: none"><li>• APAR PK74282 (crescimento de armazenamento fixo CSM) PTF UK41810</li></ul>
5694-A01	z/OS v 1.9	ISPF: <ul style="list-style-type: none"><li>• APAR OA29489 (TSO/ISPF Client Gateway) PTF UA51687</li></ul> TCP/IP: <ul style="list-style-type: none"><li>• APAR PK74282 (crescimento de armazenamento fixo CSM) PTF UK41812</li></ul>

Número do Programa	Nome do Produto	PTFs ou Níveis de Serviço Necessários
5694-A01	z/OS v 1.8	<p>ISPF:</p> <ul style="list-style-type: none"> <li>• APAR OA20345 (saída de comando aninhada) PTF UA33575</li> <li>• APAR OA20449 (incluir suporte ao NESTMACS) PTF UA34052</li> <li>• APAR OA29489 (TSO/ISPF Client Gateway) PTF UA51686</li> </ul> <p>TCP/IP:</p> <ul style="list-style-type: none"> <li>• APAR PK74282 (crescimento de armazenamento fixo CSM) PTF UK41811</li> </ul>

O Web site do produto relacionado é:

<http://www-03.ibm.com/systems/z/os/zos/>

#### Notas:

1. Ações remotas (baseadas em host) para subprojetos do z/OS UNIX exigem que a versão z/OS UNIX de REXEC ou SSH esteja ativa no host.
2. O z/OS inclui os seguintes componentes, que precisam ser instalados, configurados e estar funcionando:
  - Interactive System Productivity Facility (ISPF)
    - <http://www-01.ibm.com/software/awdtools/ispf/>
  - Language Environment
    - <http://www-03.ibm.com/servers/eserver/zseries/zos/le/>
  - RACF ou produto de segurança equivalente
    - <http://www-03.ibm.com/servers/eserver/zseries/zos/racf/>
  - Componente VTAM do IBM Communications Server
    - <http://www-01.ibm.com/software/network/commserver/zos/>
  - Componente Serviços IP do IBM Communications Server
    - <http://www-01.ibm.com/software/network/commserver/zos/>
  - Binder
  - APPC (opcional)

#### Nota:

- O APPC é um pré-requisito obrigatório se o serviço para o ISPF APAR OA29489 não estiver disponível no seu sistema de host.
- O APPC pode ser substituído pela funcionalidade do ISPF Client Gateway fornecido no ISPF para z/OS 1.10 e disponível no ISPF para z/OS 1.8 e 1.9 com os PTFs apropriados aplicados.

## SMP/E

Para instalar o Developer para System z, um dos seguintes níveis deve ser instalado:

Número do Programa	Nome do Produto	PTFs ou Níveis de Serviço Necessários
5655-G44	IBM System Modification Program Extended (SMP/E) para z/OS v 3.5	Nenhum PTF ou Nível de Serviço necessário
5655-G44	IBM System Modification Program Extended (SMP/E) para z/OS v 3.4	Nenhum PTF ou Nível de Serviço necessário

O Web site do produto relacionado é:

<http://www-03.ibm.com/systems/z/os/zos/smpe/>

## SDK para z/OS Java 2 Technology Edition

Para suportar aplicativos que usam o Remote Systems Explorer (RSE), um dos seguintes níveis deve ser instalado no host:

Número do Programa	Nome do Produto	PTFs ou Níveis de Serviço Necessários
5655-R32	IBM 64-bit SDK para z/OS, Java 2 Technology Edition, v 6.0	release de serviço 7
5655-R31	IBM 31-bit SDK para z/OS, Java 2 Technology Edition, v 6.0	release de serviço 7
5655-N99	IBM 64-bit SDK para z/OS, Java 2 Technology Edition, v 5.0	release de serviço 11
5655-N98	IBM 31-bit SDK para z/OS, Java 2 Technology Edition, v 5.0	release de serviço 11

O Web site do produto relacionado é:

<http://www.ibm.com/servers/eserver/zseries/software/java/>

**Nota:** O PTF para Developer para System z APAR PM07305 deve ser aplicado ao usar uma versão de 64-bit do Java. O PTF está disponível por meio da página de serviço recomendada do Developer para System z, <http://www-01.ibm.com/support/docview.wss?rs=2294&context=SS2QJ2&uid=swg27006335>.

---

## Co-requisitos de Host do z/OS

Os produtos listados nesta seção e outros softwares declarados são necessários para suportar recursos específicos do Developer para System z. O cliente da estação de trabalho do Developer para System z pode ser instalado com êxito sem esses requisitos. Porém, um requisito do host estabelecido deve estar instalado e em funcionamento no tempo de execução para que o recurso correspondente funcione conforme projetado.

## z/OS

Número do Programa	Nome do Produto	PTFs ou Níveis de Serviço Necessários
5694-A01	z/OS v 1.11	HLASM Nenhum PTF ou Nível de Serviço necessário  XL C/C++ Nenhum PTF ou Nível de Serviço necessário  SCLM Nenhum PTF ou Nível de Serviço necessário  LE (PL/I) Nenhum PTF ou Nível de Serviço necessário  TN3270 Nenhum PTF ou Nível de Serviço necessário
5694-A01	z/OS v 1.10	HLASM Nenhum PTF ou Nível de Serviço necessário  XL C/C++ Nenhum PTF ou Nível de Serviço necessário  SCLM Nenhum PTF ou Nível de Serviço necessário  LE (PL/I) Nenhum PTF ou Nível de Serviço necessário  TN3270 Nenhum PTF ou Nível de Serviço necessário
5694-A01	z/OS v 1.9	HLASM Nenhum PTF ou Nível de Serviço necessário  XL C/C++ Nenhum PTF ou Nível de Serviço necessário  SCLM <ul style="list-style-type: none"> <li>• APAR OA27379 (Procura SCLM)              PTF UA46330 + UA46331, UA46332, UA46333, UA46334              (dependente do- idioma)</li> </ul> LE (PL/I) Nenhum PTF ou Nível de Serviço necessário  TN3270 Nenhum PTF ou Nível de Serviço necessário

Número do Programa	Nome do Produto	PTFs ou Níveis de Serviço Necessários
5694-A01	z/OS v 1.8	HLASM Nenhum PTF ou Nível de Serviço necessário  XL C/C++ Nenhum PTF ou Nível de Serviço necessário  SCLM <ul style="list-style-type: none"> <li>• APAR OA21104 (modo de construção informativo) PTF UA35046 + UA35056, UA35057, UA35058 ou UA35059 (dependente do idioma)</li> <li>• APAR OA16924 (aprimorar SCLMINFO) PTF UA29772 + UA29922, UA29923, UA29924 ou UA29925 (dependente do idioma)</li> <li>• APAR OA16804 (incluir suporte substituto ao ID do usuário) PTF UA33524 + UA33533, UA33534, UA33535 ou UA33536 (dependente do idioma)</li> </ul> LE (PL/I) <ul style="list-style-type: none"> <li>• APAR PK41552 (novas mensagens de PL/I para Developer para System z ) PTF UK24482 (Inglês) ou UK24483 (Japonês)</li> </ul> TN3270 Nenhum PTF ou Nível de Serviço necessário

O Web site do produto relacionado é:

<http://www-03.ibm.com/systems/z/os/zos/>

**Nota:**

1. JES3 v 1.10 ou superior é um co-requisito para usuários do JES3 que desejam usar o suporte do Monitor de Tarefas para visualizar a saída de tarefas ativas.
2. O High Level Assembler (HLASM) deve estar instalado no host com o nível de serviço listado, a fim de compilar programas assembler desenvolvidos ou editados no Developer para System z.

O Web site do produto relacionado é:

<http://www.ibm.com/software/awdtools/hlasm/>

3. O compilador XL C/C++ deve estar instalado no host com o nível de serviço listado, a fim de compilar programas C/C++ desenvolvidos ou editados no Developer para System z.

O Web site do produto relacionado é:

<http://www.ibm.com/software/awdtools/czos/>

4. O SCLM deve estar instalado no host com o nível de serviço listado, a fim de suportar o SCLM Developer Toolkit.

O Web site do produto relacionado é:

<http://www.ibm.com/software/awdtools/sclmsuite/sclm/>

**Nota:**

- O APAR OA16804 só será necessário se você quiser usar criação, promoção e implementação seguras.
  - O APAR OA26997 só é necessário para suporte de segurança de membro.
  - O APAR OA27379 só é necessário para suporte de segurança de membro ou funcionalidade de procura SCLM.
5. O Ambiente de Linguagem deve ser instalado no host com o nível de serviço listado, a fim de suportar o Enterprise Service Tools para PL/I.  
O Web site do produto relacionado é:

<http://www-03.ibm.com/servers/eserver/zseries/zos/le/>

6. O TN3270 deve estar instalado no host com o nível de serviço listado, a fim de suportar o emulador do Host Connect. O TN3270 é uma parte do componente de Serviços IP do IBM Communications Server.  
O Web site do produto relacionado é:

<http://www-01.ibm.com/software/network/commserver/zos/>

## Compilador de COBOL

Para compilar programas COBOL desenvolvidos ou editados no Developer para System z, um dos seguintes níveis deve ser instalado no host:

Número do Programa	Nome do Produto	PTFs ou Níveis de Serviço Necessários
5655-S71	IBM Enterprise COBOL para z/OS v 4.2	Nenhum PTF ou Nível de Serviço necessário
5655-S71	IBM Enterprise COBOL para z/OS v 4.1	Nenhum PTF ou Nível de Serviço necessário
5535-G53	IBM Enterprise COBOL para z/OS v 3.4	Nenhum PTF ou Nível de Serviço necessário

O Web site do produto relacionado é:

<http://www.ibm.com/software/awdtools/cobol/zos/>

**Nota:** O IBM Enterprise COBOL para z/OS v 4.1 é necessário para que o Enterprise Service Tools gere a Conversão XML Compilada que usa o recurso XML PARSE baseado em XMLSS no COBOL v 4.1.

## Compilador PL/I

Para compilar programas PL/I desenvolvidos ou editados no Developer para System z, um dos seguintes níveis deve ser instalado no host:

Número do Programa	Nome do Produto	PTFs ou Níveis de Serviço Necessários
5655-H31	IBM Enterprise PL/I para z/OS v 3.9	Nenhum PTF ou Nível de Serviço necessário
5655-H31	IBM Enterprise PL/I para z/OS v 3.8	Nenhum PTF ou Nível de Serviço necessário



Número do Programa	Nome do Produto	PTFs ou Níveis de Serviço Necessários
5655-H31	IBM Enterprise PL/I para z/OS v 3.7	Nenhum PTF ou Nível de Serviço necessário
5655-H31	IBM Enterprise PL/I para z/OS v 3.6	Nenhum PTF ou Nível de Serviço necessário

O Web site do produto relacionado é:

<http://www.ibm.com/software/awdtools/pli/plizos/>

## Debug Tool para z/OS

Para suportar depuração remota do Developer para System z, um dos seguintes níveis deve ser instalado no host:

Número do Programa	Nome do Produto	Linguagem de Programação	Níveis de APARs, PTFs ou Serviço Necessários
5655-V50	IBM Debug Tool para z/OS V10.1	COBOL, PL/I, C/C++, assembler e recursos adicionais	todas as manutenções disponíveis
5655-U27	IBM Debug Tool para z/OS V9.1	COBOL, PL/I, C/C++, assembler e recursos adicionais	todas as manutenções disponíveis
5655-S16	IBM Debug Tool Utilities and Advanced Functions para z/OS V8.1.0	COBOL, PL/I, C/C++, assembler e recursos adicionais	todas as manutenções disponíveis
5655-S17	IBM Debug Tool para z/OS V8.1.0	COBOL, PL/I, Assembler e C/C++	todas as manutenções disponíveis

**Nota:** Suporte para configurações de depuração CICS no IBM Rational Developer para System z v7.6.1 ou mais alta requer uma Ferramenta de Depuração IBM v10.1 ou v9.1 (PTF número - UK52904).

O Web site do produto relacionado é:

<http://www.ibm.com/software/awdtools/debugtool/>

**Nota:** O Debug Tool Utilities and Advanced Functions (DTU&AF) é um superconjunto do Debug Tool.

Começando com a versão 9, o Debug Tool para z/OS e os Utilitários e as Funções Avançadas da Ferramenta de Depuração foram mesclados em uma única oferta.

## CICS Transaction Server

Para suportar aplicativos com instruções integradas do CICS, um dos seguintes níveis deve ser instalado:

Número do Programa	Nome do Produto	PTFs ou Níveis de Serviço Necessários
5655-S97	IBM CICS Transaction Server para z/OS v 4.1	Nenhum PTF ou Nível de Serviço necessário
5697-E93	IBM CICS Transaction Server para z/OS v 3.2	UK34221
5697-E93	IBM CICS Transaction Server para z/OS v 3.1	UK15767, UK15764, UK11782, UK11294, UK12233, UK12521, UK15261, UK15271, UK34221, UK34078

O Web site do produto relacionado é:

<http://www.ibm.com/software/htp/cics/tserver/>

**Nota:**

- O CICS Transaction Server exige configuração adicional para funcionar com a ferramenta de depuração.
- A interface RESTful disponível no CICS Transaction Server v 4.1 ou superior é necessária para suportar os recursos Application Deployment Manager, Service Component Architecture e Enterprise Service Tools, que são novos no IBM Rational Developer para System z v 7.6 ou superior.
- O CICS Transaction Server v 3.2 ou superior é necessário para suportar muitos recursos do Enterprise Service Tools.

Para obter a lista completa de especificações sobre requisitos de tempo de execução, consulte a documentação do Enterprise Service Tools no Centro de Informações do IBM Rational Developer para System z em <http://publib.boulder.ibm.com/infocenter/ratdevz/v7r6/>.

- O CICS Transaction Server v 3.1 com o serviço UK34221 é o mínimo para o Application Deployment Manager.

## IMS

Para suportar aplicativos que usam banco de dados IMS e comunicações de dados, um dos seguintes níveis deve ser instalado no host:

Número do Programa	Nome do Produto	PTFs ou Níveis de Serviço Necessários
5635-A02	IBM IMS v 11.1	Nenhum PTF ou Nível de Serviço necessário
5635-A01	IBM IMS v 10.1	Nenhum PTF ou Nível de Serviço necessário
5655-J38	IBM IMS v 9.1	Nenhum PTF ou Nível de Serviço necessário

O Web site do produto relacionado é:

<http://www.ibm.com/software/data/ims/ims/>

**Nota:**

- O IMS requer configuração adicional para funcionar com a ferramenta de depuração.

- A Versão 10.1 ou superior do IMS, o IMS Connect e o IMS SOAP Gateway são necessários para o Enterprise Service Tools.

## DB2 para z/OS

Para suportar o DB2, um dos seguintes níveis deve ser instalado no host:

Número do Programa	Nome do Produto	PTFs ou Níveis de Serviço Necessários
5635-DB2	IBM DB2 para z/OS v 9.1	Nenhum PTF ou Nível de Serviço necessário
5625-DB2	IBM DB2 Universal Database para z/OS v 8.1	Nenhum PTF ou Nível de Serviço necessário

O Web site do produto relacionado é:

<http://www.ibm.com/software/data/db2/zos/>

## Rational Team Concert para System z

Para controle de origem baseado no Jazz que usa projetos remotos do Developer para System z, o seguinte nível deve ser instalado.

Número do Programa	Nome do Produto	PTFs ou Níveis de Serviço Necessários
5724-V82	Rational Team Concert para System z Server v 2.0	<p>FMID HAHA200 – Servidor de Time</p> <ul style="list-style-type: none"> <li>• UK54064</li> <li>• UK54071</li> <li>• UK54073</li> <li>• UK54095</li> <li>• UK54098</li> </ul> <p>FMID HAHB200 – Kit de ferramentas</p> <ul style="list-style-type: none"> <li>• UK54065</li> <li>• UK54066</li> <li>• UK54099</li> </ul> <p>FMID HAHC200 – Monitor de Tarefa</p> <ul style="list-style-type: none"> <li>• Nenhum PTF ou Nível de Serviço necessário</li> </ul> <p>FMID HAHD200 – Agente BuildForge</p> <ul style="list-style-type: none"> <li>• UK54097</li> </ul>

O Web site do produto relacionado é:

<http://www-01.ibm.com/software/awdtools/rtc/>

**Nota:** O Rational Team Concert Server v 1.0 ou Rational Team Concert para System z Server v 1.0.1 fornece suporte seletivo para algumas funções do Developer para System z, como projetos locais. Recomendamos o Rational Team Concert para System z Server v 2.0.0.2 para uma experiência mais integrada e com todos os recursos

## File Manager

Para suportar a integração do File Manager, um dos seguintes níveis deve ser instalado no host:

Número do Programa	Nome do Produto	PTFs ou Níveis de Serviço Necessários
5655-U29	IBM File Manager para z/OS v 10.1	• UK54428

O Web site do produto relacionado é:

<http://www.ibm.com/software/awdtools/filemanager/>

## Fault Analyzer

Para suportar a integração do Fault Analyzer, os seguintes níveis devem estar instalados no host:

Número do Programa	Nome do Produto	PTFs ou Níveis de Serviço Necessários
5655-V51	IBM Fault Analyzer v 10.1	Nenhum PTF ou Nível de Serviço necessário
5655-U28	IBM Fault Analyzer v 9.1	Nenhum PTF ou Nível de Serviço necessário
5655-S15	IBM Fault Analyzer v 8.1	Nenhum PTF ou Nível de Serviço necessário

O Web site do produto relacionado é:

<http://www.ibm.com/software/awdtools/faultanalyzer/>

## REXX

Para usar o SCLM Developer Toolkit, um dos seguintes níveis deve estar instalado no host:

Número do Programa	Nome do Produto	PTFs ou Níveis de Serviço Necessários
5695-014	IBM Library para REXX no zSeries v 1.4	Nenhum PTF ou Nível de Serviço necessário
5695-014	IBM Library para REXX no zSeries Alternate Library v 1.4.0 (FMIDs HWJ9143, JWJ9144)	Nenhum PTF ou Nível de Serviço necessário

Uma versão do REXX/370 Alternate Library está disponível no Web site do produto:

<http://www.ibm.com/software/awdtools/rexx/rexxzseries/>

## Ported tools

O IBM Ported Tools para z/OS deve ser instalado (no z/OS UNIX) para utilizar o sftp ou scp para realizar uma implementação segura no SCLM Developer Toolkit.

Uma versão do IBM Ported Tools para z/OS está disponível no Web site do produto:

[http://www-03.ibm.com/servers/eserver/zseries/zos/unix/port\\_tools.html](http://www-03.ibm.com/servers/eserver/zseries/zos/unix/port_tools.html)

## **Ant**

O Apache Ant deve ser instalado (no z/OS UNIX) para executar construções JAVA/J2EE no SCLM Developer Toolkit.

O Apache Ant é uma ferramenta de construção de software livre, baseada em Java, que você pode transferir por download no Web site do produto:

<http://ant.apache.org/>

## **Endevor®**

O CA Endevor® Software Change Manager Release 12 deve ser instalado para o uso da Interface do Developer para System z para CA Endevor® SCM.

CA Endevor® SCM é um produto do CA. O Web site do produto relacionado é:

<http://www.ca.com/us/products/product.aspx?ID=259>



---

## Bibliografia

---

### Publicações Referenciadas

As publicações a seguir são referenciadas neste documento:

*Tabela 52. Publicações Referenciadas*

<b>Título da publicação</b>	<b>Número do pedido</b>	<b>Referência</b>	<b>Web site de referência</b>
Java Diagnostic Guide	SC34-6650	Java 5.0	<a href="http://www.ibm.com/developerworks/java/jdk/diagnosis/">http://www.ibm.com/developerworks/java/jdk/diagnosis/</a>
Guia do Usuário do Java SDK and Runtime Environment	/	Java 5.0	<a href="http://www-03.ibm.com/servers/eserver/zseries/software/java/">http://www-03.ibm.com/servers/eserver/zseries/software/java/</a>
Program Directory for IBM Rational Developer for System z	GI11-8298	Developer para System z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/">http://www-306.ibm.com/software/awdtools/rdz/library/</a>
Rational Developer for System z Common Access Repository Manager Developer's Guide	SC23-7660	Developer para System z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/">http://www-306.ibm.com/software/awdtools/rdz/library/</a>
Pré-requisitos do Rational Developer para System z	S517-9092	Developer para System z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/">http://www-306.ibm.com/software/awdtools/rdz/library/</a>
Iniciação Rápida da Configuração do Host do Rational Developer para System z	GI11-9201	Developer para System z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/">http://www-306.ibm.com/software/awdtools/rdz/library/</a>
Rational Developer for System z Host Planning Guide	GI11-8296	Developer para System z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/">http://www-306.ibm.com/software/awdtools/rdz/library/</a>
SCLM Developer Toolkit: Guia do Administrador	SC23-9801	Developer para System z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/">http://www-306.ibm.com/software/awdtools/rdz/library/</a>
APPC and WebSphere Developer for System z	SC23-5885	Whitepaper	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/">http://www-306.ibm.com/software/awdtools/rdz/library/</a>
Communications Server IP Configuration Guide	SC31-8775	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Communications Server IP Configuration Reference	SC31-8776	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Communications Server IP Diagnosis Guide	GC31-8782	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Communications Server IP System Administrator's Commands	SC31-8781	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Communications Server SNA Network Implementation Guide	SC31-8777	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Communications Server SNA Operations	SC31-8779	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Cryptographic Services System SSL Programming	SC24-5901	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>

Tabela 52. Publicações Referenciadas (continuação)

<b>Título da publicação</b>	<b>Número do pedido</b>	<b>Referência</b>	<b>Web site de referência</b>
DFSMS Macro Instructions for Data Sets	SC26-7408	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
DFSMS Using Data Sets	SC26-7410	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Language Environment Customization	SA22-7564	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Language Environment Debugging Guide	GA22-7560	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
MVS Initialization and Tuning Guide	SA22-7591	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
MVS Initialization and Tuning Reference	SA22-7592	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
MVS JCL Reference	SA22-7597	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
MVS Planning APPC/MVS Management	SA22-7599	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
MVS Planning Workload Management	SA22-7602	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
MVS System Commands	SA22-7627	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Security Server RACF Command Language Reference	SA22-7687	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Security Server RACF Security Administrator's Guide	SA22-7683	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Customização de TSO/E	SA22-7783	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
TSO/E REXX Reference	SA22-7790	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
UNIX System Services Command Reference	SA22-7802	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
UNIX System Services Planning	GA22-7800	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
UNIX System Services User's Guide	SA22-7801	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Utilizando os Serviços de Sistemas REXX e z/OS UNIX	SA22-7806	z/OS 1.9	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Resource Definition Guide	SC34-6430	CICSTS 3.1	<a href="http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html">http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html</a>
Resource Definition Guide	SC34-6815	CICSTS 3.2	<a href="http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html">http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html</a>
Resource Definition Guide	SC34-7000	CICSTS 4.1	<a href="https://publib.boulder.ibm.com/infocenter/cicsts/v4r1/index.jsp?topic=/com.ibm.cics.ts.home.doc/library/library_html.html">https://publib.boulder.ibm.com/infocenter/cicsts/v4r1/index.jsp?topic=/com.ibm.cics.ts.home.doc/library/library_html.html</a>



*Tabela 52. Publicações Referenciadas (continuação)*

<b>Título da publicação</b>	<b>Número do pedido</b>	<b>Referência</b>	<b>Web site de referência</b>
RACF Security Guide	SC34-6454	CICSTS 3.1	<a href="http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html">http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html</a>
RACF Security Guide	SC34-6835	CICSTS 3.2	<a href="http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html">http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html</a>
RACF Security Guide	SC34-7003	CICSTS 4.1	<a href="https://publib.boulder.ibm.com/infocenter/cicsts/v4r1/index.jsp?topic=/com.ibm.cics.ts.home.doc/library/library_html.html">https://publib.boulder.ibm.com/infocenter/cicsts/v4r1/index.jsp?topic=/com.ibm.cics.ts.home.doc/library/library_html.html</a>
Referência de Linguagem	SC27-1408	Enterprise COBOL para z/OS	<a href="http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html">http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html</a>

Os Web sites a seguir são referidos neste documento:

*Tabela 53. Web Sites Referidos*

<b>Descrição</b>	<b>Web site de referência</b>
Centro de Informações do Developer para System z	<a href="http://publib.boulder.ibm.com/infocenter/ratdevz/v7r6/index.jsp">http://publib.boulder.ibm.com/infocenter/ratdevz/v7r6/index.jsp</a>
Suporte do Developer para System z	<a href="http://www-306.ibm.com/software/awdtools/rdz/support/">http://www-306.ibm.com/software/awdtools/rdz/support/</a>
Biblioteca do Developer para System z	<a href="http://www-306.ibm.com/software/awdtools/rdz/library/">http://www-306.ibm.com/software/awdtools/rdz/library/</a>
Página Inicial do Developer para System z	<a href="http://www-306.ibm.com/software/awdtools/rdz/">http://www-306.ibm.com/software/awdtools/rdz/</a>
Serviço recomendado para o Developer para System z	<a href="http://www-01.ibm.com/support/docview.wss?rs=2294&amp;context=SS2QJ2&amp;uid=swg27006335">http://www-01.ibm.com/support/docview.wss?rs=2294&amp;context=SS2QJ2&amp;uid=swg27006335</a>
Pedido de aprimoramento para o Developer para System z	<a href="https://www.ibm.com/developerworks/support/rational/rfe/">https://www.ibm.com/developerworks/support/rational/rfe/</a>
Biblioteca do z/OS na Internet	<a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
Centro de Informações do CICSTS	<a href="https://publib.boulder.ibm.com/infocenter/cicsts/v4r1/index.jsp">https://publib.boulder.ibm.com/infocenter/cicsts/v4r1/index.jsp</a>
Download do Apache Ant	<a href="http://ant.apache.org/">http://ant.apache.org/</a>
Documentação do keytool Java	<a href="http://java.sun.com/j2se/1.5.0/docs/tooldocs/solaris/keytool.html">http://java.sun.com/j2se/1.5.0/docs/tooldocs/solaris/keytool.html</a>
Página inicial de suporte de CA	<a href="https://support.ca.com/">https://support.ca.com/</a>

## Publicações Informativas

As publicações a seguir podem ser úteis para você compreender os problemas de configuração para os componentes do host necessários:

*Tabela 54. Publicações Informativas*

<b>Título da publicação</b>	<b>Número do pedido</b>	<b>Referência</b>	<b>Web site de referência</b>
ABCs do z/OS System Programming Volume 9 (z/OS UNIX)	SG24-6989	Redbook	<a href="http://www.redbooks.ibm.com/">http://www.redbooks.ibm.com/</a>
Guia do Programador de Sistema' para: Workload Manager	SG24-6472	Redbook	<a href="http://www.redbooks.ibm.com/">http://www.redbooks.ibm.com/</a>

*Tabela 54. Publicações Informativas (continuação)*

<b>Título da publicação</b>	<b>Número do pedido</b>	<b>Referência</b>	<b>Web site de referência</b>
Implementação do TCPIP Volume 1: Funções Base, Conectividade e Roteamento	SG24-7532	Redbook	<a href="http://www.redbooks.ibm.com/">http://www.redbooks.ibm.com/</a>
TCPIP Implementation Volume 3: High Availability, Scalability, and Performance	SG24-7534	Redbook	<a href="http://www.redbooks.ibm.com/">http://www.redbooks.ibm.com/</a>
TCP/IP Implementation Volume 4: Security and Policy-Based Networking	SG24-7535	Redbook	<a href="http://www.redbooks.ibm.com/">http://www.redbooks.ibm.com/</a>

---

## 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 usado 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 usada 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. No J2EE, uma entidade que fornece serviços de gerenciamento de ciclo de vida, segurança, implementação e 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 usado 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 usada 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 usado 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 usam 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 de elementos XML que pertencem a documentos XML e a grupos COBOL que não são idênticos na forma (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 ambiente de trabalho. O usuário do ambiente de trabalho 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 semelhantes a 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 em equipe, um repositório compartilhado permite o acesso de multiusuários aos 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 usados 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.

**Sidedeck.** 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 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 Ambiente de Trabalho.

**Visualização Repositórios.** Exibe os locais de repositório CVS que foram incluídos no Ambiente de Trabalho.

**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.

**Visualização Console de Saída.** Exibe a saída de um processo e permite fornecer entrada do teclado a um processo.



---

# Notas de Documentação para IBM Rational Developer para System z

© Copyright IBM Corporation - 2010

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

Gerência de Relações Comerciais e Industriais da IBM Brasil  
Av. Pasteur, 138-146  
Botafogo  
Rio de Janeiro, RJ  
CEP 22290-240

Para pedidos de licenças com relação a informações sobre 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:

Licença de Propriedade Intelectual  
Lei de Propriedade Legal e Intelectual  
IBM Japan, Ltd.  
3-2-12, Roppongi, Minato-ku, Tokyo 106-8711 Japan

O parágrafo a seguir não se aplica a nenhum país em que tais disposições não estejam de acordo 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, AS GARANTIAS IMPLÍCITAS DE NÃO-INFRAÇÃO, COMERCIALIZAÇÃO OU 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. São feitas alterações periódicas 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 a qualquer momento sem aviso prévio.

Referências nestas informações a Web sites que não sejam da IBM são fornecidas apenas por conveniência e não representam de forma alguma um endosso a estes 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 Rational Software  
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 o 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 nível de desenvolvimento e não há garantia de que estas medidas serão as mesmas em sistemas disponíveis em geral. Além disso, algumas medidas podem ter sido estimadas através de extrapolação. Os resultados reais podem ser diferentes. Os usuários deste documento devem verificar os dados aplicáveis para o 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 nem qualquer outra reivindicação relacionada a produtos não-IBM. Dúvidas sobre os recursos de produtos não-IBM devem ser encaminhadas diretamente a seus fornecedores.

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 usados 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 usados por uma empresa comercial real é mera coincidência.

---

## Licença de Copyright

Essas informações contêm programas de exemplos aplicativos na linguagem fonte, ilustrando as técnicas de programação em diversas plataformas operacionais. O Cliente 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. Os programas de



amostra são fornecidos "NO ESTADO EM QUE SE ENCONTRAM", sem garantia de qualquer tipo. A IBM não será responsabilizada por qualquer dano decorrente do uso dos programas de amostra.

---

## Reconhecimentos de Marca Registrada

IBM, o logotipo IBM e [ibm.com](http://ibm.com) são marcas ou marcas registradas da International Business Machines Corp., registradas em várias jurisdições no mundo todo. Outros nomes de produtos e serviços podem ser marcas registradas da IBM ou de outras empresas. Uma lista atual das marcas registradas da IBM está disponível na Web em [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

Rational é uma marca registrada da International Business Machines Corporation e da Rational Software Corporation nos Estados Unidos e/ou em outros países.

Intel® e Pentium® são marcas registradas da Intel Corporation nos Estados Unidos e/ou em outros países.

Microsoft®, Windows® e o logotipo Windows são marcas ou marcas registradas da Microsoft Corporation nos Estados Unidos e/ou em outros países.

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.

UNIX é uma marca registrada da The Open Group nos Estados Unidos e/ou em outros países.



---

# Índice Remissivo

## Caracteres Especiais

.dstoreMemLogging 136  
.dstoreTrace 136  
/etc/inittab, inicialização, INEDT do z/OS UNIX 319  
/etc/rc, inicialização do INETD do z/OS UNIX 319  
\_RSE\_CMDSERV\_OPTS, Definindo parâmetros extras de inicialização Java com 46  
\_RSE\_JAVAOPTS, Definindo parâmetros extras de inicialização Java com 42  
\_RSE\_PORTRANGE 159

## A

acesso às bibliotecas do sistema, Aprimorar o 243  
Acesso condicional a arquivos em spool 167  
ações baseadas em Host remoto, subprojetos z/OS UNIX 101  
Ações condicionais em tarefas 164  
Ações nas tarefas - limitações de execução 166  
ADM (Application Deployment Manager) 189  
ADM, customização 71  
admin, SCLMDT 85  
administradores que não são do sistema, privilégios de atualização 199  
administrative utility, notas de migração 259  
administrative utility administradores CICS funções fornecidas 255  
Administrative Utility do CICS 72  
ADNCSDAR, não-primária RESTful e não-primária serviço da Web 74  
ADNCSDRS, primária RESTful 73  
ADNCSDTX, ID de alteração de RESTful 74  
ADNCSDWS, primária do serviço da Web 76  
ADNJSPAU, Administrative utility 255  
ADNJSPAU, Administrative Utility do CICS 72  
ADNMSGHC, manipulador de mensagens do pipeline 75  
ADNMSGHS, manipulador de mensagens do pipeline 75  
ADNTXNC, ID de alteração de RESTful 74  
ADNVCRD, repositório do CRD 72  
ADNVMFST, repositório de manifesto 77  
ajustando considerações 211  
Ajustar CRASERV.properties 58  
Ajustar ISPF.conf 59  
alterações, PROCLIB 88  
alterações, Workload Manager 87  
alterações do APPC, Ativando 327  
alterações do PROCLIB 88  
alterações do Workload Manager 87  
Alterações entre a versão 7.0 e a versão 7.1 289  
ambiente de linguagem, Bibliotecas de tempo de execução do 244  
ambiente TSO, Customizando 263  
ambiente UNIX, Ordens de procura usadas no 309  
ambiente z/OS UNIX, Ordens de procura usadas no 309  
Análise de log e de configuração usando FEKLOGS 135  
análise de uso, armazenamento de amostra 224  
anel de chave com o RACE, Criar um 295  
Ant  
    co-requisitos 339  
Ant, Instalar e customizar 84  
Apache Ant  
    co-requisitos 339  
APF, autorização 147  
aplicativo, Desenvolvimento do 244  
aplicativos suportados 3  
APPC, Configurando 323  
APPCPMxx 325  
Application Deployment Manager, customização 71  
Application Deployment Manager, customizando 251  
Application Deployment Manager, editor de Definição de Recurso CICS 251  
Application Deployment Manager, servidor de Definição de Recurso CICS 251  
Aprimorando o desempenho da verificação de segurança 245  
Aprimorar o acesso às bibliotecas do sistema 243  
armazenamento de chaves com keytool, Criar um 304  
armazenamento de uso, análise de uso 224  
Arquivo de configuração de criação de log, rsecomm.properties 94  
arquivo de configuração do Client Gateway, TSO/ISPF 47  
arquivo de configuração do JES Job Monitor, FEJJCNCFG, 27  
arquivo de configuração do RSE, rsed.envvars, 31  
arquivo de configuração do TSO/ISPF Client Gateway 47  
arquivo temporário INETD /etc/inetd.pid 319  
arquivos configuráveis 282, 290

arquivos controlados pelo programa UNIX para RSE, Definir 186  
arquivos controlados pelo programa z/OS UNIX para RSE, Definir 186  
Arquivos de Base da Configuração do Resolvedor 309  
arquivos de configuração, Developer para System z 173  
Arquivos de configuração, níveis de software idênticos, diferentes 272  
arquivos de configuração, Resolvedor de base 309  
arquivos de configuração diferentes com níveis de software idênticos 272  
Arquivos de dump 141  
arquivos de log  
    .dstoreMemLogging 136  
    .dstoreTrace 136  
    audit.log 136  
    fa.log 136  
    fekfivpi.log 136  
    fekfivps.log 136  
    ffs.log 136  
    ffsget.log 136  
    ffsput.log 136  
    lock.log 136  
    rmt\_class\_loader.cache.jar 136  
    rsecomm.log 136  
    rsedaemon.log 136  
    rseserver.log 136  
    serverlogs.count 136  
    stderr.log 136  
    stdout.log 136  
arquivos de log do conjunto de encadeamentos RSE  
    audit.log 138  
    rsedaemon.log 138  
    rseserver.log 138  
    serverlogs.count 138  
    stderr\*.log 138  
    stdout\*.log 138  
arquivos de log do daemon RSE  
    audit.log 138  
    rsedaemon.log 138  
    rseserver.log 138  
    serverlogs.count 138  
    stderr\*.log 138  
    stdout\*.log 138  
arquivos em spool, Acesso condicional a 167  
arquivos ISPF.conf, utilizar com várias configurações 265  
arquivos previamente configurados, fazendo backup  
    Versão 7.0 277  
    Versão 7.1 277  
    Versão 7.5 277  
ASCHPMxx 326  
MAX 235  
ASSIZEMAX 181

- Ativando o CA Endevor® Software Configuration Manager Repository Access Manager 61
- Ativando o Common Access Repository Manager 49
- Ativando o PDS Repository Access Manager 61
- Ativando o Repository Access Manager de Base 61
- Ativando o SCLM Repository Access Manager 61
- Ativando Repository Access Managers (RAMs) de amostra 60
- ativar compartilhamento de classe, Java Virtual Machines (JVMs) 246
- atributo do sistema de arquivos, SETUID 146
- atributo do sistema de arquivos SETUID 146
- atualizações SCLM para SCLMDT 85
- audit.log 136
- autenticação, Configurando SSL e X.509 293
- autenticação, JES Job Monitor 158
- Autenticação de cliente usando certificados X.509 168
- Autenticação do JES Job Monitor 158
- autenticação pelo daemon RSE 171
- autenticação pelo software de segurança 170
- autenticação x.509, configurando 293
- Autorização APF FEK.SFEKAUTH 178
- Autorização de controle de programa 146

## B

- banco de dados de chaves com gskkyman, Criar um 301
- Bibliotecas, tempo de execução, ambiente de linguagem 244
- bibliotecas controladas para RSE, Definir MVS 184
- bibliotecas controladas pelo programa MVS para RSE , Definir 184
- Bibliotecas de tempo de execução do ambiente de linguagem 244
- bibliotecas do sistema, Aprimorar o acesso às 243
- bibliotecas para RSE , Definir MVS 184
- bits de permissão, z/OS UNIX 146
- Bits de permissão do z/OS UNIX 146
- BPXBATCH, inicialização do INETD 320
- BPXPRMxx 241
  - INADDRANYCOUNT 234
  - MAXASSIZE 150, 181, 233
  - MAXFILEPROC 233
  - MAXMMAPAREA 233
  - MAXPROCSYS 153, 232
  - MAXPROCUSER 153, 232
  - MAXSOCKETS 234
  - MAXTHREADS 232
  - MAXTHREADTASKS 232
  - MAXUIDS 154, 233
- BPXPRMxx, configurar limites em MAXASSIZE 16

- BPXPRMxx, configurar limites em (continuação)
- MAXPROCUSER 16
- MAXTHREADS 16
- MAXTHREADTASKS 16

## C

- CA Endevor® SCM RAM 61
- CA Endevor® SCM RAM, customizando 67
- Caracteres, não alfanuméricos no diagrama de sintaxe 133
- Caracteres não alfanuméricos, diagrama de sintaxe 133
- Caracteres não editáveis, customizando para tratar 99
- Caracteres não editáveis, uchars.settings 99
- carga de trabalho, Gerenciamento de 245
- CARMA, ativando 49
- CARMA, Criação de logs do 140
- CARMA, FMID HCMA710 290
- CARMA, Inicialização alternativa do servidor 58
- CARMA, interface RSE para 51
- CEE.SCEELPA SYS1.PARMLIB(LPALSTxx) 244
- certificado, X.509 158
- certificado assinado, autoassinado ou assinado pela Autoridade de Certificação 296
- Certificado X.509 158
- certificado X.509 e terceira parte 158
- certificados, autenticação de cliente usando X.509 168
- certificados X.509, usando autenticação de cliente 168
- Certificate Revocation List (CRL), consultando rsed.envvars 170
- variáveis de ambiente do CRL 170
- chaves privadas e certificados, decidir onde armazenar 293
- CICS, regiões de conexão não-primária do 74
- CICS Transaction Server co-requisitos 335
- CICSplex SM Business Application Services (BAS) 252
- classes de segurança, Ativar configurações e 176
- classificação de carga de trabalho, WLM 201
- CLASSPATH 272
- Client Gateway, Usando o método de acesso do TSO/ISPF 264
- clonando configuração RSE existente 296
- co-requisitos Ant 339
- Apache Ant 339
- CICS Transaction Server 335
- Compilador COBOL 334
- Compilador PL/I 334
- DB2 337

- co-requisitos (continuação)
- Fault Analyzer 338
- Ferramenta de Depuração 335
- Ferramentas portadas 338
- File Manager 338
- host 331
- IMS 336
- Java 2 Technology Edition 331
- Rational Team Concert para System z 337
- REXX 338
- SDK para z/OS 331
- z/OS 332
- co-requisitos de host 331
- co-requisitos do z/OS Assembler de Alto Nível 333
- C/C++ 333
- Language Environment 334
- SCLM 333
- COBOL verificação remota 145
- coexistência, atualizar rsed.envvars para ativar a coexistência 297
- comando Modify (F) 125
- Comando Start (S) 123
- comando Stop (P) 129
- comandos, JES Job Monitor Modify 125
- comandos, JES Job Monitor Start 123
- comandos, Modify (F) 125
- comandos, Modify do daemon de bloqueio 129
- comandos, Modify do daemon RSE 126
- comandos, Start (S) 123
- comandos, Start do daemon de bloqueio 124
- comandos, Start do daemon RSE 124
- comandos, Stop (P) 129
- Comandos do operador 123
- Comandos ISPF 48, 59
- Comandos TSO 59
- COMMNDxx, incluir tarefas iniciadas 17
- Common Access Repository Manager, Ativando 49
- Common Access Repository Manager (CARMA), FMID HCMA710 290
- compartilhamento de classe, ativando em Java Virtual Machines (JVMs) 246
- compartilhamento de classe entre Java Virtual Machines (JVMs) 246
- Compilador COBOL co-requisitos 334
- Compilador PL/I co-requisitos 334
- componentes, Instalando e configurando 3
- Componentes, opcionais 48
- componentes de host, Instalando e configurando 3
- Componentes opcionais 48
- comunicação, criptografada por SSL 167, 254
- comunicação, Externa 160
- comunicação criptografada, SSL 167, 173, 254
- comunicação criptografada por SSL 167, 173, 254

Comunicação Externa 160  
 comunicação externa para portas  
   especificadas, limitando 159  
 Comunicação interna 161  
 conexão, daemon de bloqueio 113  
 conexão, Segurança de 159  
 conexão da configuração do host Secure  
   Socket Layer, Testar 298  
 conexão da configuração do host SSL,  
   Testar 298  
 conexão do daemon de bloqueio 113  
 conexão do daemon RSE 113  
 Conexão do Host, Emulador de 155  
 conexão do ISPF do TSO/ISPF Client  
   Gateway do ISPF, verificando 114  
 conexão do JES Job Monitor 113  
 conexão do serviço TSO Commands  
   (APPC) 115  
 conexão não-primária do CICS, regiões  
   de 74  
 conexão recusada 153  
 Conexão recusada 153  
 conexão REXEC, verificando 118  
 conexão SCLM Developer Toolkit,  
   verificando  
   verificações do SCLMDT 116  
 conexão SCLMDT, verificando 116  
 configuração, customização 15  
 configuração, idêntica em um  
   sysplex 271  
 configuração de amostra 239  
   contagem do conjunto de  
   encadeamentos 239  
   definindo limites 240  
   determinando limites mínimos 240  
 configuração de customização 15  
 configuração de produtos e software  
   obrigatórios 9  
 configuração de rastreo,  
   rsecomm.properties 94  
 configuração de rastreo RSE,  
   rsecomm.properties, 94  
 configuração do APPC seguro,  
   VTAM 328  
 configuração do JES Job Monitor  
   GEN\_CONSOLE\_NAME 166  
 configuração do REXEC 101  
 configuração do SSH 101  
 configuração do TCP/IP, netstat 112  
 Configuração idêntica em um  
   sysplex 271  
 configuração RSE existente, Clonar  
   a 296  
 configurações ambientais, INETD 322  
 configurações ambientais do INETD 322  
 configurações de segurança,  
   verificar 186  
 configurações e classes de segurança,  
   Ativar 176  
 configurando componentes de host 3  
 configurando objetivos, WLM 203  
 conjuntos de dados MVS, Falha na  
   abertura de 154  
 considerações, pré-configuração 8  
 considerações, pré-implementação 11  
 Considerações, pré-instalação 4  
 considerações, Segurança 157  
 considerações, servidor 10  
 considerações CICSTS 251  
 Considerações de ID do usuário 9  
 considerações de pré-configuração 8  
 Considerações de segurança 157  
 considerações de uso, APPC 105  
 considerações do servidor 10  
 Considerações sobre Desempenho 243  
 considerações sobre migração 277  
 Considerações sobre migração 3  
 considerações sobre pré-  
   implementação 11  
 Considerações sobre pré-instalação 4  
 considerações sobre uso de APPC 105  
 considerações WLM xvi, 201  
 consultar uma Certificate Revocation List  
   (CRL)  
   rsed.envvars 170  
   variáveis de ambiente do CRL 170  
 Contagem de encadeamentos 219  
 Contagem de processos 216  
 Contagem do espaço de endereço 213  
 controle de auditoria  
   \_RSE\_HOST\_CODEPAGE 163  
   audit.\* options 163  
   daemon.log 163  
   enable.audit.log 163  
 correquisitos, Developer para System  
   z 329  
 CRA#ASLM, SCLM RAM de amostra 61  
 CRA#CIRX, IRXJCL versus CRAXJCL 69  
 CRA#CRAM, RAM de base de  
   amostra 61  
 CRA#UADD, componentes do  
   CARMA 50, 62  
 CRA#UQRY, componentes do  
   CARMA 50, 62  
 CRA#VCAD, Endeavor SCM RAM 62  
 CRA#VCAS, Endeavor SCM RAM 62  
 CRA\$VDEF, componentes do  
   CARMA 50  
 CRA\$VMSG, componentes do  
   CARMA 50, 62  
 CRA#VPDS, PDS RAM de amostra 61  
 CRA#VSLM, SCLM RAM de amostra 61  
 CRA\$VSTR, componentes do  
   CARMA 50  
 CRAISPRX, arquivo de configuração do  
   ISPF 47  
 CRAISPRX, inicialização alternativa do  
   CARMA do ISPF 59  
 CRANDVRA, customizando 66  
 CRASERV.properties, Ajustar 58  
 CRASRV.properties  
   clist.dsname 52  
   crastart.configuration.file 52  
   crastart.steplib 52  
   crastart.stub 52  
   crastart.syslog 52  
   crastart.tasklib 52  
   crastart.timeout 52  
   port.range 52  
   port.start 52  
   startup.script.name 52  
 CRASRV.properties, ajustando 63, 65  
 CRASRV.properties, ajustar 54, 56  
 CRASTART, usar inicialização do  
   servidor CARMA alternativa 55  
 crastart.conf, ajustar 56  
 crastart.conf, iniciação crastart do  
   CARMA 56  
 crastart.endevor.conf, ajustando 66  
 CRASUBCA, ajustando 63  
 CRASUBMT, inicialização em lote do  
   CARMA 54  
 CRAXJCL, IRXJCL versus CRAXJCL 69  
 CRD, repositório do 72  
 criação de log, conjunto de  
   encadeamento 137  
 criação de log, daemon de bloqueio 137  
 criação de log, daemon RSE 137  
 Criação de log, Fault Analyzer  
   Integration 139  
 criação de log, SCLM Developer  
   Toolkit 140  
 Criação de log, teste IVP do fekfivpi 141  
 criação de log de auditoria, gerenciado  
   pelo daemon RSE 163  
 criação de log de auditoria e daemon  
   RSE 163  
 criação de log de instalação de recurso,  
   CICS 253  
 Criação de log de instalação de recurso  
   do CICS 253  
 Criação de log de integração do File  
   Manager  
   rsecomm.log 139  
 Criação de log de teste, IVP do  
   fekfivpi 141  
 Criação de log de teste IVP  
   fekfivpi.log 141  
   fekfivps.log 141  
 Criação de log de teste IVP do fekfivpi  
   fekfivpi.log 141  
 criação de log do conjunto de  
   encadeamento 137  
 criação de log do daemon de  
   bloqueio 137  
 criação de log do daemon RSE 137  
 Criação de log do Fault Analyzer  
   Integration  
   fa.log 139  
   rsecomm.log 139  
 criação de log do SCLM Developer  
   Toolkit  
   rsecomm.log 140  
 criação de logs, JES Job Monitor 137  
 criação de logs, transações APPC 140  
 criação de logs, usuário do RSE 138  
 criação de logs de transações APPC 140  
 Criação de logs do CARMA  
   rsecomm.log 140  
 Criação de logs do Common Access  
   Repository Manager 140  
 criação de logs do JES Job Monitor 137  
 criação de logs do serviço TSO  
   Commands 140  
 criação de logs do usuário, RSE 138  
 criação de logs do usuário do RSE  
   .dstoreMemLogging 138  
   .dstoreTrace 138  
   ffs.log 138  
   ffsget.log 138

- criação de logs do usuário do RSE  
(*continuação*)
  - ffspout.log 138
  - lock.log 138
  - rmt\_class\_loader.cache.jar 138
  - rsecomm.log 138
  - stderr.log 138
  - stdout.log 138
- criando LSTRANS.FILE 82
- criptografia, ssl.properties 91
- Criptografia de comunicação usando SSL 159
- criptografia SSL do RSE, ssl.properties,
  - propriedade de servidor 91
  - propriedades do daemon 91
- cron, limpeza do diretório WORKAREA 105
- customização - ISPF.conf, 264
- customização, básica 15
- customização, SCLM Developer Toolkit 79
- customização, tarefas opcionais de 87
- customização básica 15
- Customizando o Ambiente TSO 263
- customizando o Application Deployment Manager 251
- customizar os IDs de transação do servidor CRD 74

## D

- dados de auditoria
  - log de ações 163
- daemon, Bloqueio 23
- daemon, RSE 22
- daemon de bloqueio
  - mensagens do console 130
- Daemon de bloqueio 23, 196
- daemon de bloqueio, comando Modify 129
- daemon de bloqueio, comando Start 124
- daemon de bloqueio, LOCKD 107
- daemon do REXEC, permissões dos IDs dos usuários quando iniciado pelo INETD 322
- daemon do RSE, comando Modify 126
- daemon do RSE, comando Start 124
- daemon do SSH, permissões dos IDs dos usuários quando iniciado pelo INETD 322
- Daemon em bloqueio (LOCKD) 189
- Daemon RSE 10, 22, 160
  - mensagens do console 130
- daemon RSE, autenticação pelo 171
- daemon RSE, conexão do 113
- daemon RSE, RSED 107
- daemon RSE (RSED) 189
- DB2
  - co-requisitos 337
- definições, pré-requisitos de LINKLIST e LPA 20
- definições, Segurança 26, 175
- definições de porta, PROFILE.TCPIP 318
- definições de recurso, várias 234
- definições de recurso chave 231
  - rsed.envvars 231
  - SYS1.PARMLIB(BPXPRMxx) 231

- definições de recursos CICS, administrador 251
- definições de recursos CICS, desenvolvedor 251
- Definições de segurança 26, 175
- definições de segurança, Lista de verificação 175
- Definições disponíveis para o resolvidor 312
- definições LINKLIST, pré-requisitos 20
- Definições locais disponíveis para o resolvidor 312
- definições LPA, pré-requisitos 20
- Definir arquivos controlados pelo programa z/OS UNIX para RSE 186
- Definir bibliotecas controladas pelo programa MVS para RSE 184
- Definir servidor RSE como um z/OS UNIX seguro 183
- Definir suporte PassTicket para RSE 185
- Definir verificação de Port Of Entry para RSE 172
- Dependência do nome do host 307
- desempenho, Considerações sobre 243
- desempenho da verificação de segurança, Aprimorando o 245
- Desenvolvimento de Aplicativos 244
- Developer para System z, entendimento 189
- Developer para System z, visão geral do componente
  - representação gráfica 189
- diagrama de sintaxe, Caracteres não-alfanuméricos e espaços em branco 133
- diagrama de sintaxe, Como ler 132
- diagrama de sintaxe, Mais longo que uma linha 133
- diagrama de sintaxe, operandos 132
- diagrama de sintaxe, Seleccionando mais de um operando 133
- diagrama de sintaxe, símbolos 132
- Disponibilidade, porta 111
- Disponibilidade de porta 111
- donos das tarefas 192
- Dumps de Java 141
- Dumps do MVS 141

## E

- Editor de Definição de Recurso CICS (CRD), Application Deployment Manager 251
- ELAXMJCL, mudanças no DB2 88
- ELAXMSAM 88
- Emulador de Conexão do Host 155
- endereço do host não resolvido, TCP/IP Resolver
  - lock.log 313
- entendimento do Developer para System z 189
- Enterprise Service Tools, suporte 89
- erro, Rastreio de feedback de 145
- espaço de endereço, Tamanho do 150
- Espaço em disco, Java Virtual Machines (JVMs) 248
- espaços, diagrama de sintaxe 133

- espaços em branco, diagrama de sintaxe 133
- específicas, limitar comunicação externa para portas 159
- EST, suporte 89
- estimativa de tamanho, diretrizes 223
- estrutura do diretório, z/OS UNIX
  - representação gráfica 198
- estrutura do diretório z/OS UNIX
  - representação gráfica 198
- ETC.SERVICES
  - aliases 316
  - port\_number 316
  - protocolo 316
  - service\_name 316
- exec de alocação (JCL de transação APPC), Usando 267
- exec de alocação, usando 265
- Execução Remota, usando 100
- Executando várias instâncias 271
- Explorador de Sistemas Remotos 10

## F

- fa.log 136
- Falha na abertura de conjuntos de dados MVS 154
- falha na abertura de conjuntos de dados MVS 154
- Fault Analyzer
  - co-requisitos 338
- fazendo backup dos arquivos configurados
  - Versão 7.0 277
  - Versão 7.1 277
  - Versão 7.5 277
- feedback de erro, Rastreio de 145
- FEJJCNFG 161, 241, 273
  - CONSOLE\_NAME 166
  - MAX\_THREADS 234
- FEJJCNFG, arquivo de configuração do JES Job Monitor 27
- FEJJCNFG, JES Job Monitor 173
- FEJJJCL, alterações de PROCLIB, JES Job Monitor 21
- FEJTSO 27
- FEKAPPC, implementação de APPC 104
- FEKAPPC, implementação de APPC 104
- FEKAPPCX, implementação de APPC 104
- FEKAPPL 158
- fekfivpi.log 136
- fekfivpi.log, criação de log de teste IVP 141
- fekfivps.log, criação de log de teste IVP 141
- FEKFRSRV 105
- FEKLOCKD 23
- FEKLOGS, análise de log e de configuração usando 135
- FEKRACF, definições de segurança 26, 175
- fekrivp 147
- FEKRSED 22
- FEKSETUP 15, 73, 80, 104



- Ferramenta de Depuração
  - co-requisitos 335
- Ferramenta de Depuração IBM,
  - co-requisito 20
- Ferramentas portadas
  - co-requisitos 338
- ffs.log 136
- ffsget.log 136
- ffsput.log 136
- File Manager
  - co-requisitos 338
- Fluxo de conexão 194
  - representação gráfica 194
- Fluxo do daemon de bloqueio
  - representação gráfica 196
- FMID HCMA710 290
- FMID HHOP710 289
- FMID HHOP750 286
- FMID HHOP760 280
- FMIEXT.properties
  - ativado 99
  - fmlistenport 99
- Fragmentos, sintaxe 133
- Fragmentos de sintaxe 133

## G

- Gerenciamento de carga de trabalho 245
- grupos de propriedades, baseados no
  - host 96
- grupos de propriedades baseados no
  - host 96
- gskkyman, Criar um banco de dados de
  - chaves com 301

## H

- HCMA710 290
- heap Java fixo, Tamanho de 245
- HHOP710 289
- HHOP750 286
- HHOP760 280
- host
  - pré-requisitos 329
- host do z/OS
  - pré-requisitos 329

## I

- ID do usuário, Considerações do 9
- ID do Usuário e Senha 158
- ID do usuário e senha única 158
- IDs de transação do servidor CRD,
  - customizando 74
- IEASYSxx 241
  - MAXUSER 154, 235
- IMS
  - co-requisitos 336
- INETD, configurando 315
- inetd.conf 315
  - protocolo 315
  - server\_program 315
  - server\_program\_arguments 315
  - service\_name 315
  - socket\_type 315
  - userid 315

- inetd.conf (*continuação*)
  - wait\_flag 315
- informações de configuração, ordens de
  - procura das 308
- iniciação rápida, opção Java
  - (-Xquickstart) 246
- inicialização, IVP 110
- inicialização, Requisitos da JCL de 150
- inicialização, z/OS UNIX INETD 319
- Inicialização alternativa do servidor
  - CARMA 58
- Inicialização CA Endeavor® SCM RAM,
  - submissão em lote 63
- inicialização CARMA, submissão em
  - lote 53
- Inicialização do CA Endeavor® SCM RAM,
  - CRASTART 65
- Inicialização do CRASTART, inicialização
  - do CA Endeavor® SCM RAM
  - usando 65
- inicialização do IVP
  - ivpinit 110
- Instalação do SMP/E, sticky bit 148
- Instalando e configurando componentes
  - de host 3
- integração, File Manager 98
- Integração do File Manager 98
- integração do File Manager, Criação de
  - log de 139
- interface, RESTful versus serviço da
  - Web 73
- interface de serviço da Web 252
  - ADMI 75
  - ADMR 75
  - ADMS 75
- Interface de Serviço da Web, servidor
  - CRD usando 75
- Interface de Serviço da Web versus
  - interface RESTful 73
- interface RESTful 252
  - ADMI 74
  - ADMR 74
  - ADMS 74
- Interface RESTful, servidor CRD
  - usando 73
- interface RESTful versus interface de
  - serviço da Web 73, 252
- Interface RSE para CARMA 51
- interna, Comunicação 161
- IRXJCL, IRXJCL versus CRAXJCL 69
- IRZ de diagnóstico, mensagens de
  - erro 91
- ISP.SISPLoad
  - ISPF TSO/ISPF Client Gateway 184
- ISPF, Usar vários execs de alocação 265
- ISPF.conf 47
  - allocjob 47
  - ISPF\_timeout 47
  - isplib 47
  - ispmlib 47
  - ispplib 47
  - ispslib 47
  - isptlib 47
  - sysproc 47
- ISPF.conf, Ajustar 59
- ISPF.conf, Customização básica 264

- ISPF.conf para SCLMDT, Atualizações
  - do 80
- ISPF TSO/ISPF Client Gateway
  - ISP.SISPLoad 184
- IVP
  - fekfivpa 115
  - fekfivpd 113
  - fekfivpi 114
  - fekfivpj 113
  - fekfivpl 113
  - fekfivpr 118
  - fekfivps 116
  - fekfivpz 119
- IVP, serviços básicos e opcionais
  - fekfivpa 109
  - fekfivpd 109
  - fekfivpi 109
  - fekfivpj 109
  - fekfivpl 109
  - fekfivpr 109
  - fekfivps 109
  - fekfivpt 109
  - fekfivpz 109
  - scripts IVP 109
- IVTPRMxx
  - ECSA MAX 235
  - FIXED MAX 235

## J

- J2EE 79, 84
- Java 79, 84
- Java, Dumps de 141
- Java 2 Technology Edition
  - co-requisitos 331
- JAVA\_DUMP\_TDUMP\_PATTERN 142
- Java Virtual Machines (JVMs),
  - compartilhamento de classe entre 246
- JCL de inicialização, Requisitos de 150
- JCL de transação APPC, Customização
  - básica 266
- JCL de transação APPC, Usando um exec
  - de alocação 267
- JCL de transação APPC, Usar perfis do
  - ISPF existentes com 267
- JES JMON
  - GEN\_CONSOLE\_NAME 166
- JES JMON, FEJJCNF
  - \_BPXK\_SETIBMOPT
  - \_TRANSPORT 28
- APPLID 28
- AUTHMETHOD 28
- CODEPAGE 28
- CONCHAR 28
- CONSOLE\_NAME 28
- GEN\_CONSOLE\_NAME 28
- HOST\_CODEPAGE 28
- LIMIT\_COMMANDS 28
- LIMIT\_VIEW 28
- LISTEN\_QUEUE\_LENGTH 28
- MAX\_DATASETS 28
- MAX\_THREADS 28
- SERV\_PORT 28
- SUBMITMETHOD 28
- TIMEOUT 28
- TIMEOUT\_INTERVAL 28
- TSO\_TEMPLATE 28

JES JMON, FEJJC�FG (continuação)  
TZ 28

JES Job Monitor, comando Modify 125

JES Job Monitor, comando Start 123

JES Job Monitor, conexão do 113

JES Job Monitor, FEJJC�FG 173

JES Job Monitor, JMON 107

JES Job Monitor (JMON) 189

JMON 21, 182, 273

fekfvpj 113

JMON, JES Job Monitor 107

JVMs, compartilhamento de classe  
entre 246

## K

keytool, Criar um armazenamento de  
chaves com 304

## L

ler um diagrama de sintaxe, Como 132  
liberando um bloqueio

RSE, comando cancelar modify 197

LIMIT\_COMMANDS 165

LIMIT\_VIEW 167

limitações de execução, Ações nas  
tarefas 166

limitando comunicação externa, portas  
específicas 159

limite de tamanho, espaço de  
endereço 223

limite de tamanho, heap Java 222

limite de tamanho do espaço de  
endereço 223

limite de tamanho de heap, Java 222

limite do tamanho do heap Java 222

limites, z/OS UNIX configurados em  
BPXPRMxx 16

limites de tamanho do cache, Java Virtual  
Machines (JVMs) 247

Limites do sistema 153

limites z/OS UNIX configurados em  
BPXPRMxx 16

limpeza do diretório, WORKAREA  
skulker 105

limpeza do diretório WORKAREA  
skulker 105

LINKLIST, definições para outros  
produtos 21

lista de verificação, administrador  
SCLM 85

lista de verificação, cliente 13

lista de verificação, Cliente 12

lista de verificação, requisitos 15

lista de verificação, transação APPC 103

lista de verificação de requisitos 15

lista de verificação de transação,  
APPC 103

Lista de verificação de transações  
APPC 103

Lista de Verificação do Administrador de  
SCLM 85

Lista de verificação do cliente 12, 13

lista de verificação do qualificador,  
ELAXF\* 26

Lista de Verificação para Qualificadores  
de Alto Nível ELAXF\* 26

Locais do dump, z/OS UNIX 143

Locais do dump do UNIX 143

Locais do dump do z/OS UNIX 143

lock.log 136

LOCKD 10

LOCKD, daemon de bloqueio 107

LPALSTxx 244

LPALSTxx, definições LPA em 18

LSTRANS.FILE, criando 82

## M

Manipulador de mensagens do  
pipeline 75

mensagens, administrative utility 260

mensagens, console 130

mensagens de erro, Diagnóstico de  
IRZ 91

mensagens do administrative utility 260

mensagens do console 130

JES Job Monitor 130

Mensagens do console

daemon de bloqueio 130

Daemon RSE 130

servidor do conjunto de

encadeamentos RSE 130

mensagens do pipeline, Manipulador  
de 75

método de acesso do APPC, Usando 266

método de acesso do TSO/ISPF Client  
Gateway, Usando o 264

método de conexão RSE, alternativo 101

método de conexão RSE alternativo 101

métodos, Autenticação 157

Métodos de acesso, TSO 263

Métodos de acesso do TSO 263

Métodos de autenticação 157

migração 277

migração, versão 7.5 para 7.6 280

monitorando, rede 239

monitorando o RSE 236

monitorando z/OS sistema de arquivo  
UNIX 239

monitorando z/OS UNIX 237

mudanças, DB2 88

mudanças no arquivo de configuração,  
outros produtos 277

mudanças no DB2 88

MVS, Dumps do 141

## N

netstat 149

netstat, configuração do TCP/IP 112

níveis de software idênticos com arquivos  
de configuração diferentes 272

nível de software, idênticos em arquivos  
de configuração diferentes 272

Nome alternativo da LU do parceiro,  
serviço do TSO Commands 328

nome de transação, alternativo para  
FEKFRSRV por serviço do TSO  
Commands 328

nome do host, Dependência do 307

nomes de host, aplicando no Developer  
para System z 311

notas de migração, administrative  
utility 259

## O

objetivos, configurando em WLM 203

obrigatórios, configuração de produtos e  
software 9

onde armazenar chaves privadas e  
certificados 293

opção Xquickstart Java 246

opções de configuração alternativa, APPC  
e VTAM 328

opções de configuração alternativa do  
APPC 328

opções de configuração alternativa do  
VTAM 328

operador, comandos do 123

operando, Selecionando mais de um em  
um diagrama de sintaxe 133

Operandos, diagrama de sintaxe 132

ordem de procura, ambiente nativo  
MVS 317

ordem de procura, ambiente z/OS  
UNIX 317

Ordens de procura, ambiente z/OS  
UNIX 309

ordens de procura das informações de  
configuração 308

## P

parâmetros de inicialização com  
\_RSE\_CMDSERV\_OPTS, Definindo Java  
extra 46

parâmetros de inicialização com  
\_RSE\_JAVAOPTS, Definindo Java  
extra 42

parâmetros de inicialização Java com  
\_RSE\_CMDSERV\_OPTS, Definindo 46

parâmetros de inicialização Java com  
\_RSE\_JAVAOPTS, Definindo 42

PARMLIB, alterações 16

PassTickets, usando 162

PDS Repository Access Manager,  
Ativando 61

perfil de segurança, Limitações  
armazenadas no 151

perfis, Definir conjunto de dados 178

perfis do conjunto de dados, Definir 178  
perfis do ISPF (JCL de transação  
APPC) 267

perfis do ISPF, Usar existentes 264

perfis do ISPF existentes (JCL de  
transação APPC), Usar 267

perspectiva de projetos z/OS 97  
pipeline, Manipulador de mensagens  
do 75

plataformas suportadas 3

portas, CARMA e TCP/IP 162

portas, REXEC 101

portas, TCP/IP 160

portas do CARMA e TCP/IP 162

Portas TCP/IP 160



portas TCP/IP, representação gráfica 160  
 Portas TCP/IP reservadas 149  
 PORTRANGE 41, 149  
 PORTRANGE disponível para RSE, Definindo 41  
 pré-requisitos  
   host 329  
   SMP/E 330  
 pré-requisitos, Developer para System z 329  
 Pré-requisitos de Host do z/OS 329  
 pré-requisitos do host 329  
 privilégios de atualização, administradores que não são do sistema 199  
 problemas de configuração, Resolução de problemas 135  
 problemas de requisito, conhecidos 154  
 procedimento armazenado, DB2 87  
 procedimento armazenado do DB2 87  
 procedimentos de construção remota, ELAXF\* 25  
 procedimentos de construção remota ELAXF\* 25  
 procedimentos ELAXF\*, Amostra  
   ELAXFADT 25  
   ELAXFASM 25  
   ELAXFBMS 25  
   ELAXFCOC 25  
   ELAXFCOP 25  
   ELAXFCOT 25  
   ELAXFCP1 25  
   ELAXFCPC 25  
   ELAXFCPP 25  
   ELAXFDCL 25  
   ELAXFGO 25  
   ELAXFLNK 25  
   ELAXFMFS 25  
   ELAXFPL1 25  
   ELAXFPLP 25  
   ELAXFPLT 25  
   ELAXFPP1 25  
   ELAXFTSO 25  
   ELAXFUOP 25  
 PROCLIB, alterações 21  
 produtos, obrigatórios 5  
 produtos e software obrigatórios, configuração de 9  
 produtos obrigatórios 5  
 PROFILE.TCPIP definições de porta 318  
 programas de verificação de instalação 109  
 PROGxx, autorizações APF 18  
 PROGxx, definições LINKLIST 19  
 projectcfg.properties 97  
   PROJECT-HOME 98  
   WSED-VERSION 98  
 projetos, baseados no host 97  
 projetos baseados no host 97  
 propertiescfg.properties 96  
   DEFAULT-VALUES 97  
   ENABLED 97  
   PROPERTY-GROUP 97  
   RDZ-VERSION 97  
 proteção de aplicativo para RSE, Definir 185

Publicações Referenciadas 341

## R

RACF  
   permite 179  
 RACF, Criar um anel de chave com o 295  
 rastreo 143  
 rastreo, CARMA 144  
 rastreo, daemon de bloqueio 144  
 rastreo, JES Job Monitor 143  
 rastreo, RSE 94, 143  
 rastreo CARMA 144  
 rastreo de daemon de bloqueio 144  
 Rastreo de feedback de erro 145  
 rastreo do JES Job Monitor 143  
 rastreo RSE 143  
 Rational Team Concert para System z  
   co-requisitos 337  
 recurso do CICS, criação de log de instalação de 253  
 recursos, necessários 5  
 recursos necessários 5  
 rede, monitorando 239  
 Referenciadas, publicações 341  
 região de conexão não primária do CICS 76  
 região de conexão primária, CICS 73  
 região de conexão primária do CICS 73, 76  
 regiões de conexão, primária versus não primária 252  
 regiões de conexão não-primária do CICS 74  
 regiões de conexão primária versus não primária 252  
 regras de classificação, WLM 202  
 regras de classificação WLM 202  
 Remover arquivos antigos de WORKAREA 86  
 repositório, manifesto 77  
 repositório de manifesto 77  
 repositório do CRD 72, 173  
 Repository Access Manager 61  
 Repository Access Manager, Ativando de base 61  
 Repository Access Manager, ativando o CA Endevor® SCM 61  
 Repository Access Manager, Ativando o PDS 61  
 Repository Access Manager, Ativando o SCLM 61  
 Repository Access Manager de base, Ativando 61  
 Repository Access Managers (RAMs) de amostra, Ativando 60  
 Requisitos, Developer para System z 329  
 Requisitos da JCL de inicialização 150  
 requisitos de acesso 9  
 requisitos do INETD, Developer para System z 322  
 reservadas, Portas TCP/IP 149  
 Resolução de problemas de configuração 135  
 resolvidor, Definições locais disponíveis para o 312

resolvedores, Compreendendo os 308  
 REXEC, conexão 118  
 REXEC, usando 100  
 REXX  
   co-requisitos 338  
 rmt\_class\_loader\_cache.jar 136  
 RSE 10  
 RSE , Definir bibliotecas controladas pelo programa MVS para 184  
 RSE , Definir suporte PassTicket para 185  
 RSE , Definir verificação de Port Of Entry para 172  
 RSE, Definindo o PORTRANGE 41  
 RSE, Definir arquivos controlados pelo programa z/OS UNIX para 186  
 RSE, Definir como um servidor z/OS UNIX seguro 183  
 RSE, definir proteção de aplicativo para 185  
 RSE, monitorando 236  
 RSE, rsed.envvars  
   \_RSE\_JAVAOPTS 174  
 RSE, ssl.properties 174  
 RSE como um aplicativo Java  
   representação gráfica 191  
 rsecomm.log 136  
   Criação de log de integração do File Manager 139  
   criação de log do SCLM Developer Toolkit 140  
 rsecomm.properties 144  
   debug\_level 94  
   log\_location 94  
   server.version 94  
 rsecomm.properties, 94  
 RSED 22  
 RSED, daemon RSE 107  
 rsed.envvars 126, 230, 272  
   \_BPX\_SHAREAS 39  
   \_BPX\_SPAWN\_SCRIPT 39  
   \_BPXK\_SETIBMOPT  
     \_TRANSPORT 38  
   \_CEE\_DMPTARG 36  
   \_CEE\_RUNOPTS 39  
   \_CMDSERV\_BASE\_HOME 37  
   \_CMDSERV\_CONF\_HOME 37, 266  
   \_CMDSERV\_WORK\_HOME 37  
   \_FEKFS\_CMD\_PARTNER\_LU\_ 38  
   \_FEKFS\_CMD\_TP\_NAME\_ 38  
   &ISPROF=&SYSUID.ISPPROF= 47  
   \_RSE\_CMDSERV\_OPTS 39  
   \_RSE\_DAEMON\_CLASS 39  
   \_RSE\_HOST\_CODEPAGE 36  
   \_RSE\_JAVAOPTS 39, 141, 263  
   \_RSE\_LOCKD\_CLASS 39  
   \_RSE\_LOCKD\_PORT 36  
   \_RSE\_POOL\_SERVER\_CLASS 39  
   \_RSE\_PORTRANGE 38, 159  
   \_RSE\_SERVER\_CLASS 39  
   \_RSE\_SERVER\_TIMEOUT 39  
   \_SCLMDT\_TRANTABLE 38  
 ANT\_HOME 38  
 CGI\_DTWORK 39  
 CLASSPATH 39  
 DAPPLID 45  
 Daudit.cycle 44

rsed.envvars (*continuação*)

- Daudit.retention.period 44
- Ddaemon.log 42
- Ddeny.nonzero.port 45
- DDENY\_PASSWORD 45
- DDSTORE\_IDLE\_SHUTDOWN\_TIMEOUT 45
- DDSTORE\_LOG\_DIRECTORY 42
- DDSTORE\_MEMLOGGING\_ON 45
- DDSTORE\_TRACING\_ON 45
- Denable.audit.log 44
- Denable.automount 44
- Denable.certificate.mapping 44
- Denable.port.of.entry 44
- Denable.standard.log 44
- DHIDE\_ZOS\_UNIX 45
- Dipv6 44
- Dkeep.last.log 44
- Dmaximum.clients 42, 231
- Dmaximum.threadpool.process 44, 231
- Dmaximum.threads 42, 231
- Dminimum.threadpool.process 42, 231
- Dprocess.cleanup.interval 45
- Dsingle.logon 45
- DSTORE\_LOG\_DIRECTORY 140, 143
- DTSO\_SERVER 45
- Duser.log 42
- GSK\_CRL\_SECURITY\_LEVEL 38
- GSK\_LDAP\_PASSWORD 38
- GSK\_LDAP\_PORT 38
- GSK\_LDAP\_SERVER 38
- GSK\_LDAP\_USER 38
- JAVA\_HOME 36
- JAVA\_PROPAGATE 39
- LANG 36
- LIBPATH 39
- PATH 36, 39
- RSE\_HOME 36
- RSE\_JAVAOPTS 37
- RSE\_LIB 39
- RSE\_SAF\_CLASS 37
- SCLMDT\_BASE\_HOME 39
- SCLMDT\_CONF\_HOME 38
- SCLMDT\_WORK\_HOME 39
- STEPLIB 36, 37, 38, 168
- TZ 36
- Xms 42, 231
- Xmx 42, 231

rsed.envvars, arquivo de configuração do RSE 31

rsed.envvars, atualizações para tradução de nome longo/curto 84

rsed.envvars, atualizar para ativar a coexistência 297

rsed.envvars para SCLMDT, Atualizações do 81

rsedaemon.log 136

rseserver.log 136

## S

saídas do sistema, Limitações importas por 151

SCLM 84

SCLM, Tradução de nome longo/abreviado do 82

SCLM Developer Toolkit 184

SCLM Developer Toolkit, customização 79

SCLM Developer Toolkit (SCLMDT) 189

SCLM Repository Access Manager, Ativando 61

SCLMDT, Atualizações do ISPF.conf para 80

SCLMDT, Atualizações do rsed.envvars para 81

SCLMDT admin 85

script de shell, REXEC/SSH 119

script de shell REXEC/SSH 119

script de shell SSH 119

SDSF 80

Secure Shell, usando 100

Secure Socket Layer, Configurando 293

Secure Socket Layer, Criptografia de comunicação usando 159

segmento, Definir OMVS 177

segmento OMVS, Definir 177

segurança, Application Deployment Manager (ADM) 253

segurança, CICSTS 172

segurança, Definir comando do JES 182

segurança, INETD 321

segurança, JES 164

segurança, pipeline 253

segurança, recurso 255

segurança, SCLM 173

segurança, transação 253

segurança da LU, VTAM 328

segurança de comando, Definir JES 182

Segurança de comando do JES, Definir 182

Segurança de Conexão 159

segurança de encadeamento no servidor RSE

PassTickets 162

Segurança de Pipeline 253

Segurança de SCLM 173

segurança de transação 253

segurança do Application Deployment Manager 253

segurança do cache, Java Virtual Machines (JVMs) 247

segurança do CICSTS 172

segurança do INETD 321

Segurança do JES 164

segurança do recurso 255

segurança do repositório, CRD 253

segurança do repositório CRD 253

senha e ID do usuário 158

senha única e ID do usuário 158

serverlogs.count 136

Serviço SCLM Developer Toolkit 19

Serviço TSO Command 189

serviço TSO Commands 152, 263

serviço TSO Commands, transação APPC para 102

servidor CARMA, Inicialização alternativa do 50, 55, 58

Servidor CRD usando a interface RESTful 73

Servidor de Definição de Recurso CICS (CRD), Application Deployment Manager 251

servidor do conjunto de encadeamentos RSE

mensagens do console 130

servidor do JES Job Monitor 21

servidor do Job Monitor, JES 21

servidor RSE 160

servidor UNIX, Definir RSE como 183

servidor z/OS UNIX, Definir RSE como 183

servidor z/OS UNIX seguro, Definir RSE como um 183

sessão shell, inicialização do INETD 320

shell script do ivpinit, configurando variáveis de ambiente RSE com 110

Símbolos, diagrama de sintaxe 132

Sintaxe, exemplo 133

sistema, Aprimorar o acesso às bibliotecas do 243

sistema, Limites do 153

sistemas de arquivo zFS, Usando 243

sistemas de arquivos, zFS 243

sistemas operacionais suportados 3

skulker, limpeza do diretório WORKAREA 105

SMP/E

pré-requisitos 330

Software Configuration Manager 61

software de segurança, autenticação pelo 170

software obrigatórios, configuração de produtos e 9

spool, Acesso condicional a arquivos em 167

SSH, usando 100

SSL, Configurando 293

SSL, Criptografia de comunicação usando 159

ssl.properties 91

- daemon\_key\_label 93
- daemon\_keydb\_file 93
- daemon\_keydb\_password 93
- enable\_ssl 93
- server\_keystore\_file 93
- server\_keystore\_label 93
- server\_keystore\_password 93
- server\_keystore\_type 93

ssl.properties, ativar o SSL criando um novo daemon RSE 297

ssl.properties, Ativar SSL atualizando 297

stderr.\*.log 136

stderr.log 136

stdout.\*.log 136

stdout.log 136

STEPLIB, Evite o uso de 243

sticky bit, disponibilidade do módulo de carregamento MVS para z/OS UNIX 148

submissão em lote, inicialização do CA Endeavor® SCM RAM usando 63

submissão em lote, usar inicialização do servidor CARMA 53

subprojetos z/OS UNIX, ações baseadas em host Remoto para 101

- subsistemas suportados 3
- suporte, Enterprise Service Tools 89
- suporte, EST 89
- Suporte à linguagem bidirecional 90
- suporte à linguagem bidirecional do CICS 90
- suporte de autenticação de cliente, incluir X.509 301
- suporte para RSE, Definir PassTicket 185
- suporte PassTicket para RSE , Definir 185
- SYS1.PARMLIB(APPCPMxx) 325
- SYS1.PARMLIB(ASCHPMxx) 326
- SYS1.PARMLIB(BPXPRMxx) 241
  - MAXASSIZE 150, 181
  - MAXPROCSYS 153
  - MAXPROCUSER 153
  - MAXUIDS 154
- SYS1.PARMLIB(BPXPRMxx), Java Virtual Machines (JVMs) 247
- SYS1.PARMLIB(BPXPRMxx), Limitações definidas em 150
- SYS1.PARMLIB(IEASYSxx) 241
  - MAXUSER 154
- SYSEXEC 47
- sysplex, configuração idêntica em 271
- SYSPROC 47

## T

- tabelas, Conversão 309
- Tabelas, host local 310
- Tabelas de Conversão 309
- Tabelas do host, local 310
- Tabelas do Host Local 310
- Tamanho de heap Java fixo 245
- Tamanho do espaço de endereço 150
- tarefa iniciada do JES Job Monitor, JMON 21
- tarefa iniciada do RSE, RSED 22
- tarefas, Ações condicionais em 164
- tarefas iniciadas, Definir para o Developer para System z
  - tarefas iniciadas JMON 180
  - tarefas iniciadas LOCKD 180
  - tarefas iniciadas RSED 180
- tarefas iniciadas, verificando 107
- tarefas iniciadas, verificando serviços 109
- tarefas iniciadas do Developer para System z, Definir 180
- tarefas opcionais, customização de 87
- tarefas opcionais de customização 87
- TCP/IP, aplicando no Developer para System z 311
- TCP/IP, Configurando 307
- TCP/IP, Definições locais disponíveis para o resolvidor 312
- TCP/IP reservadas, Portas 149
- TCP/IP Resolver, endereço do host não resolvido
  - lock.log 313
- Testar a conexão da configuração do host SSL 298
- tipos de subsistema
  - ASCH 202

- tipos de subsistema (*continuação*)
  - CICS 202
  - JES 202
  - OMVS 202
  - STC 202
- TN3270 334
- Tradução de nome, SCLM 82
- Tradução de nome longo/abreviado, SCLM 82
- tradução de nome longo/curto, atualizações para rsed.envvars 84
- Transação APPC
  - resolução de problemas do APPC 152
- transação APPC, Implementação 104
- transação APPC, preparação 102
- Transação APPC e serviço TSO
  - Commands 152
- transação APPC para o serviço TSO
  - Commands 102
- transação do serviço TSO Commands, Definindo a 327
- transações APPC do Developer para System z, configurações com várias 268
- transações do CICS 173
- TSO Commands (APPC), conexão do serviço 115
- TSO/ISPF, customização - ISPF.conf, 264
- TSO/ISPF, Usando um exec de alocação 265
- TSO/ISPF, Usar perfis do ISPF existentes 264
- TSO/ISPF, Usar vários execs de alocação 265
- TSO/ISPF, utilizar com várias configurações 265
- TSO/ISPF Client Gateway, Usando o método de acesso do 264

## U

- uchars.settings, caracteres não editáveis 99
- usando os PassTickets 162
- usando SSL, Criptografia de comunicação 159
- Usando um exec de alocação 265
- Usar perfis existentes do ISPF 264
- usar submissão em lote para a inicialização do servidor CARMA 53
- Use STEPLIB 20
- uso de armazenamento 222
- uso de espaço, z/OS UNIX sistema de arquivos UNIX 228
- uso de recursos, ajustando 211
- uso de recursos, visão geral 212
- uso de STEPLIB, Evite o 243
- uso do espaço do sistema de arquivos, z/OS UNIX 228
- uso do espaço do sistema de arquivos z/OS UNIX 228
- usuário, Considerações do ID do 9
- Utilitários do gerenciamento do cache, Java Virtual Machines (JVMs) 248

## V

- validação da Autoridade de Certificação
  - gskkyman 169
  - SAF key ring 169
  - TRUST, HIGHTRUST 169
- várias configurações do Developer para System z, utilizar várias transações
  - APPC com 268
- várias configurações do Developer para System z, utilizar vários arquivos
  - ISPF.conf com 265
- Várias definições de recurso 234
  - FEJJCNFG 234
  - Placa EXEC, JLC do servidor 234
  - SYS1.PARMLIB(ASCHPMxx) 235
  - SYS1.PARMLIB(IEASYSxx) 235
  - SYS1.PARMLIB(IVTPRMxx) 235
- várias instâncias, Executando 271
- Várias LUs 328
- Várias transações APPC 268
- variáveis de padrão de dump de transação 142
- variável PARM, limitações no JCL 24
- Vários arquivos ISPF.conf 265
- vários execs de alocação, TSO/ISPF 265
- verificação de instalação 107
- verificação de POE 160, 172
- Verificação de Port of Entry 172
- Verificação de Port Of Entry 160
- verificação de segurança, Aprimorando o desempenho da 245
- verificando conexão do ISPF do TSO/ISPF Client Gateway 114
- verificando shell script do REXEC/SSH 119
- Verificar configurações de segurança 186
- versão 7.0 e a versão 7.1, Alterações entre 289
- Virtual Telecommunications Access Method 324
- visão geral do componente, Developer para System z
  - representação gráfica 189
- VSAM 323
- VTAM 324

## W

- Web Owning Region 252
- WORKAREA, Remover arquivos antigos de 86
- workload manager 201

## X

- X.509, incluindo suporte de autenticação de cliente 301
- Xquickstart, opção Java 246

## Z

- z/OS
  - co-requisitos 332
- z/OS sistemas de arquivo UNIX, monitorando 239



---

## Comentários do Leitor

IBM Rational Developer para System z  
Guia de Configuração do Host  
Versão 7.6.1

Publicação N° S517-9094-04

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 no Brasil

S517-9094-04

