

IBM Rational Developer for z Systems
Versão 9.5.1:

Guia de Configuração



IBM Rational Developer for z Systems
Versão 9.5.1:

Guia de Configuração



Nota

Antes de usar estas informações, certifique-se de ler as informações gerais em “Avisos” na página 155.

Primeira edição (dezembro de 2015)

Esta edição se aplica ao IBM Rational Developer for z Systems Versão 9.5.1 (número de programa 5724-T07 ou parte do número de programa 5697-CDT) e a todas as liberações e modificações subsequentes até que indicado de outra forma 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 por meio de um representante IBM ou da filial da IBM que atende em sua região. As publicações não são guardadas no endereço abaixo.

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

IBM Brasil - Centro de Traduções
Rodovia SP 101 Km 09
Building 501 P.O. Box 12195
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 ou divulgação restritos pelo documento GSA ADP Schedule Contract com a IBM Corp.

© Copyright IBM Corporation 2015, 2015.

Conteúdo

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

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

Sobre este Documento	xi
---------------------------------------	-----------

Quem Deve Usar este Documento	xii
Resumo das Mudanças	xii
Descrição do Conteúdo do Documento	xiv
Planejamento	xiv
Customização Básica	xiv
CARMA (Common Access Repository Manager)	xv
SCLM Developer Toolkit	xv
Depurador Integrado	xv
Análise de código baseada em host	xv
Outras tarefas de customização	xv
Verificação de Instalação	xvi
Definições de segurança	xvi
Guia de Migração	xvi
Referência de Configuração do Host	xvi

IBM Rational Developer for z Systems Host Configuration Guide . 1

Capítulo 1. Planejamento	3
---	----------

Considerações sobre migração	3
Considerações sobre planejamento	3
Visão Geral do Produto	3
Exigências de Capacidade	4
Requisitos de Tempo	4
Considerações sobre pré-instalação	5
ID do Usuário de Instalação	5
Produtos Obrigatórios	5
Recursos Necessários	6
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
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 os Limites do z/OS UNIX em BPXPRMxx	17
Incluir as Tarefas Iniciadas em COMMNDxx	17
Definições SVC no IEASVCxx	18
Ativação do produto em IFAPRDxx	18
Definições de LPA em LPALSTxx	19
Autorizações de APF no PROGxx	19

Definições de LINKLIST no PROGxx	20
Definições de LINKLIST e LPA de Requisito	22
Definições LINKLIST para Outros Produtos	23
Alterações do PROCLIB	23
DBGMR, tarefa iniciada do gerenciador de depuração	23
procedimentos de construção remota do ELAXF*	24
Definições de segurança	26
rdz.env, o arquivo de configuração de ambiente	27
Componentes específicos	30
Verificação de Instalação	31

Capítulo 3. CARMA (Common Access Repository Manager). 33

Requisitos e Lista de Verificação	33
Selecionar o Método de Inicialização do Servidor e o RAM Ativo	34
Inicialização do Servidor CARMA	34
CRASTART	34
Envio em Lote	34
RAMs de Produção	34
CA Endevor SCM RAM	35
RAM de Pacotes do CA Endevor SCM	35
RAMs de Amostra	35
PDS RAM	35
RAM de Base	35
RAM do SCLM	35
RAM Pré-configurado e Combinações de Inicialização do Servidor	35
CRASTART com CA Endevor SCM RAM	35
Criar os Conjuntos de Dados VSAM do CARMA	36
Customizar CRASRV.properties	36
Customizar crastart.endevor.conf	36
(Opcional) Customização Adicional do CA Endevor SCM RAM	37
CRASTART com RAMs de Amostra	38
Criar os Conjuntos de Dados VSAM do CARMA	38
CARMA	38
RAMs de Amostra	38
Customizar CRASRV.properties	38
Customizar crastart.conf	39
Customização da RAM adicional (opcional)	39
Submissão em Lote com CA Endevor SCM RAM	40
Criar os Conjuntos de Dados VSAM do CARMA	40
Customizar CRASRV.properties	40
Customizar CRASUBCA	40
(Opcional) Customização Adicional do CA Endevor SCM RAM	43
Submissão em Lote com RAMs de Amostra	43
Criar os Conjuntos de Dados VSAM	43
CARMA	43
RAMs de Amostra	43
Customizar CRASRV.properties	44
Customizar CRASUBMT	44
Customização da RAM adicional (opcional)	45

Detalhes da Configuração do CARMA	46
CRASRV.properties, a Interface RSE para CARMA	46
crastart*.conf, A Inicialização do Servidor CRASTART	49
Coletando Arquivos de Log CRASTART	52
CRASUB*, a Inicialização do Servidor de Envio em Lote	52
Conjuntos de Dados CARMA VSAM	54
CRADEF, o Conjunto de Dados de Configuração	54
CRAMSG, o Conjunto de Dados de Mensagem.	54
CRASTRS, o Conjunto de Dados de Sequência Customizado	54
Repository Access Managers (RAMs) CARMA..	55
CA Endeavor SCM RAM	55
RAM de Pacotes do CA Endeavor SCM	55
PDS RAM	56
RAM de Base.	56
RAM do SCLM	56
CRACFG, CRASCL, CRASHOW e CRATMAP, os arquivos de configuração do CA Endeavor SCM RAM	57
CRACFG, interação do CA Endeavor SCM RAM com o SCM	57
CRASCL, SCL de modelo de RAM do CA Endeavor SCM	57
Filtros Padrão CRASHOW, CA Endeavor SCM RAM	57
CRATMAP, os Mapeamentos de Extensões de Arquivos do CA Endeavor SCM RAM.	58
CRANDVRA e CRADYNDA, os executáveis de alocação de RAM do CA Endeavor SCM	58
CRANDVRA, o executável de alocação inicial de RAM do CA Endeavor SCM	58
CRADYNDA, o executável de realocação de RAM do CA Endeavor SCM	59
Ações em Lote do CA Endeavor SCM RAM	59
CRABCFG, a Configuração da Ação em Lote do CA Endeavor SCM RAM	60
CRABATCA, a JCL de Ação em Lote do CA Endeavor SCM RAM	61
CRABJOBC, o Cartão JOB de Ação em Lote do CA Endeavor SCM RAM	62
CRAALLOC, o executável de alocação customizado de RAM	62
Códigos de Retorno CARMA	62
(Opcional) Suportando Múltiplos RAMs.	63
Exemplo	63
(Opcional) Executável de Alocação Customizado ..	64
(Opcional) Saída do Usuário CARMA	65
(Opcional) IRXJCL versus CRAXJCL	66
Criar CRAXJCL	67

Capítulo 4. SCLM Developer Toolkit .. 69

Requisitos e Lista de Verificação	69
Pré-requisitos.	70
Atualizações do ISPF.conf para SCLMDT	70
Atualizações do rdz.env para SCLMDT	71
(Opcional) Tradução de Nome Longo/Abreviado..	72

Criar LSTRANS.FILE, o VSAM de Conversão de nome Longo/Curto.	72
Atualizações de rdz.env para tradução de nome longo/curto	74
(Opcional) Instalar e Customizar Ant.	74
atualizações SCLM para SCLMDT	75
Remover Arquivos Antigos de WORKAREA e /tmp	76

Capítulo 5. Depurador Integrado. . . 77

Requisitos e Lista de Verificação	77
Integrated Debugger e COBOL v4.	78
Integrated Debugger e Outros Depuradores Baseados em Ambientes de Linguagens	79
Integrated Debugger e comunicação criptografada	79
Depurador Integrado e SYSPLEX	79
Parâmetros de configuração do Integrated Debugger	80
Atualizações parmlib do Integrated Debugger	80
Atualizações do SVC do Depurador Integrado.	81
Atualizações de TCP/IP do Integrated Debugger ..	81
Atualizações de segurança do Integrated Debugger	82
Atualizações do Language Environment do Integrated Debugger	82
Atualizações de CICS do Integrated Debugger.	83
Atualizações do Depurador Integrado IMS	84
Atualizações de procedimentos armazenados do DB2 do Integrated Debugger	84
Atualizações de ELAXF* do Integrated Debugger..	84

Capítulo 6. Análise de código baseada em host 87

Requisitos e Lista de Verificação	87
Revisão de Código	87
Modificando o Processamento de Revisão de Código	88
Cobertura de Códigos	88
Chamada única de cobertura de código	88
Várias chamadas de cobertura de código	89
Saída da Cobertura de Código	89

Capítulo 7. Outras tarefas de customização 91

include.conf, Inclusões forçadas para assistência de conteúdo C/C++	91
Subprojetos do z/OS UNIX	92
Configuração de REXEC ou SSH	93
Suporte a pré-processador de inclusão	93
Suporte xUnit para Enterprise COBOL e PL/I.	94
Enterprise Service Tools, suporte	95
Suporte para linguagem bidirecional do CICS	95
Mensagens de diagnósticos IRZ para o Enterprise Service Tools	96
FEKRNPLI, pré processador remoto PL/I	97

Capítulo 8. Verificação de Instalação 99

Verificar as Tarefas Iniciadas.	99
DBGMGR, o gerenciador de depuração	99
Verificar os Serviços	99
inicialização do IVP	100
Conexão com o CARMA	100
Conexão SCLMDT.	100

Capítulo 9. Definições de segurança 103

Requisitos e Lista de Verificação	103
Ativar Configurações e Classes de Segurança	104
Definir as Tarefas Iniciadas do Developer for z Systems	104
Definir o Debug Manager como um servidor z/OS UNIX seguro	105
Definir as bibliotecas controladas pelo programa MVS para o Debug Manager	105
Definir acesso ao Integrated Debugger	106
Definir os Perfis do Conjunto de Dados	106
Verificar as Configurações de Segurança	107

Capítulo 10. Guia de Migração . . . 109

Considerações sobre migração	109
Fazendo Backup de Arquivos Configurados Anteriormente	109
Migrar da versão 9.5 para a versão 9.5.1	110
IBM Rational Developer for z Systems, FMID HHOP951	111
IBM Rational Developer for z Systems Host Utilities, FMID HAKG951	118
Arquivos Configuráveis	118
Migre da versão 9.1 para a versão 9.5	119
IBM Rational Developer for z Systems, FMID HHOP950	119
Arquivos Configuráveis	122
Utilitários de Host do IBM Rational Developer for z Systems, FMID HAKG950	127
Arquivos Configuráveis	127
Migrar da versão 9.0 para a versão 9.1	129
IBM Rational Developer for z Systems, FMID HHOP910	129
Arquivos Configuráveis	131

Utilitários de Host IBM Rational Developer for z Systems, FMID HAKG910	138
Arquivos Configuráveis	138
Migrar da Versão 8.5 para a Versão 9.0	139
IBM Rational Developer for z Systems, FMID HHOP900	139
Arquivos Configuráveis	141
IBM Rational Developer for z Systems Host Utilities, FMID HAKG900	147
Arquivos Configuráveis	147

Capítulo 11. Referência de Configuração do Host. 149

Entendendo o Developer for z Systems.	149
Considerações de segurança	149
Considerações de TCP/IP	149
Considerações WLM	149
Considerações Push-to-client	149
considerações CICSTS	149
Configurando o AT-TLS	149

Bibliografia 151

Publicações Referenciadas	151
Publicações Informativas	152

Avisos 155

Informações sobre a interface de programação ..	157
Marcas registradas.	157
Termos e condições para a documentação do produto	157
Licença de Copyright.	158
Reconhecimentos de Marca Registrada	158

Figuras

1. DBGMGR: tarefa iniciada do gerenciador de depuração	24
2. Arquivo de configuração do ambiente rdz.env:	27
3. CRASRV.properties: CRASTART com o CA Endeavor SCM RAM.	36
4. crastart.endeavor.conf: CRASTART com CA Endeavor SCM RAM.	37
5. CRASRV.properties: CRASTART com RAMs de amostra	39
6. crastart.conf: CRASTART com RAMs de amostra	39
7. CRASRV.properties: Envio em lote com o CA Endeavor SCM RAM.	40
8. CRASUBCA: Envio em lote com o CA Endeavor SCM RAM	42
9. CRASRV.properties: Envio em lote com RAMs de amostra	44
10. CRASUBMT: Envio em lote com RAMs de amostra	45
11. CRASRV.properties – Arquivo de Configuração CARMA	46
12. crastart*.conf: Inicialização do servidor CARMA usando CRASTART	51
13. CRASUB*: Inicialização do CARMA usando o envio em lote	53
14. CRACFG - interação do CA Endeavor SCM RAM com o SCM	57
15. Filtros Padrão CRASHOW - CA Endeavor SCM RAM.	57
16. CRATMAP: Filtros padrão do CA Endeavor SCM RAM.	58
17. CRABCFG: Configuração de ação em lote do CA Endeavor SCM RAM	60
18. CRABATCA: JCL de ação em lote do CA Endeavor SCM RAM.	61
19. CRABJOB: Cartão JOB de ação em lote do CA Endeavor SCM RAM	62
20. Atualizações do ISPF.conf para SCLMDT	71
21. Atualizações do rdz.env para SCLMDT	71
22. FLM02LST: JCL de configuração de conversão de nomes longos/abreviados.	73
23. include.conf - Inclusões forçadas para assistente de conteúdo C/C++	92

Tabelas

1.	Recursos Necessários.	6	14.	Atualização dinâmica de SVC do Depurador	
2.	Recursos Opcionais	6		Integrado	81
3.	Administradores necessários para as tarefas		15.	IVPs para Serviços	99
	necessárias	7	16.	Variáveis de configuração de segurança	103
4.	Administradores Necessários para Tarefas		17.	Customizações da versão 9.5.1	113
	Opcionais	7	18.	Customizações da versão 9.5.1 z/OS Explorer	117
5.	Lista de verificação do cliente: Partes		19.	Customizações do Host Utilities Versão 9.0	119
	obrigatórias	12	20.	Customizações da versão 9.5	122
6.	Lista de verificação do cliente: Partes opcionais	12	21.	Customizações do Host Utilities Versão 9.0	128
7.	Corresponder módulos de carregamento com		22.	Customizações da versão 9.1.0	132
	funções	20	23.	Customizações do Host Utilities Versão 9.0	138
8.	Procedimento ELAXF* de amostra	24	24.	Customizações da versão 9.0	141
9.	Lista de Verificação do Qualificador de Alto		25.	Customizações do Host Utilities Versão 9.0	147
	Nível ELAXF*.	25	26.	Publicações Referenciadas	151
10.	ELAXF*	26	27.	Web Sites Referidos	152
11.	Reconexão automática para o Debug Manager	28	28.	Publicações Informativas.	152
12.	Códigos de Retorno CARMA	63			
13.	Lista de Verificação do Administrador de				
	SCLM	75			

Sobre este Documento

Este documento descreve a configuração das funções do IBM® Rational Developer for z Systems. Ele inclui instruções para configurar o IBM Rational Developer for z Systems Versão 9.5.1 em seu sistema host z/OS.

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

- O *IBM Explorer for z/OS* é chamado de *z/OS Explorer*.
- *IBM Rational Developer for z Systems* é chamado *Developer for z Systems*.
- *IBM Rational Developer for z Systems* O *Integrated Debugger* é chamado *Integrated Debugger*.
- O *IBM Rational Developer for z Systems Interface for CA Endevor® SCM* é chamado de *CA Endevor® SCM RAM*.
- *Common Access Repository Manager* é abreviado para *CARMA*.
- *Software Configuration and Library Manager Developer Toolkit* é chamado *SCLM Developer Toolkit*, abreviado como *SCLMDT*.
- O *IBM z/OS Automated Unit Testing Framework* é chamado de *zUnit*.
- O *z/OS UNIX System Services* é chamado de *z/OS UNIX*.
- O *Customer Information Control System Transaction Server* é chamado de *CICSTS*, abreviado para *CICS*.

Este documento faz parte de um conjunto de documentos que descrevem a configuração do sistema host do Developer for z Systems. Cada um desses documentos tem um público alvo específico. Para concluir a configuração do Developer for z Systems, não é necessário ler todos os documentos.

- O *IBM Rational Developer for z Systems Host Configuration Guide* (SC27-8577) descreve em detalhes todas as tarefas de planejamento, tarefas de configuração e opções (incluindo as opcionais) e fornece cenários alternativos.
- O *IBM Rational Developer for z Systems Host Configuration Reference* (SC27-8578) descreve o design do Developer for z Systems e fornece informações básicas para várias tarefas de configuração do Developer for z Systems, componentes do z/OS e outros produtos (tais como WLM e CICS) relacionados ao Developer for z Systems.

As informações nesse documento se aplicam a todos os pacotes do IBM Rational Developer for z Systems Versão 9.5.

Para obter as versões mais atualizadas deste documento, consulte o *IBM Rational Developer for z Systems Host Configuration Guide* (SC23-7658) disponível em <http://www-05.ibm.com/e-business/linkweb/publications/servlet/pbi.wss?CTY=US&FNC=SRX&PBL=SC27-8577>.

Para obter as versões mais atualizadas da documentação completa, incluindo instruções de instalação, White Papers, podcasts e tutoriais, consulte a página da biblioteca do IBM Rational Developer for z Systems website (http://www-01.ibm.com/software/sw-library/en_US/products/Z964267S85716U24/).

Quem Deve Usar este Documento

Este documento destina-se a programadores de sistemas que estão instalando e configurando o IBM Rational Developer for z Systems Versão 9.5.1.

Este documento lista detalhadamente as etapas necessárias para fazer uma configuração completa do produto, incluindo alguns cenários não padrão. Informações básicas que podem ajudá-lo a planejar e executar a configuração podem ser encontradas no *IBM Rational Developer for z Systems Host Configuration Reference* (SC27-8578). Para usar este documento, você deve estar familiarizado com o z/OS UNIX System Services e sistemas host do MVS.

Resumo das Mudanças

Esta seção resume as mudanças para o *IBM Rational Developer for z Systems Versão 9.5.1 Host Configuration Guide*, SC43-2905-00 (atualizado em dezembro de 2015).

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

Novas informações:

- Informações de migração da versão 9.5.1.
- Diretivas novas e removidas em `rdz.env`

Informações removidas:

Na versão 9.5.1, as funções relacionadas ao Monitor de tarefas RSE e foram movidas de IBM Rational Developer for z Systems para outro produto, o IBM Explorer for z/OS. Esse movimento inclui a documentação relacionada.

- Dados específicos de RSE são removidos de todos os capítulos.
- Dados específicos do Monitor de tarefas JES são removidos de todos os capítulos.
- Dados específicos do serviço de comandos TSO são removidos de todos os capítulos.
- Dados de direcionar ao cliente são removidos de todos os capítulos.
- A documentação de comando do operador é removida.

Este documento contém informações que foram fornecidas anteriormente no *IBM Rational Developer for z Systems Versão 9.5 Host Configuration Guide*, SC23-7658-14.

Novas informações:

- Informações de migração da versão 9.5.1.
- Informações de migração da Versão 9.5.
- Diretivas novas e removidas em `rdz.env`.
- Comandos do operador `DEBUG STORAGE` e `SEND`.
- Enviar suporte de mensagens.
- Suporte interativo do Gateway ISPF.
- Pré-processador PL/I.

Informações movidas:

- As informações do Integrated Debugger movidas para um capítulo separado.

Informações removidas:

- O Application Deployment Manager não é mais fornecido, assim todas as informações relacionadas ao Application Deployment Manager são removidas.

Este documento contém informações que foram fornecidas previamente no Guia de Configuração de Host do *IBM Rational Developer for z Systems Versão 9.1.1*, SC43-1622-14.

Novas informações:

- Definição de tarefa iniciada DBGMR alterada.
- Nova configuração de segurança para o Integrated Debugger.
- Novo status rejectLogon para o conjunto de encadeamentos RSE no comando Modificar processo de exibição.
- Diretivas novas e removidas em rdz.env.
- Instruções de configuração alteradas do Integrated Debugger opcional.

Este documento contém informações que foram fornecidas previamente no *Guia de configuração do host do IBM Rational Developer for z Systems Versão 9.1*, SC43-1622-12.

Novas informações:

- Informações de migração da versão 9.1.0.
- Novas opções para a definição da tarefa iniciada RSED.
- Diretivas novas e mudadas em rdz.env.
- Novos comandos do operador.
- Novas opções para o Integrated Debugger.

Este documento contém informações que foram anteriormente fornecidas no *IBM Rational Developer for z Systems Versão 9.0.1 Host Configuration Guide*, SC43-1622-11.

Novas informações:

- Instruções adicionais de configuração para o Integrated Debugger opcional.

Este documento contém informações que foram fornecidas anteriormente no Guia de Configuração do Host *IBM Rational Developer for z Systems Versão 9.0.1*, SC43-1622-10.

Novas informações:

- Informações de migração Versão 9.0.1.
- Novas atualizações opcionais do PARMLIB.
- Nova tarefa iniciada opcional.
- Novos comandos do operador.
- Diretivas novas e mudadas em rdz.env
- Novas diretivas no CRASRV.properties.
- Novo arquivo de configuração para o CARMA CA Endeavor® SCM RAM.
- Suporte para a saída do usuário chamada durante a inicialização do CARMA.

Esse documento contém informações que foram previamente dadas no *IBM Rational Developer for z Systems Versão 9.0 Guia de Configuração do Host*, SC43-1622-09.

Novas informações:

- Informações de migração da versão 9.0.
- Diretivas novas e mudadas em `rdz.env`.
- Diretivas novas e removidas no FEJCNFG.
- Novos membros do ELAXF* PROCLIB.
- Novos comandos do operador JMON e RSED.
- Informações incluídas sobre a análise de código baseada em host.

Informações removidas:

- A tarefa iniciada LOCKD não é mais usada, portanto, todas as informações sobre o daemon de bloqueio serão removidas.
- O procedimento armazenado de amostra do DB2 foi substituído pelos novos processos de construção ELAXF*, de modo que todas as informações sobre o procedimento armazenado DB2 seja removido.
- As informações sobre migração para liberações não suportadas foram removidas.

Este documento contém informações que foram fornecidas anteriormente no *IBM Rational Developer for z Systems Versão 8.5.1: Guia de Configuração do Host*, S517-9094-08.

Novas informações:

- Diretivas novas e mudadas em `rdz.env`.
- Suporte aprimorado para ações em lote do CA Endevor® SCM RAM.
- Requisitos de instalação do ID do usuário.

Esse documento contém informações que foram previamente dadas no *IBM Rational Developer for z Systems Versão 8.5 Guia de Configuração do Host*, SC43-1622-07.

Novas informações:

- Novas diretivas opcionais em FEJCNFG. Consulte FEJCNFG, o arquivo de configuração do Monitor de tarefas JES.
- Novas diretivas opcionais em `rdz.env`.
- Novo arquivo de configuração opcional.
- Novo componente opcional.
- Novo componente opcional.
- Novo componente opcional.
- Comandos do operador novos e aprimorados.

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 for z Systems.

Customização Básica

As etapas de customização a seguir são para uma configuração do Developer for z Systems básica:

- “Configuração da Customização” na página 15
- “Alterações PARMLIB” na página 16

- “Alterações do PROCLIB” na página 23
- “Definições de segurança” na página 26
- “rdz.env, o arquivo de configuração de ambiente” na página 27

CARMA (Common Access Repository Manager)

O Common Access Repository Manager (CARMA) é uma plataforma do servidor para os Repository Access Managers (RAMs). Um RAM é uma Interface de Programação de Aplicativos (API) para um Software Configuration Manager (SCM) que é baseado em um sistema z/OS. Ao agrupar a funcionalidade do SCM em um RAM, uma única API está disponível para um cliente acessar qualquer SCM suportado.

O Developer for z Systems fornece diversos exemplos de RAMs e códigos-fonte pré-construídos para criar seu próprio RAM.

O IBM Rational Developer for z Systems Interface for CA Endevor[®] Software Configuration Manager dá aos clientes do Developer for z Systems acesso direto ao CA Endevor[®] SCM.

SCLM Developer Toolkit

O SCLM Developer Toolkit fornece as ferramentas necessárias para estender os recursos do SCLM para o cliente. O próprio SCLM é um gerenciador de código-fonte baseado em host que está incluído no ISPF.

O SCLM Developer Toolkit possui um plug-in baseado em Eclipse que interage com o SCLM e fornece acesso a todos os processos do SCLM para desenvolvimento de código legado e suporte para desenvolvimento completo de Java[™] e J2EE na estação de trabalho com sincronização para SCLM no mainframe. As atividades de sincronização incluem construção, montagem e implementação do código J2EE a partir do mainframe.

Depurador Integrado

O componente de host do Depurador integrado do Developer for z Systems permite que os clientes depurem vários aplicativos baseados no Language Environment (LE).

Análise de código baseada em host

Semelhante ao cliente do Developer for z Systems, o host do Developer for z Systems suporta a execução das ferramentas de análise de código, que são fornecidas como um produto separado, o IBM Rational Developer for z Systems Host Utilities. Um benefício de executar a análise de código no host é que ele pode ser integrado em seu processamento em lote diário.

As ferramentas de análise de código a seguir estão disponíveis no host:

- Revisão de código: Usando regras com níveis de severidade diferentes, a revisão de código varre o código-fonte e relata violações de regra.
- Cobertura de código: Analise um programa em execução e gere um relatório de linhas que são executadas, comparado com o número total de linhas executáveis.

Outras tarefas de customização

Esta seção combina uma variedade de tarefas opcionais de customização. Para configurar o serviço necessário, siga as instruções na seção apropriada.

Customizações para arquivos de configuração do Developer for z Systems:

- include.conf, Inclusões forçadas para assistente de conteúdo C/C++

Customizações relacionadas ao Developer for z Systems em ou para outros produtos:

- Subprojetos do z/OS UNIX
- Suporte a pré-processador de inclusão
- Suporte xUnit para Enterprise COBOL e PL/I
- Enterprise Service Tools, suporte
- Suporte para linguagem bidirecional do CICS
- Mensagens de diagnósticos IRZ para o Enterprise Service Tools
- FEKRNPLI, pré processador remoto PL/I

Verificação de Instalação

Após concluir a customização do produto, é possível verificar a configuração bem-sucedida dos principais componentes do produto usando os Installation Verification Programs (IVPs) descritos neste capítulo.

Definições de segurança

Esta seção descreve as definições de segurança necessárias e opcionais com os comandos de amostra do RACF.

Guia de Migração

Esta seção destaca as mudanças na instalação e na configuração em comparação com as liberações anteriores do produto. Também fornece algumas diretrizes gerais para migrar para este release.

Referência de Configuração do Host

Esta seção resume as informações no *Rational Developer for z Systems Host Configuration Reference* (SC27-8578).

IBM Rational Developer for z Systems Host Configuration Guide

Capítulo 1. Planejamento

Use as informações deste capítulo e os requisitos de software do Developer for z Systems, incluindo pré-requisitos e correquisitos para planejar a instalação e a implementação do Developer for z Systems. Os seguintes assuntos são descritos:

- “Considerações sobre migração”
- “Considerações sobre planejamento”
- “Considerações sobre pré-instalação” na página 5
- “Considerações de Pré-configuração” na página 8
- “Considerações sobre pré-implementação” na página 11
- “Lista de verificação do cliente” na página 12

Para obter uma lista completa dos requisitos de hardware e software do Developer for z Systems, incluindo pré-requisitos e correquisitos, consulte a guia **Requisitos do Sistema** na página do produto do Developer for z Systems (<http://www-03.ibm.com/software/products/en/developerforsystemz/>).

Considerações sobre migração

O Capítulo 10, “Guia de Migração”, na página 109 descreve as mudanças na instalação e na configuração em comparação com liberações anteriores do produto. Use estas informações para planejar sua migração para a liberação atual do Developer for z Systems.

Nota:

- Na versão 9.5.1, o Developer for z Systems se torna um produto que é instalado na parte superior do IBM Explorer for z/OS. Assegure-se de que tenha uma versão suportada do IBM Developer for z/OS.
- Se você for um usuário anterior do IBM Rational Developer for System z, salve os arquivos customizados relacionados antes de instalar esta versão do IBM Rational Developer for z Systems. Para obter uma visão geral de arquivos que precisaram de customização, consulte Capítulo 10, “Guia de Migração”, na página 109.
- Se você planeja executar várias instâncias do Developer for z Systems, consulte “Executando diversas instâncias” no *IBM Rational Developer for z Systems Referência de Configuração do Host* (SC27-8578).

Considerações sobre planejamento

Visão Geral do Produto

O Developer for z Systems consiste em um cliente, instalado no computador pessoal do usuário, e em um servidor, instalado em um ou mais sistemas host. Ambos o cliente e o host são instalados na parte superior do IBM Explorer for z/OS. Esta documentação contém informações para um sistema host z/OS.

O cliente fornece aos desenvolvedores um ambiente de desenvolvimento baseado no Eclipse que facilita uma interface gráfica uniforme para 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 em tarefas que são iniciadas ad hoc. Estas tarefas permitem que o cliente trabalhe com os vários componentes do sistema host z/OS, tais como conjuntos de dados MVS, comandos TSO, arquivos e comandos do z/OS UNIX, envio de tarefa e saída de tarefa.

O Developer for z Systems aprimora a funcionalidade de acesso básico fornecida pelo z/OS Explorer. O Developer for z Systems pode, por exemplo, interagir com subsistemas e outro software de aplicativo no sistema host, tal como CICS e o Software Configuration Managers (SCMs), se o Developer for z Systems estiver configurado para isso e se esses produtos de correquisito estiverem disponíveis.

Para obter informações sobre o próprio Developer for z Systems, como ele interage com seu sistema e com os produtos de pré-requisito e de correquisito, consulte o *Rational Developer for z Systems Host Configuration Reference (SC27-8578)*. O *IBM Explorer for z/OS Host Configuration Reference (SC27-8438)* fornece informações semelhantes para o z/OS Explorer, que é um requisito para Developer for z Systems.

Para saber mais sobre a funcionalidade que é oferecida pelo Developer for z Systems, consulte o website Developer for z Systems, <http://www-03.ibm.com/software/products/en/developerforsystemz/>, ou seu representante IBM local.

Exigências de Capacidade

As qualificações de SMP/E são necessárias para uma instalação do host do Developer for z Systems.

A configuração do Developer for z Systems requer mais do que as permissões e conhecimento típicos de programação de sistema, portanto, pode ser necessária a assistência de outros. A Tabela 3 na página 7 e a Tabela 4 na página 7 listam os administradores que são necessários para as tarefas de customização necessárias e opcionais.

Requisitos de Tempo

A quantidade de tempo necessário para instalar e configurar os componentes do sistema host do Developer for z Systems depende de vários fatores, como estes:

- A configuração atual do z/OS UNIX e do TCP/IP
- a disponibilidade de software obrigatório e de manutenção
- A disponibilidade de um 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 configuração do sistema host do Developer for z Systems requer de um a dois dias para ser concluído e não inclui a instalação e a configuração do IBM Explorer for z/OS, que é um produto necessário. Esse requisito de tempo é para que uma instalação limpa seja executada por um programador de sistema experiente. Se forem encontrados problemas ou se as qualificações necessárias não estiverem disponíveis, a configuração demorará mais.

Considerações sobre pré-instalação

Para obter instruções detalhadas sobre a instalação do SMP/E do produto, consulte *Program Directory for IBM Rational Developer for z Systems* (GI11-8298).

Os servidores Developer for z Systems são dispostos em um sistema único e não são cientes do SYSPLEX. Se estiver usando os servidores em um SYSPLEX, você deverá assegurar que os dados solicitados pelos usuários finais (conjuntos de dados, saída de tarefa, arquivos z/OS UNIX) estejam disponíveis no sistema em que o Developer for z Systems está instalado. Consulte "Considerações sobre pré-implementação" na página 11 para clonar Developer for z Systems para outros sistemas.

Para executar várias instâncias do Developer for z Systems em um único sistema host, consulte "Executando diversas instâncias" no *Rational Developer for z Systems Referência de Configuração do Host* (SC27-8578) .

O sistema de arquivos (HFS ou zFS) em que o Developer for z Systems está instalado deve ser montado com o bit de permissão SETUID ativado (este é o padrão do sistema). Montar o sistema de arquivos com o parâmetro NOSETUID evita que o Developer for z Systems crie o ambiente de segurança do usuário e rejeite as solicitações de conexão do cliente. O mesmo é verdadeiro para os sistemas de arquivos que hospedam os binários do z/OS Explorer, Java e z/OS UNIX.

ID do Usuário de Instalação

O ID do usuário usado para instalar o Developer for z Systems ou para instalar manutenção deve ter pelo menos os seguintes atributos:

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

Nota: Um tamanho da região grande é necessário para o ID do usuário que executa os Installation Verification Programs (IVPs), porque são executadas funções que requerem uma grande quantidade de memória (como Java). 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 que é alocado para o usuário, com acessos 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 grupo padrão do ID do usuário requer um GID.
- UID=0 ou autorização READ ao perfil BPX.SUPERUSER na classe FACILITY.
- Se os perfis BPX.FILEATTR.APF ou BPX.FILEATTR.PROGCTL estiverem definidos na classe FACILITY, acesso READ para estes perfis.
- Acesso READ, WRITE e EXECUTE ao diretório /tmp (ou um diretório referenciado na variável de ambiente TMPDIR).

Produtos Obrigatórios

Developer for z Systems tem uma lista de softwares obrigatórios que devem estar instalados e operacionais antes de o produto funcionar. Há também uma lista de software de co-requisito para suportar recursos específicos do Developer for z Systems. Estes requisitos devem ser instalados e operacionais no tempo de execução para os recursos correspondentes para funcionar conforme projetado.

Para obter uma lista completa dos requisitos de software do Developer for z Systems, incluindo pré-requisitos e correquisitos, consulte a guia **Requisitos do Sistema** no website do Developer for z Systems (<http://www-03.ibm.com/software/products/en/developerforsystemz/>) .

Planeje antecipadamente para ter esses produtos de requisito disponíveis, uma vez que isso pode levar algum tempo, dependendo das políticas em seu site. Os requisitos chave para uma configuração básica são:

- z/OS 1.12 ou superior
- IBM Explorer for z/OS 3.0
- Liberação de serviço mais recente do Java 6.0 ou superior (31 ou 64 bits)

Recursos Necessários

O Developer for z Systems requer a alocação dos recursos do sistema listados na Tabela 1. Os recursos listados na Tabela 2 são necessários para serviços opcionais. Planeje ter estes recursos disponíveis porque, dependendo das políticas de seu site, pode levar algum tempo para obter o software.

Tabela 1. Recursos Necessários

Recurso	Valor-padrão	Informações
Procedimentos de construção do MVS	ELAXF*	"Alterações do PROCLIB" na página 23

Tabela 2. Recursos Opcionais

Recurso	Valor-padrão	Informações
IPL com CLPA	não aplicável	Capítulo 5, "Depurador Integrado", na página 77
tarefa iniciada	DBGMR	Capítulo 5, "Depurador Integrado", na página 77
conjunto de dados LINKLIST	FEL.SFELAUTH e FEL.SFELLOAD	<ul style="list-style-type: none"> • Capítulo 4, "SCLM Developer Toolkit", na página 69 • Capítulo 5, "Depurador Integrado", na página 77
conjunto de dados LPA	FEL.SFELLPA	<ul style="list-style-type: none"> • "Definições de LPA em LPALSTxx" na página 19 • Capítulo 3, "CARMA (Common Access Repository Manager)", na página 33 • Capítulo 5, "Depurador Integrado", na página 77
Perfis de segurança	AQE.**	• Capítulo 5, "Depurador Integrado", na página 77
intervalo de portas para uso confinado no host	qualquer porta disponível é usada	<ul style="list-style-type: none"> • Capítulo 3, "CARMA (Common Access Repository Manager)", na página 33 • Capítulo 6, "Análise de código baseada em host", na página 87

Tabela 2. Recursos Opcionais (continuação)

Recurso	Valor-padrão	Informações
intervalo de portas para uso confinado no host	5336	Capítulo 5, “Depurador Integrado”, na página 77
porta para comunicação cliente-host	<ul style="list-style-type: none"> 5335 para Depurador Integrado 	<ul style="list-style-type: none"> Capítulo 5, “Depurador Integrado”, na página 77
atualização da CSD do CICS	diversos valores	<ul style="list-style-type: none"> Capítulo 5, “Depurador Integrado”, na página 77
atualização do CICS JCL	<ul style="list-style-type: none"> FEL.SFELLOAD FEL.SFELAUTH 	<ul style="list-style-type: none"> “Suporte para linguagem bidirecional do CICS” na página 95 Capítulo 5, “Depurador Integrado”, na página 77

A configuração do Developer for z Systems requer mais do que as permissões e conhecimento típicos de programação de sistemas; portanto, pode ser necessária a assistência de outros. A Tabela 3 e a Tabela 4 listam os administradores que são 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/A

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/A
Security	<ul style="list-style-type: none"> Definir perfis do conjunto de dados Definir os conjuntos de dados controlados pelo programa Definir permissão para enviar tarefas xxx* Defina perfis CICS para depuração Inclua o certificado para comunicação criptografada Definir as tarefas iniciadas Definir os perfis do servidor z/OS UNIX Definir os perfis para depuração Defina perfis para envio de mensagens 	<ul style="list-style-type: none"> "Considerações de segurança" em <i>Referência de Configuração do Host</i> (SC14-7290)

Tabela 4. Administradores Necessários para Tarefas Opcionais (continuação)

Administrador	Tarefa	Informações
TCP/IP	Definir novas portas TCP/IP	"Portas TCP/IP" no <i>Referência de Configuração do Host</i> (SC27-8578)
SCLM	<ul style="list-style-type: none"> Defina os tradutores de idiomas SCLM para suporte do Java EE Defina os tipos de SCLM para suporte de Java EE 	Capítulo 4, "SCLM Developer Toolkit", na página 69
CICS TS	<ul style="list-style-type: none"> Atualização de JCL da região do CICS Atualização da CSD da região do CICS Definir grupo do CICS Definir nomes de transação do CICS Definir um programa para o CICS Definir o depurador para CICS 	<ul style="list-style-type: none"> "Suporte para linguagem bidirecional do CICS" na página 95 Capítulo 5, "Depurador Integrado", na página 77
WLM	<ul style="list-style-type: none"> Designar objetivos às tarefas do Developer for z Systems 	<ul style="list-style-type: none"> "Considerações do WLM" no <i>Referência de Configuração do Host</i> (SC27-8578) (SC14-7290)
LDAP	Definir grupos para push-to-client	"Considerações de Direcionar ao cliente" no <i>Host Configuration Reference</i> (SC27-8578)

Considerações de Pré-configuração

Para obter informações sobre o próprio Developer for z Systems, como ele interage com seu sistema e com os produtos de pré-requisito e de correquisito, consulte o *Rational Developer for z Systems Referência de Configuração do Host Guide* (SC27-8578). O *IBM Explorer for z/OS Host Configuration Reference* (SC27-8438) fornece informações semelhantes para o z/OS Explorer, que são um requisito para o Developer for z Systems. Essas informações podem ajudá-lo a criar uma configuração que suporte suas atuais necessidades e futuro crescimento.

Gerenciamento de carga de trabalho

Ao contrário dos aplicativos tradicionais do z/OS, o Developer for z Systems não é um aplicativo monolítico que pode ser identificado facilmente para Workload Manager (WLM). O Developer for z Systems consiste em vários componentes que interagem para fornecer ao cliente acesso aos serviços e dados do sistema host. Para planejar sua configuração do WLM, consulte "Considerações do WLM" no *Referência de Configuração do Host* (SC27-8578).

Nota: O Developer for z Systems consiste em diversas tarefas que se comunicam entre si e com o cliente. Estas tarefas usam vários cronômetros para detectar a perda de comunicação com seus parceiros. Podem surgir problemas de tempo limite (por falta de tempo de CPU durante a janela de tempo limite) em sistemas

com um alto carregamento da CPU ou configurações incorretas do Gerenciamento de Carga de Trabalho (WLM) para Developer for z Systems.

Uso de Recursos e Limites do Sistema

O Developer for z Systems usa uma quantidade variável de recursos, tais como espaços de endereço e processos e encadeamentos do z/OS UNIX. A disponibilidade desses recursos é limitada por várias definições do sistema. Para estimar o uso de recursos principais para que possa planejar a configuração do sistema, consulte "Considerações de ajuste" no *Referência de Configuração do Host* (SC27-8578). O Developer for z Systems pode ser executado no modo de 31 bits ou de 64 bits, alterando significativamente as limitações de recursos de armazenamento.

Configuração Necessária de Produtos de Requisito

Consulte o programador de sistema MVS, o administrador de segurança e o administrador do TCP/IP para verificar se os produtos e software de requisito estão instalados, testados e funcionando. Algumas tarefas de customização de requisito que podem ser ignoradas são listadas a seguir:

- Todos os usuários do Developer for z Systems devem ter acesso READ e EXECUTE aos diretórios Java.
- Ações remotas (baseadas em host) para subprojetos z/OS UNIX requerem que a versão do z/OS UNIX de REXEC ou SSH esteja ativa no sistema host.

Considerações de ID do usuário

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

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

Nota: Um tamanho da região grande é necessário para o ID do usuário que executa os Installation Verification Programs (IVPs), porque são executadas funções que requerem uma grande quantidade de memória (como Java). 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 se referir 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 configurado, para que os padrões do sistema sejam 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 do z/OS Explorer e Developer for z Systems, /usr/lpp/ibm/zexpl/*, /etc/zexpl/*, /var/zexpl/*, /usr/lpp/ibm/rdz/* e /var/rdz/* padrão.
- Acesso READ, WRITE e EXECUTE ao diretório WORKAREA do z/OS Explorer, /var/zexpl/WORKAREA padrão e ao diretório de log de usuário, /var/zexpl/logs padrão.
- Acesso READ aos conjuntos de dados de instalação do z/OS Explorer, FEK.SFEK* e FEL.SFEL* padrão.
- Acessos READ, WRITE e EXECUTE ao diretório /tmp ou a um diretório referenciado na variável de ambiente TMPDIR.

Considerações do Servidor

O z/OS Explorer e o Developer for z Systems consistem em múltiplos servidores ativos permanentemente, que podem ser tarefas iniciadas ou tarefas de usuário. Esses próprios servidores fornecem os serviços solicitados ou iniciam outros servidores (como encadeamentos ou tarefas de usuário do z/OS UNIX) para fornecer o serviço. Há nenhuma ordem de inicialização específica. O único requisito é que os servidores estejam ativos e em execução antes que o primeiro usuário tente se conectar. Os mecanismos de segurança usados pelo z/OS Explorer e servidores e serviços do Developer for z Systems dependem de os conjuntos de dados e os sistemas de arquivos em que eles residem serem seguros. Isto indica que apenas administradores de sistema confiáveis devem ser capazes de atualizar as bibliotecas de programa e os arquivos de configuração.

- O Debug Manager (DBGMR) fornece serviços relacionados à depuração.
- O JES Job Monitor (JMON) fornece todos os serviços relacionados ao JES. JMON faz parte do z/OS Explorer.
- O Explorador de Sistema Remoto (RSE) fornece serviços principais, tais como conectar o cliente ao sistema host e iniciar outros servidores para serviços específicos. O RSE faz parte do z/OS Explorer. O RSE consiste em duas entidades lógicas:
 - Daemon RSE (RSED), que gerencia a configuração da conexão e que é responsável pela execução no modo de servidor único.
 - Servidor RSE, que manipula solicitações de clientes individuais.

Conforme documentado em "Portas TCP/IP" no *Referência de Configuração do Host (SC27-8578)*, alguns serviços do sistema host e portanto suas portas, devem estar disponíveis para o cliente se conectar e devem ser definidos para seu firewall que protege o sistema host. Todas as outras portas usadas pelo Developer for z Systems possuem tráfego apenas do host. Listadas abaixo estão as portas que são necessárias para comunicação externa em uma configuração básicas do Developer for z Systems.

- Daemon RSE para configuração de comunicação cliente-host (usando TCP), porta padrão 4035.

- Servidor RSE para comunicação cliente-host (usando TCP). Por padrão, qualquer porta disponível é usada, mas as portas disponíveis podem ser limitadas a um intervalo especificado.

Considerações sobre pré-implementação

O Developer for z Systems suporta a clonagem de uma instalação em um sistema diferente, evitando, assim, a necessidade de uma instalação do 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 deve substituir sua contraparte nas listas a seguir.

Nota: A lista a seguir não abrange as necessidades de implementação do software de pré-requisito e de correquisito (incluindo o z/OS Explorer).

Developer for z Systems

- FEL.SFELAUTH(*)
- FEL.SFELLMOD(*)
- FEL.SFELLOAD(*)
- FEL.SFELLPA(*)
- FEL.SFELPROC(*)
- FEL.#CUST.CNTL(*)
- FEL.#CUST.PARMLIB(*)
- FEL.#CUST.PROCLIB(*)
- /usr/lpp/ibm/rdz/*
- /var/rdz/*
- definições, conjuntos de dados, arquivos e diretórios que resultam de tarefas de customização em FEL.#CUST.JCL

Utilitários do Host do Developer for z Systems

- AKG.SAKGPROC(*)
- /usr/lpp/rdzutil/*

Nota:

- FEL e /usr/lpp/ibm/rdz são o qualificador de alto nível e o caminho usados durante a instalação do Developer for z Systems. FEL.#CUST, /etc/zexpl e /var/rdz são os locais padrão usados durante a customização do produto, em que /etc/zexpl é o diretório de configuração do z/OS Explorer.
- AKG e /usr/lpp/ibm/rdzutil são o qualificador de alto nível e o caminho usados durante a instalação do Developer for z Systems Host Utilities.
- É necessário instalar o Developer for z Systems em um sistema de arquivos privado (HFS ou zFS), possivelmente combinado com o z/OS Explorer, para facilitar a implementação das partes do produto z/OS UNIX. Se não for possível usar um sistema de arquivos privado, use uma ferramenta de arquivamento, como o comando tar do z/OS UNIX para transportar os diretórios do z/OS UNIX de um sistema para outro. Este método serve para preservar os atributos (como controle de programa) para os arquivos e diretórios do Developer for z Systems.

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

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

Lista de verificação do cliente

Os usuários do cliente Developer for z Systems devem saber o resultado de algumas customizações do sistema 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 lista os resultados necessários de etapas de customização obrigatórias. A Tabela 6 lista os resultados necessários das etapas de customização opcionais.

Tabela 5. Lista de verificação do cliente: Partes obrigatórias

Customização	Valor
(pré-requisito) Número da porta TCP/IP do daemon RSE. O padrão é 4035. Essa porta é configurada durante a configuração do z/OS Explorer.	

Tabela 6. Lista de verificação do cliente: Partes opcionais

Customização	Valor
Local dos procedimentos ELAXF*, se eles não estiverem em uma biblioteca de procedimentos de sistemas. O padrão é FEL.#CUST.PROCLIB. Consulte a nota sobre JCLLIB em “procedimentos de construção remota do ELAXF*” na página 24.	
Nomes de procedimentos ou de etapas dos procedimentos ELAXF*, se eles forem alterados. Consulte a nota sobre como alterá-los em “procedimentos de construção remota do ELAXF*” na página 24.	
Local do procedimento AKGCR, se ele não estiver em uma biblioteca de procedimentos do sistema. O padrão é AKG.#CUST.PROCLIB. Consulte a nota sobre JCLLIB em “Revisão de Código” na página 87.	
Local do procedimento AKGCC, se ele não estiver em uma biblioteca de procedimentos do sistema. O padrão é AKG.#CUST.PROCLIB. Consulte a nota sobre JCLLIB em “Cobertura de Códigos” na página 88.	
Local da instrução exec do Pré-processador de Inclusão FEKRNPLI. O padrão é FEL.#CUST.CNTL. Consulte o “Suporte a pré-processador de inclusão” na página 93.	
Local dos módulos de carregamento do depurador se não estiverem em LINKLIST. O padrão é FEL.SFELAUTH. Consulte Capítulo 5, “Depurador Integrado”, na página 77	

Tabela 6. Lista de verificação do cliente: Partes opcionais (continuação)

Customização	Valor
Local dos módulos de carregamento de teste da unidade, se não estiverem em LINKLIST ou STEPLIB de rdz.env. O padrão é FEL.SFELLOAD. Consulte o “Suporte xUnit para Enterprise COBOL e PL/I” na página 94.	
Local do procedimento AZUZUNIT, se ele não estiver em uma biblioteca de procedimentos do sistema. O padrão é FEL.#CUST.PROCLIB. Consulte a nota sobre JCLLIB em “Suporte xUnit para Enterprise COBOL e PL/I” na página 94.	
Local dos arquivos XML *.xsd e *.xsl de amostra usados para formatação de saída de teste de unidade. Os padrões são /usr/lpp/ibm/rdz/samples/zunit/xsd e /usr/lpp/ibm/rdz/samples/zunit/xsl. Consulte o “Suporte xUnit para Enterprise COBOL e PL/I” na página 94.	
(correquisito) Número da porta TN3270 para o Emulador de Conexão do Host. O padrão é 23. Consulte "Portas TCP/IP" em <i>Referência de Configuração do Host</i> (SC27-8578).	
(correquisito) Número da porta REXEC ou SSH que, por padrão é 512 ou 22. Consulte o “Subprojetos do z/OS UNIX” na página 92.	
(correquisito) Número da porta do servidor Debug Tool (nenhum padrão). Consulte (Opcional) Suporte de depuração DB2 e IMS.	
Local da biblioteca de amostra SFELSAMP para amostras do CARMA RAM. O padrão é FEL.SFELSAMP. Consulte o <i>CARMA Developer's Guide</i> (SC23-7660).	
Local da JCL CRA#ASLM para alocações do conjunto de dados CARMA SCLM RAM. O padrão é FEL.#CUST.JCL. Consulte a nota sobre CRA#ASLM em “RAM do SCLM” na página 56.	

Capítulo 2. Customização Básica

As etapas de customização a seguir são comuns para os vários serviços do Developer for z Systems. Consulte os capítulos sobre os componentes específicos 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 LPA
- Conjunto de dados autorizado para APF
- Várias atualizações PARMLIB
- Várias atualizações de software de segurança
- Várias portas TCP/IP para comunicação interna e de cliente-host
- (Opcional) IPL para ativar um SVC

Para verificar a instalação e começar a usar o Developer for z Systems em seu site, execute 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 for z Systems. Para obter detalhes, consulte “Configuração da Customização”.
2. Atualize os limites do sistema z/OS UNIX, atualize o registro do produto, inicie tarefas iniciadas e defina conjuntos de dados autorizados por APF e LINKLIST e, opcionalmente, conjuntos de dados de SVCs e LPA. Para obter detalhes, consulte “Alterações PARMLIB” na página 16.
3. Crie procedimentos de tarefa iniciada e procedimentos de compilação e vinculação. Para obter detalhes, consulte “Alterações do PROCLIB” na página 23.
4. Atualize as definições de segurança. Para obter detalhes, consulte “Definições de segurança” na página 26.
5. Customize os arquivos de configuração do Developer for z Systems. Para obter detalhes, consulte:
 - “rdz.env, o arquivo de configuração de ambiente” na página 27

Configuração da Customização

O Developer for z Systems contém vários arquivos de configuração de amostra e a JCL de amostra. Para evitar sobrescrever suas customizações ao aplicar manutenção, copie todos estes membros e os arquivos do z/OS UNIX para um local diferente e customize a cópia.

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

Para criar cópias customizáveis de arquivos de configuração e JCL de configuração, e para criar os diretórios z/OS UNIX necessários, customize e envie o membro FELSETUP da amostra no conjunto de dados FEL.SFELSAMP. As etapas necessárias de customização são descritas dentro do membro.

Essa tarefa realiza as seguintes ações:

- Criar FEL.#CUST.PARMLIB e preenchê-lo com arquivos de configuração de amostra.
- Criar FEL.#CUST.PROCLIB e preenchê-lo com membros SYS1.PROCLIB da amostra.
- Criar FEL.#CUST.JCL e preenchê-lo com JCL de configuração de amostra.
- Criar FEL.#CUST.CNTL e preenchê-lo com scripts de inicialização do servidor de amostra.
- Criar FEL.#CUST.ASM e preenchê-lo com código-fonte do assembler de amostra.
- Criar FEL.#CUST.SQL e preenchê-lo com arquivos de comando SQL de amostra.
- Preencher o diretório de configuração do z/OS Explorer, /etc/zexpl/* com arquivos de configuração de amostra.
- Criar /var/rdz/* como diretórios de trabalho para várias funções do Developer for z Systems e preenchê-lo com arquivos de amostra.

Nota:

- As etapas de configuração nesta publicação usam o membro e os locais de arquivos criados pela tarefa FELSETUP, a menos que indicado de outra forma. As amostras originais, que não devem ser atualizadas, estão em FEL.SFELSAMP e /usr/lpp/ibm/rdz/samples/.
- Para obter mais detalhes sobre quais membros da amostra são copiados para qual conjunto de dados e para obter mais detalhes sobre quais diretórios são criados, sua bitmask de permissão e para onde os vários arquivos de amostra são copiados, consulte os comentários em FEL.SFELSAMP(FELSETUP).
- Para auxiliar na migração de uma configuração existente, os comentários em FEL.SFELSAMP(FELSETUP) também documentam as mudanças entre versões diferentes de Rational Developer for z Systems.
- Se você deseja manter todos os arquivos do z/OS Explorer e Developer for z Systems z/OS UNIX no mesmo sistema de arquivos (HFS ou zFS), mas também deseja que os arquivos de configuração sejam colocados em /etc/zexpl, poderá usar links simbólicos para solucionar esse problema. Os comandos z/OS UNIX da amostra a seguir criam um novo diretório no sistema de arquivos existente (/usr/lpp/ibm/rdz/cust) e definem um link simbólico (/etc/zexpl) para ele:

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

Alterações PARMLIB

As mudanças de PARMLIB a seguir são documentadas nesta seção:

- “Configurar os Limites do z/OS UNIX em BPXPRMxx” na página 17
- “Incluir as Tarefas Iniciadas em COMMNDxx” na página 17
- “Definições SVC no IEASVCxx” na página 18
- “Ativação do produto em IFAPRDxx” na página 18
- “Definições de LPA em LPALSTxx” na página 19
- “Autorizações de APF no PROGxx” na página 19
- “Definições de LINKLIST no PROGxx” na página 20
- “Definições de LINKLIST e LPA de Requisito” na página 22

- “Definições LINKLIST para Outros Produtos” na página 23

Para obter mais informações sobre as definições PARMLIB listadas nas próximas seções, consulte *MVS Initialization and Tuning Reference* (SA22-7592). Para obter mais informações sobre os comandos do console de amostra, consulte *MVS System Commands* (SA22-7627).

Configurar os Limites do z/OS UNIX em BPXPRMxx

O SCLM Developer Toolkit inicia vários processos de propriedade do usuário simultaneamente durante determinadas atividades do usuário.

Defina OMVS=xx no membro parmlib IEASYSxx para especificar qual membro parmlib BPXPRMxx deve ser usado durante o IPL.

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. Esta configuração é destinada a ser um limite de todo o sistema, porque deve estar ativo para cada cliente que usa o Developer for z Systems.

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

- DISPLAY OMVS,0
- SETOMVS MAXPROCUSER=50

Nota:

- O valor de MAXPROCUSER sugerido aqui é baseado em usuários que possuem um ID do usuário do z/OS UNIX (UID) exclusivo. Aumente esse valor se os usuários compartilharem o mesmo UID.
- Durante a instalação de SMP/E do Developer for z Systems, você foi informado para colocar o código em um sistema de arquivos separado (zFS de HFS) e para atualizar BPXPRMxx para montar este sistema de arquivos durante o IPL do sistema. Incluída está uma repetição do comando de montagem de amostra, caso esta atualização ainda deva ser feita:

```
MOUNT FILESYSTEM('#dsn')
  MOUNTPPOINT('-PathPrefix-usr/lpp/rdz')
  MODE(RDWR) /* pode ser MODE(READ) */
  TYPE(ZFS) PARM('AGGRGROW') /* zFS, com extensões */
/* TYPE(HFS) */ /* HFS, extensão auto. */
```

- Durante a instalação de SMP/E do Developer for z Systems Host Utilities, você foi avisado para colocar o código em um sistema de arquivos separado (zFS de HFS) e atualizar BPXPRMxx para montar este sistema de arquivos durante o IPL do sistema. Incluída está uma repetição do comando de montagem de amostra, caso esta atualização ainda deva ser feita:

```
MOUNT FILESYSTEM('#dsn')
  MOUNTPPOINT('-PathPrefix-usr/lpp/rdzutil')
  MODE(RDWR) /* pode ser MODE(READ) */
  TYPE(ZFS) PARM('AGGRGROW') /* zFS, com extensões */
/* TYPE(HFS) */ /* HFS, extensão auto. */
```

Incluir as Tarefas Iniciadas em COMMNDxx

Incluir comandos iniciais para os servidores Developer for z Systems para SYS1.PARMLIB(COMMNDxx) para iniciá-los automaticamente no próximo sistema IPL. Defina CMD=xx no membro parmlib IEASYSxx para especificar qual membro parmlib COMMNDxx deve ser usado durante o IPL.

O Depurador Integrado opcional requer que o servidor Developer for z Systems DBGMGR fique ativo em seu sistema.

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

- S DBGMGR

Nota: Não há uma ordem de inicialização específica para os servidores . O único requisito é que os servidores estejam ativos e em execução antes que o primeiro usuário tente se conectar.

Definições SVC no IEASVCxx

O Depurador Integrado opcional requer que uma chamada do supervisor do Developer for z Systems (SVC) esteja definida em seu sistema.

Os SVCs definidos pela instalação são definidos no SYS1.PARMLIB(IEASVCxx) e requerem que um IPL seja ativado. O módulo de carregamento relacionado deve ser carregado em LPA no horário de IPL. Observe, no entanto, que o Depurador Integrado permite fazer uma instalação ou atualização dinâmica usando a tarefa iniciada do Gerenciador de Depuração. Defina SVC=xx no membro parmlib IEASYSxx para especificar qual membro parmlib IEASVCxx deve ser usado durante o IPL.

Especifique o seguinte no IEASVCxx para definir o SVC do Developer for z Systems:
SVC Parm 251, REPLACE, TYPE(3), EPNAME(AQESVC03) /* RDz debug */

O SVC número 251 é o padrão, mas qualquer valor, dentro do intervalo de 200 a 255 ditado por z/OS, pode ser usado. O Integrated Debugger detectará o número SVC a ser usado.

Nota: Versões do Developer for z Systems anteriores à versão 9.1.1 usavam outro SVC, AQESVC01, que não está mais em uso. Ela poderá ser removida se o seu sistema host tiver apenas o Developer for z Systems versão 9.1.1 ou superior.

Ativação do produto em IFAPRDxx

Se você comprou o Developer for z Systems como parte do código do produto 5697-CDT, IBM Enterprise COBOL Suite for z/OS, V1.1 (ou posterior), e ainda não o fez para outro programa no código do produto 5697-CDT, inclua uma entrada no membro parmlib IFAPRDxx para ativar os programas relacionados. Defina PROD=xx no membro parmlib IEASYSxx para especificar qual membro parmlib IFAPRDxx deve ser usado durante o IPL.

Especifique o seguinte em IFAPRDxx para definir o Enterprise COBOL Suite for z/OS (código do produto 5697-CDT):

```
PRODUCT OWNER('IBM CORP')  
NAME('IBM COBOL SUITE')  
ID(5697-CDT)  
VERSION(*) RELEASE(*) MOD(*)  
FEATURENAME(*)  
STATE(ENABLED)
```

Como alternativa e opção, se você comprou o Developer for z Systems separadamente (NÃO como parte do Enterprise COBOL Suite for z/OS), é possível incluir uma entrada no membro parmlib IFAPRDxx para Developer for z Systems usando o código do produto independente, 5724-T07.

```

PRODUCT OWNER('IBM CORP')
NAME('IBM RDZ')
ID(5724-T07)
VERSION(*) RELEASE(*) MOD(*)
FEATURENAME(*)
STATE(ENABLED)

```

Após o membro parmlib IFAPRDxx ser atualizado, ele pode ser ativado dinamicamente (até o próximo IPL) com o seguinte comando do console:

```
SET PROD=xx
```

Nota: O Developer for z Systems registra os seguintes recursos:

- RDZ-RSED (para o Developer for z Systems)
- RDZ-CC (para Cobertura de Código em Utilitários de Host do Developer for z Systems)
- RDZ-CR (para Revisão de Código em Utilitários de Host do Developer for z Systems)

Definições de LPA em LPALSTxx

O serviço opcional Common Access Repository Manager (CARMA) suporta diferentes métodos de inicialização do servidor para o servidor CARMA. O método de inicialização CRASTART requer que os módulos na biblioteca de carregamento FEL.SFELLPA estejam na Área do pacote de links (LPA).

Os conjuntos de dados de LPA são definidos em SYS1.PARMLIB(LPALSTxx). Defina LPA=xx no membro parmlib IEASYSxx para especificar qual membro parmlib LPALSTxx deve ser usado durante o IPL.

As definições de LPA podem ser configuradas dinamicamente (até o próximo IPL) com o seguinte comando do console:

```
SETPROG LPA,ADD,DSN=FEL.SFELLPA,MASK=*
```

Nota:

- Conjuntos de dados listados no LPALSTxx devem ser catalogados no catálogo principal ou em um catálogo do usuário identificado no membro LPALSTxx.
- A inclusão de um novo conjunto de dados no LPALSTxx requer um IPL com CLPA (crie o LPA) para ser ativada.
- Todas as bibliotecas que são carregadas no LPA são automaticamente consideradas para serem autorizadas pelo APF e controladas pelo programa. Certifique-se de ter os controles de segurança apropriados em vigor para essas bibliotecas.
- Se você optar por não colocar uma biblioteca designada para colocação de LPA no LPA e usar então LINKLIST ou STEPLIB, certifique-se de definir a autorização do APF e o status de controle de programa.

Autorizações de APF no PROGxx

Para que o Debug Manager opcional funcione, os módulos nas bibliotecas de carregamento FEL.SFELAUTH e FEL.SFELLPA devem estar autorizados pelo APF. Observe que FEL.SFELLPA deve estar em STEPLIB para Debug Manager e, portanto, requer autorização explícita do APF.

Para que o serviço SCLM Developer Toolkit funcione, a biblioteca de tempo de execução REXX (REXX.*,SEAGLPA) deve ser autorizada pelo APF.

As autorizações de APF são definidas em SYS1.PARMLIB(PROGxx) por padrão. Defina PROG=xx no membro parmlib IEASYSxx para especificar qual membro parmlib PROGxx deve ser usado durante o IPL.

As autorizações de APF podem ser configuradas dinamicamente (até o próximo IPL) com os seguintes comandos de console, em que volser é o volume no qual o conjunto de dados reside, se não for gerenciado por SMS:

- SETPROG APF,ADD,DSN=FEL.SFELAUTH,SMS
- SETPROG APF,ADD,DSN=FEL.SFELLPA,SMS
- SETPROG APF,ADD,DSN=REXX.V1R4M0.SEAGLPA,VOL=volser

Nota:

- 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.
- Bibliotecas LPA, como REXX.*.SEAGLPA, são automaticamente autorizadas pelo APF quando localizadas no LPA e, portanto, não requerem definições explícitas.
- Alguns dos produtos de pré-requisito e de correquisito, tais como o z/OS Explorer, também requerem autorização do APF. Consulte os guias de customização do produto relacionados para obter informações adicionais.

Definições de LINKLIST no PROGxx

As definições LINKLIST para o Developer for z Systems podem ser agrupadas em três categorias:

- Bibliotecas de carregamento do Developer for z Systems que são necessárias para funções do Developer for z Systems. Essas definições são descritas nesta seção.
- Bibliotecas de carregamento de requisito que são necessárias para funções do Developer for z Systems. Essas definições são descritas em “Definições de LINKLIST e LPA de Requisito” na página 22.
- Bibliotecas de carregamento do Developer for z Systems que são requeridas por outros produtos. Essas definições são descritas em “Definições LINKLIST para Outros Produtos” na página 23.

Tabela 7. Corresponder módulos de carregamento com funções

Biblioteca de carregamento	Módulos de carregamento	Uso	STEPLIB
FEL.SFELAUTH	AQE* e CEE*	Capítulo 5, “Depurador Integrado”, na página 77	Procedimento ELAXFGO ou CICS
FEL.SFELLMOD	IRZ* e IIRZ*	“Mensagens de diagnósticos IRZ para o Enterprise Service Tools” na página 96	Lote do CICS, IMS ou MVS
FEL.SFELLOAD	AZU* e IAZU*	“Suporte xUnit para Enterprise COBOL e PL/I” na página 94	rdz.env ou lote do MVS
	BWB*	Capítulo 4, “SCLM Developer Toolkit”, na página 69	rdz.env

Tabela 7. Corresponder módulos de carregamento com funções (continuação)

Biblioteca de carregamento	Módulos de carregamento	Uso	STEPLIB
	CRA*	Capítulo 3, “CARMA (Common Access Repository Manager)”, na página 33	CRASUB* ou crastart*.conf
	ELAX*	“procedimentos de construção remota do ELAXF*” na página 24 (feedback de erro e pré-processador de inclusão)	Procedimentos ELAXF*
	FEJB*	“Suporte para linguagem bidirecional do CICS” na página 95	CICS
FEL.SFELLPA	CRA*	Capítulo 3, “CARMA (Common Access Repository Manager)”, na página 33	CRASRV.properties
	AQE*	Capítulo 5, “Depurador Integrado”, na página 77	não aplicável (LPA requerido)

Para que os serviços do Developer for z Systems listados funcionem, todos os módulos documentados na Tabela 7 na página 20 que estão relacionados ao serviço devem estar disponíveis por meio do STEPLIB ou LINKLIST (ou LPA). Observe que a biblioteca SFELLMOD não é usada pelo próprio Developer for z Systems, mas pelo código gerado pelo Developer for z Systems. Consulte a coluna STEPLIB na Tabela 7 na página 20 se você optar por usar a STEPLIB para saber onde a definição da STEPLIB (ou DFHRPL para CICS) deve ser feita. Entretanto, esteja ciente do seguinte:

- 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 as bibliotecas no STEPLIB não autorizadas.
- As bibliotecas incluídas na STEPLIB DD em uma JCL não são propagadas para os processos z/OS UNIX iniciados pela JCL.

Os conjuntos de dados LINKLIST são definidos em SYS1.PARMLIB(PROGxx), se seu site seguiu as recomendações da IBM. Defina PROG=xx no membro parmlib IEASYSxx para especificar qual membro parmlib PROGxx deve ser usado durante o IPL.

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) DSNAME(FEL.SFELAUTH) VOLUME(volser)
- LNKLIST ADD NAME(listname) DSNAME(FEL.SFELLOAD)

As definições de LINKLIST podem ser criadas dinamicamente (até o próximo IPL) com o grupo de comandos de console a seguir, onde volser é o volume no qual o conjunto de dados reside, se ele não estiver catalogado no catálogo principal:

1. LNKLIST DEFINE,NAME=LLTMP,COPYFROM=CURRENT
2. LNKLIST ADD NAME=LLTMP,DSN=FEL.SFELAUTH,VOL=volser
3. LNKLIST ADD NAME=LLTMP,DSN=FEL.SFELLOAD
4. LNKLIST ACTIVATE,NAME=LLTMP

Definições de LINKLIST e LPA de Requisito

As seguintes bibliotecas adicionais devem ser disponibilizadas, por meio de STEPLIB ou de LINKLIST/LPALIB, para suportar o uso de serviços opcionais. Esta lista não inclui conjuntos de dados que são específicos para um produto com o qual o Developer for z Systems interage, como o z/OS Explorer:

- Biblioteca de carregamento de sistema
 - SYS1.LINKLIB
- Tempo de execução do Ambiente de Linguagem
 - CEE.SCEERUN
 - CEE.SCEERUN2
- Biblioteca de classe DLL do C++
 - CBC.SCLBDLL
- ISPF's TSO/ISPF Client Gateway (para SCLMDT)
 - ISP.SISPLoad
 - ISP.SISPLPA
- REXX runtime library (para SCLMDT)
 - REXX.*.SEAGLPA
- Biblioteca de carregamento de sistemas (para o Integrated Debugger)
 - SYS1.MIGLIB
- Biblioteca de carregamento de sistemas (para Integrated Debugger no z/OS 1.13 e superior)
 - SYS1.SIEAMIGE
- Biblioteca de carregamentos de sistemas (para unidade de teste Enterprise COBOL e PL/I)
 - SYS1.CSSLIB
 - SYS1.SIXML0D1

Nota:

- 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.
- Todas as bibliotecas que são carregadas no LPA são automaticamente consideradas para serem autorizadas pelo APF e controladas pelo programa. Certifique-se de ter os controles de segurança apropriados em vigor para essas bibliotecas.
- As bibliotecas que são projetadas para colocação de LPA, como REXX.*.SEAGLPA, podem exigir controle de programa adicional ou autorizações do APF se forem acessadas por meio de LINKLIST ou STEPLIB.
- Alguns dos produtos de pré-requisito e correquisito, tais como o z/OS Explorer, também requerem definições STEPLIB ou LINKLIST/LPALIB. Consulte os guias de customização do produto relacionados para obter informações adicionais.

Os conjuntos de dados LINKLIST são definidos em SYS1.PARMLIB(PROGxx) por padrão. Os conjuntos de dados de LPA são definidos em SYS1.PARMLIB(LPALSTxx).

Se você optar por usar STEPLIB, deverá definir as bibliotecas não disponíveis por meio de LINKLIST/LPALIB na diretiva STEPLIB de rdz.env, o arquivo de configuração do RSE. No entanto, lembre-se do seguinte:

- A utilização de STEPLIB no z/OS UNIX tem um impacto de desempenho negativo.
- Se uma biblioteca STEPLIB não for autorizada pelo APF, todas as outras bibliotecas STEPLIB deverão ser autorizadas. As bibliotecas perderão sua autorização do APF se forem combinadas com as bibliotecas no STEPLIB não autorizadas.
- 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 Developer for z Systems possui um componente de geração de códigos chamado Enterprise Service Tools. Para que o código gerado emita mensagens de erro de diagnóstico, todos os módulos IRZM* e IIRZ* na biblioteca de carregamento FEL.SFELMOD devem ser disponibilizados seja por STEPLIB ou LINKLIST.

Os conjuntos de dados LINKLIST são definidos em SYS1.PARMLIB(PROGxx) por padrão.

Se você optar por usar STEPLIB, deverá definir as bibliotecas que não estão disponíveis por meio de LINKLIST na diretiva STEPLIB da tarefa que executa o código (IMS ou a tarefa em lote). No entanto, se uma biblioteca STEPLIB for autorizada pelo APF, todas as outras bibliotecas STEPLIB deverão ser autorizadas. As bibliotecas perderão sua autorização do APF se forem combinadas com as bibliotecas no STEPLIB não autorizadas.

Alterações do PROCLIB

As mudanças de PROCLIB a seguir são documentadas nesta seção:

- “DBGMR, tarefa iniciada do gerenciador de depuração”
- “procedimentos de construção remota do ELAXF*” na página 24

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

DBGMR, tarefa iniciada do gerenciador de depuração

Customize o membro de tarefa que iniciou a amostra FEL.#CUST.PROCLIB(DBGMR), conforme descrito no membro e copie-o em SYS1.PROCLIB. Conforme mostrado na amostra de código a seguir, forneça estas informações:

- A compensação do fuso horário, padrão EST5DST
- A porta usada para comunicação externa (cliente-host), padrão 5335
- A porta usada para comunicação interna (confinada no host), padrão 5336
- O número do SVC usado pelo Depurador Integrado, padrão 251
- O qualificador de alto nível da biblioteca de carregamento, padrão FEL

```

/*
/* RDz Debug Manager
/*
//DBGMGR   PROC PRM=,                * PRM=DEBUG TO START TRACING
//          LEPRM='RPTOPTS(ON)',
//          TZ='EST5EDT',
//          CLIENT=5335,
//          HOST=5336,
//          SVC=251,
//          HLQ=FEL
/*
//DBGMGR   EXEC PGM=AQEZPCM,REGION=0M,TIME=NOLIMIT,
//          PARM=('&LEPRM ENVAR("TZ=&TZ")/&HOST &CLIENT &SVC &PRM')
//STEPLIB DD DISP=SHR,DSN=&HLQ..SFELAUTH
//          DD DISP=SHR,DSN=&HLQ..SFELLPA
//SYSPRINT DD SYSOUT=*
//SYSOUT   DD SYSOUT=*
//          PEND
/*

```

Figura 1. DBGMGR: tarefa iniciada do gerenciador de depuração

Nota:

- Essa é uma tarefa iniciada opcional. Ela é usada pelo recurso do Depurador Integrado de Developer for z Systems. Para obter mais informações, consulte Capítulo 5, “Depurador Integrado”, na página 77.
- A JCL de amostra é inicialmente chamada de FEL.SFELSAMP(AQEJCL) e é renomeada para FEL.#CUST.PROCLIB(DBGMGR) em “Configuração da Customização” na página 15.
- Para os objetivos recomendados do Workload Manager (WLM) para esta tarefa, consulte “Considerações de WLM” no *Host Configuration Reference (SC27-8578)*.
- Se o SVC do Depurador Integrado já estiver carregado, o número do SVC especificado aqui será ignorado e o número do SVC ativo será usado.

procedimentos de construção remota do ELAXF*

O Developer for z Systems 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 usem os mesmos procedimentos com as mesmas opções e níveis do compilador.

Os procedimentos de amostra e suas funções são listados na Tabela 8.

Tabela 8. 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 compilação COBOL e conversão do CICS Integrado e do DB2 integrado.
ELAXFCOP	Procedimento de amostra para pré-processamento do DB2 de instruções EXEC SQL integradas em programas COBOL.

Tabela 8. Procedimento ELAXF* de amostra (continuação)

Membro	Propósito
ELAXFCOT	Procedimento de amostra para conversão do CICS para Instruções EXEC CICS integradas em programas COBOL.
ELAXFCPC	Procedimento de amostra para compilação C.
ELAXFCPP	Procedimento de amostra para compilação C++.
ELAXFCP1	Procedimento de amostra para compilação COBOL com instruções do pré-processador SCM (-INC e ++INCLUDE).
ELAXFDCL	Procedimento de amostra para executar um programa em modo TSO.
ELAXFGO	Procedimento de amostra para a etapa IR.
ELAXFLNK	Procedimento de amostra para realizar vinculação de C/C++, COBOL. Programas PLI e High Level Assembler.
ELAXFMFS	Procedimento de amostra para criar telas IMS MFS.
ELAXFPLP	Procedimento de amostra para pré-processamento do DB2 de instruções EXEC SQL integradas em 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 compilação PL/I e conversão do CICS integrado e conversão do DB2 integrado.
ELAXFPP1	Procedimento de amostra para compilação PL/I com instruções do pré-processador SCM (-INC e ++INCLUDE).
ELAXFSP	Procedimento de amostra para registrar um procedimento armazenado no DB2.
ELAXFSQL	Procedimento de amostra para chamar SQL.
ELAXFTSO	Procedimento de amostra para executar e depurar o 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 estão incluídas no cliente Developer for z Systems. Se o nome de um procedimento ou o nome de uma etapa em um procedimento for alterado, o arquivo de propriedades correspondente em todos os clientes deverá ser atualizado. Você não deve alterar os nomes de etapa e procedimento.

Customize os membros do procedimento de construção de amostra, FEL.#CUST.PROCLIB(ELAXF*), conforme descrito nos membros e copie-os para SYS1.PROCLIB. Forneça os qualificadores de alto nível corretos para bibliotecas do produto diferentes, conforme descrito na Tabela 9.

Tabela 9. Lista de Verificação do Qualificador de Alto Nível ELAXF*

Produto	HLQ Padrão	Valor
Developer for z Systems	FEK	
CICS	CICSTS52.CICS	
DB2	DSNA11	
IMS	IMS	
COBOL	IGY.V5R2M0	

Tabela 9. Lista de Verificação do Qualificador de Alto Nível ELAXF* (continuação)

Produto	HLQ Padrão	Valor
PL/I	PLI.V4R2M0	
C/C++	CBC	
LE	CEE	
LINKLIB do sistema	SYS1	
MACLIB do sistema	SYS1	

Alguns procedimentos ELAXF* referenciam os nomes do conjunto de dados que não possuem qualificadores de nível baixo fixo. Um exemplo é a biblioteca de execução DB2, que mantém os utilitários do DB2 que são compilados pelo seu administrador do DB2. Use o Tabela 10 para mapear os nomes do conjunto de dados padrão para os nomes usados em seu site.

Tabela 10. ELAXF*. Lista de verificação do conjunto de dados completo

Produto	DSN Padrão	Valor
Developer for z Systems - Amostras SQL	FEL.#CUST.SQL	
Bibliotecas de execução do DB2	DSNA11.RUNLIB.LOAD	

Se os procedimentos ELAXF* não puderem ser copiados para uma biblioteca de procedimentos de sistema, solicite aos usuários do Developer for z Systems incluírem um cartão JCLLIB (logo após o cartão JOB) nas propriedades da tarefa no cliente.

```
//MYJOB    JOB <job parameters>
//PROCS    JCLLIB ORDER=(FEL.#CUST.PROCLIB)
```

Definições de segurança

Para criar as definições de segurança para o Developer for z Systems, customize e envie os membros da amostra FELRACF e AQERACF. O usuário que enviar essa tarefa deve ter privilégios de administrador de segurança, como sendo RACF SPECIAL.

FELRACF e AQERACF estão localizados em FEL.#CUST.JCL, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEL.SFELSAMP(FELSETUP). Para obter mais detalhes, consulte “Configuração da Customização” na página 15.

A lista a seguir de definições relacionadas à segurança para Developer for z Systems são discutidas em detalhes em Capítulo 9, “Definições de segurança”, na página 103.

- Ativar configurações e classes de segurança
- Definir Debug Manager como um servidor z/OS UNIX seguro
- Definir acesso ao Integrated Debugger
- Definir perfis do conjunto de dados

rdz.env, o arquivo de configuração de ambiente

Os processos do servidor RSE do z/OS Explorer (daemon RSE, conjunto de encadeamentos RSE e servidor RSE) usam as definições em `rdz.env` para saber sobre as variáveis de ambiente do Developer for z Systems.

`rdz.env` está localizado em `/etc/zexpl/`, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa `FEL.SFELSAMP(FELSETUP)`. Para obter mais detalhes, consulte “Configuração da Customização” na página 15. É possível editar o arquivo com o comando do TSO **OEDIT**.

Consulte o arquivo `rdz.env` de amostra a seguir, que pode ser customizado para corresponder ao seu ambiente do sistema. Os valores padrão são fornecidos para todas as variáveis que não são especificadas explicitamente. A sintaxe do arquivo segue o padrão das regras de sintaxe do shell do z/OS UNIX. Por exemplo, comentários iniciam com um sinal de número (#) ao usar uma página de códigos dos EUA e espaços ao redor do sinal de igual (=) não são suportados.

Nota: Para que suas mudanças entrem em vigor, a tarefa iniciada pelo RSED do z/OS Explorer deve ser reiniciada.

```
#RDZ_HOME=/usr/lpp/ibm/rdz
#RDZ_HLQ=FEL

## debug
# RSE_JAVAOPTS="$_RSE_JAVAOPTS -Ddebug.miner.autoreconnect=0"
# _RSE_JAVAOPTS="$_RSE_JAVAOPTS -Ddebug.miner.localhost=localhost"
## C/C++
# RSE_JAVAOPTS="$_RSE_JAVAOPTS -Dinclude.c=/etc/zexpl/include.conf"
# _RSE_JAVAOPTS="$_RSE_JAVAOPTS -Dinclude.cpp=/etc/zexpl/include.conf"
# _RSE_JAVAOPTS="$_RSE_JAVAOPTS -DCPP_CLEANUP_INTERVAL=60000"

## remote index search
# RSE_JAVAOPTS="$_RSE_JAVAOPTS -DRIS_BUFFER=8"
# _RSE_JAVAOPTS="$_RSE_JAVAOPTS -DDISABLE_REMOTE_INDEX_SEARCH=true"

## system
# _RSE_JAVAOPTS="$_RSE_JAVAOPTS -DDISABLE_DELETE_IN_SUBPROJECT=true"

## zUnit unit test
#STEPLIB=$STEPLIB:$RDZ_HLQ.SFELLOAD
#STEPLIB=$STEPLIB:SYS1.CSSLIB:SYS1.SIXMLOD1

## RTC user build
#RDZ_UBLD_DD=$CGI_ISPCONF/ISPF.conf
#RDZ_UBLD_STEPLIB=$STEPLIB

## SCLM Developer Toolkit
# SCLMDT_CONF_HOME=/var/rdz/sclmdt
#STEPLIB=$STEPLIB:$RDZ_HLQ.SFELAUTH:$RDZ_HLQ.SFELLOAD
# SCLMDT_TRANTABLE=FEL.#CUST.LSTRANS.FILE
#ANT_HOME=/usr/lpp/Apache/Ant/apache-ant-1.7.1
```

Figura 2. Arquivo de configuração do ambiente `rdz.env`:

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

RDZ_HOME

Diretório inicial do Developer for z Systems. O padrão é `/usr/lpp/ibm/rdz`. Remova o comentário e mude para corresponder à sua instalação do Developer for z Systems.

RDZ_HLQ

Qualificador de alto nível usado para instalar o Developer for z Systems. O padrão é FEK. Remova o comentário e mude para corresponder ao local de seus conjuntos de dados do Developer for z Systems.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Ddebug.miner.autoreconnect=0"

Reconexão automática para o Debug Manager. O padrão é 1, que significa que quando uma conexão com o servidor do Debug Manager não estiver estabelecida ou for perdida, o Debug Miner tentará uma vez reconectar ao Debug Manager. Remova o comentário e especifique um valor diferente para limitar a frequência com que o Debug Miner tentará conectar-se ao Debug Manager.

Tabela 11. Reconexão automática para o Debug Manager

debug.miner.autoreconnect	Comportamento de reconexão
-1	Não reconectar
0 (padrão)	Tentativa de reconexão a cada minuto até que seja bem sucedida
1-86400	Tentativa de reconexão atingiu a quantidade de vezes especificada. O valor máximo, 86400, equivale a 24 horas.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Ddebug.miner.localhost=localhost"

Alternativa para a definição localhost TCP/IP. O Debug Miner tentará conectar-se ao Debug Manager usando a especificação localhost. Isso falhará se localhost não resolver o endereço de loopback local (127.0.0.1 para IVPv4, ::1 para IPv6). Remova o comentário e especifique o endereço loopback local quando necessário.

_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Dinclude.c=/etc/zexpl/include.conf"

Essa variável aponta para um arquivo completo do z/OS UNIX contendo uma lista de inclusões forçadas para assistência de conteúdo em código C. Uma inclusão forçada consiste em um arquivo ou diretório, conjunto de dados ou membro do conjunto de dados que é analisado quando uma operação do assistente de conteúdo é executada, independentemente de o arquivo ou membro estar incluído no código-fonte usando uma diretiva de pré-processador. Para especificar o nome do arquivo de configuração, remova o comentário e customize.

_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Dinclude.cpp=/etc/zexpl/include.conf"

Essa variável aponta para um arquivo completo do z/OS UNIX contendo uma lista de inclusões forçadas para assistência de conteúdo em código C++. Uma inclusão forçada consiste em um arquivo ou diretório, conjunto de dados ou membro do conjunto de dados que é analisado quando uma operação do assistente de conteúdo é executada, independentemente de o arquivo ou membro estar incluído no código-fonte usando uma diretiva de pré-processador. Para especificar o nome do arquivo de configuração, remova o comentário e customize.

_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -DCPP_CLEANUP_INTERVAL=60000"

Intervalo de limpeza para arquivos de cabeçalho C/C++ não usados em milissegundos. O padrão é 60000, que significa 1 minuto. Para mudar o intervalo de limpeza, remova o comentário e customize. Especificar um valor de 0 impede o armazenamento em cache de arquivos de cabeçalho C/C++, reduzindo, portanto, o desempenho do assistente de conteúdo remoto no editor.

_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -DRIS_BUFFER=8"

Tamanho do buffer, em megabytes, usado durante a criação de índice remota. O padrão é 8 MB. Para alterar o tamanho do buffer, remova o comentário e customize. Os valores válidos são números inteiros entre 1 e 2000 (ambos inclusive). Um buffer maior acelera a criação de índice, mas usa uma parte maior do heap Java do conjunto de encadeamentos. O buffer é limpo automaticamente para o índice se ele estiver cheio antes da conclusão da criação do índice.

**_RSE_JAVAOPTS="\$_RSE_JAVAOPTS
-DDISABLE_REMOTE_INDEX_SEARCH=true"**

Desative o item de menu Remote Index Search no cliente. O padrão é false. Para evitar que usuários criem índices remotos para conjuntos de dados do sistema host, remova o comentário e especifique true.

**#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS
-DDISABLE_DELETE_IN_SUBPROJECT=true"**

Desative o item de menu Excluir no menu de contexto de subprojetos do z/OS. O padrão é false. Para evitar que os usuários usem o item de menu Excluir no menu de contexto de subprojetos do z/OS, remova o comentário e especifique true.

STEPLIB

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

É possível ignorar a necessidade de ter bibliotecas de pré-requisito em LINKLIST/LPALIB, removendo o comentário e customizando uma ou mais das seguintes diretivas STEPLIB. Para obter informações adicionais sobre o uso das bibliotecas na lista a seguir, consulte "Alterações PARMLIB" na página 16:

```
# SCLM Developer Toolkit
STEPLIB=$STEPLIB:$RSE_HLQ.SFELAUTH:$RSE_HLQ.SFELLOAD
# zUnit, xUnit suporte para Enterprise COBOL e PL/I
STEPLIB=$STEPLIB:$RSE_HLQ.SFELLOAD:SYS1.CSSLIB:SYS1.SIXMLOD1
```

Nota:

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

Especifica as instruções DD que serão usadas ao gerar JCL para as construções do usuário do IBM Rational Team Concert a partir de um cliente Developer for z Systems que chama os comandos TSO ou ISPF. Por padrão, o Developer for z Systems usa as definições em ISPF.conf, que são referenciadas por CGI_ISPCONF em rdz.env. Remova o comentário e altere para usar as definições do DD no arquivo especificado, que deve seguir as regras de sintaxe especificadas no *ISPF.conf*, o *arquivo de configuração do Gateway do Cliente TSO/ISPF do ISPF*. Essa é uma diretiva opcional.

RSE_UBLD_STEPLIB

Especifica as instruções STEPLIB que serão usadas ao gerar JCL para as construções do usuário do IBM Rational Team Concert a partir de um cliente Developer for z Systems que chama os comandos TSO ou ISPF. Por padrão, o Developer for z Systems usa a definição STEPLIB em rdz.env. Remova o comentário e altere para usar a definição STEPLIB especificada. Essa é uma diretiva opcional.

_SCLMDT_CONF_HOME

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

Nota: O SCLMDT inclui /CONFIG e /CONFIG/PROJECT no caminho especificado em SCLMDT_CNF_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 FEL.#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 usada somente se a tradução de nome longo/curto 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 é usada somente quando o suporte de construção do Java EE for usado com o SCLM Developer Toolkit.

Componentes específicos

O Developer for z Systems consiste em vários recursos não relacionados, cada um com suas próprias tarefas de customização. Siga as instruções na seção apropriada para configurar o serviço necessário.

Customizações aos componentes independentes para Developer for z Systems:

- Capítulo 3, “CARMA (Common Access Repository Manager)”, na página 33
- Capítulo 4, “SCLM Developer Toolkit”, na página 69
- Capítulo 5, “Depurador Integrado”, na página 77
- Capítulo 6, “Análise de código baseada em host”, na página 87

Customizações para arquivos de configuração do Developer for z Systems:

- “include.conf, Inclusões forçadas para assistência de conteúdo C/C++” na página 91

Customizações relacionadas ao Developer for z Systems em ou para outros produtos:

- “Subprojetos do z/OS UNIX” na página 92
- “Suporte a pré-processador de inclusão” na página 93
- “Suporte xUnit para Enterprise COBOL e PL/I” na página 94
- “Enterprise Service Tools, suporte” na página 95

- “Suporte para linguagem bidirecional do CICS” na página 95
- “Mensagens de diagnósticos IRZ para o Enterprise Service Tools” na página 96
- “FEKRNPLI, pré processador remoto PL/I” na página 97

Verificação de Instalação

A descrição detalhada dos vários programas de verificação de instalação (IVPs) está localizada em Capítulo 8, “Verificação de Instalação”, na página 99.

Capítulo 3. CARMA (Common Access Repository Manager)

O Common Access Repository Manager (CARMA) é uma plataforma do servidor para os Repository Access Managers (RAMs). Uma RAM é uma API (Interface de Programação de Aplicativos) para um SCM (Software Configuration Manager) baseado em z/OS. Ao agrupar a funcionalidade do SCM em um RAM, uma única API está disponível para um cliente acessar qualquer SCM suportado.

O Developer for z Systems fornece diversos exemplos de RAMs e códigos-fonte pré-construídos para criar seu próprio RAM.

Os SCMs que são baseados em sistemas host precisam de espaços de endereço de usuário único para acessar seus serviços, o que requer que o CARMA inicie um servidor CARMA para cada usuário. Não é possível criar um único servidor de suporte para diversos usuários.

Requisitos e Lista de Verificação

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

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

Para começar a usar o CARMA em seu site, execute as seguintes tarefas. A menos que especificado o contrário, todas as tarefas são obrigatórias.

1. Escolha um método para iniciar o CARMA e escolha quais RAMs devem ser ativados. Várias combinações de RAMs e métodos de inicialização do servidor estão disponíveis como uma configuração pré-configurada. Para obter detalhes, consulte “Selecionar o Método de Inicialização do Servidor e o RAM Ativo” na página 34.
2. Crie conjuntos de dados CARMA VSAM. Para obter detalhes, consulte “Conjuntos de Dados CARMA VSAM” na página 54 e “Repository Access Managers (RAMs) CARMA” na página 55.
3. A customização inicial dos arquivos de configuração do RSE para interação com o CARMA. A customização completa depende do método escolhido para iniciar o CARMA. Para obter detalhes, consulte “CRASRV.properties, a Interface RSE para CARMA” na página 46.
4. Dependendo do método de inicialização CARMA escolhido e dos RAMs escolhidos, faça a customização necessária dos arquivos de configuração relacionados. Para obter detalhes, consulte:
 - “crastart*.conf, A Inicialização do Servidor CRASTART” na página 49
 - “CRASUB*, a Inicialização do Servidor de Envio em Lote” na página 52
5. Opcionalmente, customize os membros de configuração específicos do CA Endevor® SCM. Para obter detalhes, consulte “CRACFG, CRASCL, CRASHOW

e CRATMAP, os arquivos de configuração do CA Endevor® SCM RAM” na página 57 e “Ações em Lote do CA Endevor® SCM RAM” na página 59.

6. Opcionalmente, atualize o executável de alocação do conjunto de dados. Para obter detalhes, consulte “CRANDVRA, o executável de alocação inicial de RAM do CA Endevor® SCM” na página 58, “CRAALLOC, o executável de alocação customizado de RAM” na página 62 e “(Opcional) Executável de Alocação Customizado” na página 64.
7. Opcionalmente, crie uma saída do usuário de inicialização. Para obter detalhes, consulte a saída do usuário do CARMA (opcional).
8. Opcionalmente, crie um CRAXJCL como substituição para IRXJCL. Para obter detalhes, consulte “(Opcional) IRXJCL versus CRAXJCL” na página 66.

Nota: Os membros de amostra referenciados neste capítulo estão localizados em FEL.#CUST.* e /etc/zexpl, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEL.SFELSAMP(FELSETUP). Para obter mais detalhes, consulte “Configuração da Customização” na página 15.

Selecionar o Método de Inicialização do Servidor e o RAM Ativo

Developer for z Systems suporta múltiplos métodos para iniciar um servidor CARMA. Developer for z Systems também fornece múltiplos Repository Access Managers (RAMs), que podem ser divididos em dois grupos, RAMs de produção e RAMs de amostra. Essa publicação descreve várias combinações possíveis de RAMs e métodos de inicialização do servidor. Cada um dos cenários de configuração descritos está disponível como uma configuração pré-configurada.

Inicialização do Servidor CARMA

Developer for z Systems suporta múltiplos métodos para iniciar um servidor CARMA. Cada método tem benefícios e desvantagens.

CRASTART

O método "CRASTART" inicia o servidor CARMA como uma subtarefa dentro do RSE. Este método fornece uma configuração muito flexível usando um arquivo de configuração separado que define alocações do conjunto de dados e chamadas de programa que são necessárias para iniciar um servidor CARMA. Este método fornece o melhor desempenho e usa a menor quantidade de recursos, mas requer que o módulo CRASTART esteja localizado no LPA.

Envio em Lote

O método "envio em lote" inicia o servidor CARMA enviando uma tarefa. Este 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. No entanto, este método usa um inicializador JES para cada desenvolvedor que inicia um servidor CARMA.

RAMs de Produção

Os RAMs de tipo de produção são totalmente funcionais, pré-construídos para poderem ser usados para acessar um SCM em um ambiente de produção.

CA Endeavor® SCM RAM

O IBM Rational Developer for z Systems Interface for CA Endeavor® Software Configuration Manager dá aos clientes do Developer for z Systems acesso direto ao CA Endeavor® SCM.

RAM de Pacotes do CA Endeavor® SCM

O RAM de pacotes CA Endeavor® SCM fornece aos clientes Developer for z Systems acesso direto a pacotes CA Endeavor® SCM.

RAMs de Amostra

Os RAMs de amostra são fornecidos com o propósito de testar a configuração do ambiente do CARMA e como exemplos para o desenvolvimento de seus próprios RAMs. O código-fonte está incluído.

Atenção: Não use os RAMs de amostra fornecidos em um ambiente de produção.

PDS RAM

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

RAM de Base

O RAM de base fornece uma estrutura funcional que pode ser utilizada como ponto de início para o desenvolvimento de seu próprio RAM.

RAM do SCLM

O SCLM RAM fornece uma entrada básica para SCLM, o Software Configuration Manager do ISPF. O SCLM RAM não é ativado por padrão.

RAM Pré-configurado e Combinações de Inicialização do Servidor

Várias combinações de RAMs e métodos de inicialização do servidor estão disponíveis como uma configuração pré-configurada. Os cenários listados precisam apenas de uma customização menor para ajuste ao seu ambiente.

- “CRASTART com CA Endeavor® SCM RAM”
- “CRASTART com RAMs de Amostra” na página 38
- “Submissão em Lote com CA Endeavor® SCM RAM” na página 40
- “Submissão em Lote com RAMs de Amostra” na página 43

Informações detalhadas sobre diferentes etapas de cada cenário podem ser localizadas em “Detalhes da Configuração do CARMA” na página 46.

É possível incluir um RAM em qualquer configuração do CARMA, agora ou em alguma parte no futuro. Consulte “(Opcional) Suportando Múltiplos RAMs” na página 63 para obter informações adicionais sobre como incluir um RAM em uma configuração existente.

CRASTART com CA Endeavor® SCM RAM

As informações nesta seção descrevem como configurar CARMA com as seguintes especificações:

- Inicialização do servidor: Método CRASTART. Este método requer que o CRASTART esteja no LPA.
- RAM: CA Endeavor® SCM RAM.

Esta etapa de customização pode ser omitida se você desejar usar um dos outros cenários com diferentes especificações.

Criar os Conjuntos de Dados VSAM do CARMA

Para definir e preencher os conjuntos de dados VSAM relacionados ao CARMA, customize e envie as seguintes tarefas de JCL. Para instruções de customização, consulte a documentação no membro. Os conjuntos de dados VSAM existentes são substituídos.

Para obter detalhes adicionais sobre esta etapa, consulte “Conjuntos de Dados CARMA VSAM” na página 54.

- FEL.#CUST.JCL(CRA\$VCAD)
- FEL.#CUST.JCL(CRA\$VCAS)
- FEL.#CUST.JCL(CRA\$VMSG)

Customizar CRASRV.properties

O servidor RSE usa as configurações em `/etc/zexpl/CRASRV.properties` para iniciar e conectar a um servidor CARMA. É possível editar o arquivo com o comando do TSO **EDIT**. Para que as mudanças entrem em vigor, reinicie a tarefa iniciada pelo RSED.

Quando você usa os locais de arquivos padrão, as únicas mudanças necessárias são mudar o valor da diretiva `clist.dsname` para `*CRASTART` e mudar o valor de `crastart.configuration.file` para `crastart.endevor.conf`. Para obter informações adicionais sobre as diferentes diretivas, consulte “CRASRV.properties, a Interface RSE para CARMA” na página 46.

```
clist.dsname=*CRASTART
crastart.configuration.file=crastart.endevor.conf
```

Figura 3. CRASRV.properties: CRASTART com o CA Endeavor® SCM RAM

Customizar crastart.endevor.conf

CRASTART usa as definições em `/etc/zexpl/crastart.endevor.conf` para criar um ambiente TSO/ISPF válido para iniciar um servidor CARMA. É possível editar o arquivo com o comando do TSO **EDIT**. As mudanças entram em vigor para todos os servidores CARMA que são iniciados após a atualização.

Para instruções de customização, consulte a documentação no arquivo. Para obter mais informações sobre o método de inicialização CRASTART, consulte “crastart*.conf, A Inicialização do Servidor CRASTART” na página 49.

Nota: Devido às limitações de largura da página, algumas linhas na amostra a seguir foram quebradas para a linha seguinte. Todas as linhas que começam com uma endentação devem ser incluídas no final da linha anterior.

```

* DD usado pelo RAM
TYPEMAP = FEL.#CUST.PARMLIB(CRATMAP)
SHOWVIEW= FEL.#CUST.PARMLIB(CRASHOW)
CRACFG = FEL.#CUST.PARMLIB(CRACFG)
* uncomment CRABCFG and CRABSKEL to use batch actions
*CRABCFG = FEL.#CUST.PARMLIB(CRABCFG)
*CRABSKEL= FEL.#CUST.CNTL
* uncomment and provide correct DSN to use Package Ship
*APIHJC = #shiphjc
CONLIB = CA.NDVR.CSIQLOAD
-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(EXT2ELM) 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)
C1EXMSG= SYSOUT(H)
C1MSG1 = SYSOUT(H)
MSG3FILE= DUMMY

* DD utilizado pelo servidor CARMA (CRASERV)
* preste atenção às autorizações APF ao usar TASKLIB
TASKLIB = FEL.SFELLOAD,CA.NDVR.CSIQAUTH,CA.NDVR.CSIQAUTU
CRADEF = FEL.#CUST.CRADEF
CRAMSG = FEL.#CUST.CRAMSG
CRASTRS = FEL.#CUST.CRASTRS
CARMALOG = SYSOUT(H)
SYSPRINT= SYSOUT(H)

* DD usado pelo ISPF (via NDVRC1)
-COMMAND=ALLOC FI(ISPCTL0) NEW DELETE DSORG(PS) RECFM(F,B) LRECL(80)
  BLKSIZE(32720) SPACE(5,5) TRACKS UNIT(SYSALLDA)
-COMMAND=ALLOC FI(ISPCTL1) NEW DELETE DSORG(PS) RECFM(F,B) LRECL(80)
  BLKSIZE(32720) SPACE(5,5) TRACKS UNIT(SYSALLDA)
-COMMAND=ALLOC FI(ISPPROF) NEW DELETE DSORG(PO) RECFM(F,B) LRECL(80)
  BLKSIZE(32720) SPACE(5,5) TRACKS UNIT(SYSALLDA) DIR(5)
ISPTABL = -ISPPROF
ISPTLIB = -ISPPROF,ISP.SISPTENU
ISPLIB = ISP.SISPMENU
ISPLLIB = ISP.SISPPENU
ISPSLIB = ISP.SISPSENU

* DD utilizado pelo TS0 (IKJEFT01)
SYSPROC = FEL.SFELPROC
SYSTSIN = DUMMY
SYSTSPRT = SYSOUT(H)

PROGRAM=IKJEFT01 %CRANDVRA NDVRC1 PGM(CRASERV) PARM(&CRAPRM1.
  &CRAPRM2. &CRAPRM3. &CRAPRM4. &CRAPRM5. &CRAPRM6. &CRAPRM7.
  &CRAPRM8. )

```

Figura 4. *crastart.endevor.conf*: CRASTART com CA Endevor® SCM RAM

(Opcional) Customização Adicional do CA Endevor® SCM RAM

O CA Endevor® SCM RAM possui componentes adicionais que podem ser customizados, se necessário.

- O processamento da inicialização do CARMA possui uma saída opcional do usuário. Para obter mais informações, consulte a saída do usuário do CARMA (opcional).

- O CA Endeavor® SCM RAM possui vários arquivos de configuração FEL.#CUST.PARMLIB(CRA*) que podem ser customizados. Para obter mais informações, consulte “CRACFG, CRASCL, CRASHOW e CRATMAP, os arquivos de configuração do CA Endeavor® SCM RAM” na página 57.
- O CA Endeavor® SCM RAM possui um executável de alocação, FEL.SFELPROC(CRANDVRA), que pode ser customizado. Para obter mais informações, consulte “CRANDVRA, o executável de alocação inicial de RAM do CA Endeavor® SCM” na página 58.
- O CA Endeavor® SCM RAM suporta a execução de ações CA Endeavor® SCM no modo em lote. As ações em lote requerem um arquivo de configuração, FEL.#CUST.PARMLIB(CRABCFG), e uma JCL de estrutura básica, FEL.#CUST.CNTL(CRABATCA), que devem ser customizados. Para obter mais informações, consulte “Ações em Lote do CA Endeavor® SCM RAM” na página 59.

CRASTART com RAMs de Amostra

As informações nesta seção descrevem como configurar CARMA com as seguintes especificações:

- Inicialização do servidor: Método CRASTART. Este método requer que o CRASTART esteja no LPA.
- RAM: RAMs de amostra, que não devem ser usados para propósitos de produção.

Essa etapa de customização pode ser ignorada se você quiser utilizar algum outro cenário com especificações diferentes.

Criar os Conjuntos de Dados VSAM do CARMA

Customize e envie as seguintes tarefas JCL para definir e preencher os conjuntos de dados VSAM relacionados ao CARMA. Para instruções de customização, consulte a documentação no membro. Os conjuntos de dados VSAM existentes são substituídos.

Para obter detalhes adicionais sobre esta etapa, consulte “Conjuntos de Dados CARMA VSAM” na página 54 e “Repository Access Managers (RAMs) CARMA” na página 55.

CARMA

- FEL.#CUST.JCL(CRA\$VDEF)
- FEL.#CUST.JCL(CRA\$VMSG)
- FEL.#CUST.JCL(CRA\$VSTR)

RAMs de Amostra

- FEL.#CUST.JCL(CRA#VPDS)

Customizar CRASRV.properties

O servidor RSE usa as configurações em /etc/zexpl/CRASRV.properties para iniciar e conectar a um servidor CARMA. É possível editar o arquivo com o comando do TSO **OEDIT**. Para que as mudanças entrem em vigor, a tarefa iniciada pelo RSED deve ser reiniciada.

Durante o uso dos locais de arquivo padrão, a única mudança necessária é alterar o valor da diretiva `clist.dsname` para `*CRASTART`. Para obter informações adicionais

sobre as diferentes diretivas, consulte “CRASRV.properties, a Interface RSE para CARMA” na página 46.

```
clist.dsname=*CRASTART
crastart.configuration.file=crastart.conf
```

Figura 5. CRASRV.properties: CRASTART com RAMs de amostra

Customizar crastart.conf

CRASTART usa as definições em /etc/zexpl/crastart.conf para criar um ambiente TSO/ISPF válido para iniciar um servidor CARMA. É possível editar o arquivo com o comando do TSO **OEDIT**. As mudanças entram em vigor para todos os servidores CARMA que são iniciados após a atualização.

Para instruções de customização, consulte a documentação no arquivo. Para obter mais informações sobre o método de inicialização CRASTART, consulte “crastart*.conf, A Inicialização do Servidor CRASTART” na página 49.

```
* DD usado pelo RAM
CRARAM1 = FEL.#CUST.CRARAM1                                * PDS RAM
* DD usado pelo servidor CARMA (CRASERV)
TASKLIB = FEL.SFELLOAD
CRADEF = FEL.#CUST.CRADEF
CRAMSG = FEL.#CUST.CRAMSG
CRASTRS = FEL.#CUST.CRASTRS
CARMALOG = SYSOUT(H)
SYSPRINT= SYSOUT(H)

* DD utilizado pelo ISPF (ISPSTART)
-COMMAND=ALLOC FI(ISPCTL0) NEW DELETE DSORG(PS) RECFM(F,B) LRECL(80)
  BLKSIZE(32720) SPACE(5,5) TRACKS UNIT(SYSALLDA)
-COMMAND=ALLOC FI(ISPCTL1) NEW DELETE DSORG(PS) RECFM(F,B) LRECL(80)
  BLKSIZE(32720) SPACE(5,5) TRACKS UNIT(SYSALLDA)
-COMMAND=ALLOC FI(ISPPROF) NEW DELETE DSORG(PO) RECFM(F,B) LRECL(80)
  BLKSIZE(32720) SPACE(5,5) TRACKS UNIT(SYSALLDA) DIR(5)
ISPTABL = -ISPPROF
ISPTLIB = -ISPPROF,ISP.SISPTENU
ISPLIB = ISP.SISPMENU
ISPPLIB = ISP.SISPPENU
ISPSLIB = ISP.SISPSENU

* DD utilizado pelo TSO (IKJEFT01)
SYSPROC = #hlq.SFELPROC                                    * CRAALLOC
SYSTSIN = DUMMY
SYTSPRT = SYSOUT(H)

PROGRAM=IKJEFT01 %CRAALLOC ISPSTART PGM(CRASERV) PARM(&CRAPRM1.
  &CRAPRM2. &CRAPRM3. &CRAPRM4. &CRAPRM5. &CRAPRM6. &CRAPRM7.
  &CRAPRM8. )
```

Figura 6. crastart.conf: CRASTART com RAMs de amostra

Nota: Devido às limitações de largura da página, algumas linhas na amostra foram agrupadas sobre a linha seguinte. Todas as linhas que começam com uma indentação devem ser incluídas no final da linha anterior.

Customização da RAM adicional (opcional)

As RAMs customizadas possuem componentes adicionais que podem ser customizados, se necessário.

- O processamento da inicialização do CARMA possui uma saída opcional do usuário. Para obter mais informações, consulte a saída do usuário do CARMA (opcional).
- A inicialização da RAM customizada possui um executável de alocação, FEL.SFELPROC(CRAALLOC), que pode ser customizado. Para obter informações adicionais, consulte o CRAALLOC, o executável de alocação da RAM customizada.

Submissão em Lote com CA Endevor® SCM RAM

As informações nesta seção descrevem como configurar CARMA com as seguintes especificações:

- Inicialização do servidor: método de envio em lote. Esse método requer inicializadores do JES.
- RAM: CA Endevor® SCM RAM.

Esta etapa de customização pode ser omitida se você desejar usar um dos outros cenários com diferentes especificações.

Criar os Conjuntos de Dados VSAM do CARMA

Customize e envie as seguintes JCLs para definir e preencher os conjuntos de dados VSAM relacionados ao CARMA. Para instruções de customização, consulte a documentação no membro. Os conjuntos de dados VSAM existentes são substituídos.

Para obter detalhes adicionais sobre esta etapa, consulte “Conjuntos de Dados CARMA VSAM” na página 54.

- FEL.#CUST.JCL(CRA\$VCAD)
- FEL.#CUST.JCL(CRA\$VCAS)
- FEL.#CUST.JCL(CRA\$VMSG)

Customizar CRASRV.properties

O servidor RSE usa as configurações em /etc/zexpl/CRASRV.properties para iniciar e conectar a um servidor CARMA. É possível editar o arquivo com o comando do TSO **0EDIT**. Para que as mudanças entrem em vigor, a tarefa iniciada pelo RSED deve ser reiniciada.

Ao usar locais de arquivos padrão, a única mudança necessária é alterar o valor da diretiva `clist.dsname` para `FEL.#CUST.CNTL(CRASUBCA)`. Para obter informações adicionais sobre as diferentes diretivas, consulte “CRASRV.properties, a Interface RSE para CARMA” na página 46.

```
clist.dsname='FEL.#CUST.CNTL(CRASUBCA)'
```

Figura 7. CRASRV.properties: Envio em lote com o CA Endevor® SCM RAM

Customizar CRASUBCA

A CLIST `FEL.#CUST.CNTL(CRASUBCA)` e a JCL integrada enviam um servidor CARMA. As mudanças entram em vigor para todos os servidores CARMA que são iniciados após a atualização.

Para instruções de customização, consulte a documentação no membro. Para obter mais informações sobre o método de inicialização de envio em lote, consulte

“CRASUB*, a Inicialização do Servidor de Envio em Lote” na página 52.

```

PROC 8 CRAPRM1 CRAPRM2 CRAPRM3 CRAPRM4 CRAPRM5 CRAPRM6 CRAPRM7 CRAPRM8
SUBMIT * END($$)
//CRA&PORT JOB CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1)
//*
//RUN      EXEC PGM=IKJEFT01,DYNAMNBR=125,REGION=0M,TIME=NOLIMIT
//*
/* DD usado pelo RAM
//TYPEMAP DD DISP=SHR,DSN=FEL.#CUST.PARMLIB(CRATMAP)
//SHOWVIEW DD DISP=SHR,DSN=FEL.#CUST.PARMLIB(CRASHOW)
//CRACFG DD DISP=SHR,DSN=FEL.#CUST.PARMLIB(CRACFG)
/* uncomment CRABCFG and CRABSKEL to use batch actions
/*CRABCFG DD DISP=SHR,DSN=FEL.#CUST.PARMLIB(CRABCFG)
/*CRABSKEL DD DISP=SHR,DSN=FEL.#CUST.CNTL
/* uncomment and provide correct DSN to use Package Ship
/*APIHJC DD DISP=SHR,DSN=#shipjhc
//CONLIB DD DISP=SHR,DSN=CA.NDVR.CSIQLOAD
//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))
//EXT2ELM 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))
//C1MSGSG1 DD SYSOUT(H)
//C1EXMSGSG DD SYSOUT(H)
//MSG3FILE DD DUMMY
/*
/* DD utilizado pelo servidor CARMA (CRASERV)
/* preste atenção às autorizações APF ao usar STEPLIB
//STEPLIB DD DISP=SHR,DSN=FEL.SFELLOAD
//          DD DISP=SHR,DSN=CA.NDVR.CSIQAUTH
//          DD DISP=SHR,DSN=CA.NDVR.CSIQAUTU
//CRADEF DD DISP=SHR,DSN=FEL.#CUST.CRADEF
//CRAMSG DD DISP=SHR,DSN=FEL.#CUST.CRAMSG
//CRASTRS DD DISP=SHR,DSN=FEL.#CUST.CRASTRS
//CARMALOG DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
/*
/* DD usado pelo ISPF (via NDVRC1)
//ISPPROF DD DISP=(NEW,DELETE,DELETE),UNIT=SYSALLDA,
//          LRECL=80,RECFM=FB,SPACE=(TRK,(1,1,5))
//ISPCTL0 DD DISP=(NEW,DELETE,DELETE),UNIT=SYSALLDA,
//          LRECL=80,RECFM=FB,SPACE=(TRK,(5,5))
//ISPCTL1 DD DISP=(NEW,DELETE,DELETE),UNIT=SYSALLDA,
//          LRECL=80,RECFM=FB,SPACE=(TRK,(5,5))
//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
/*
/* DD utilizado pelo TSO (IKJEFT01)
//SYSPROC DD DISP=SHR,DSN=FEL.SFELPROC
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
%CRANDVRA NDVRC1 PGM(CRASERV) PARM(&CRAPRM1 &CRAPRM2 &STR(+))
&CRAPRM3 &STR(+)
&CRAPRM4 &STR(+)
&CRAPRM5 &STR(+)
&CRAPRM6 &STR(+)
&CRAPRM7 &STR(+)
&CRAPRM8 &STR(+) )
$$
EXIT CODE(0)

```

Figura 8. CRASUBCA: Envio em lote com o CA Endevor® SCM RAM

(Opcional) Customização Adicional do CA Endeavor® SCM RAM

O CA Endeavor® SCM RAM possui componentes adicionais que podem ser customizados, se necessário.

- O processamento da inicialização do CARMA possui uma saída opcional do usuário. Para obter informações adicionais, consulte “(Opcional) Saída do Usuário CARMA” na página 65.
- O CA Endeavor® SCM RAM possui vários arquivos de configuração, FEL.#CUST.PARMLIB(CRACFG), FEL.#CUST.PARMLIB(CRASHOW) e FEL.#CUST.PARMLIB(CRATMAP), que podem ser customizados. Para obter mais informações, consulte “CRACFG, CRASCL, CRASHOW e CRATMAP, os arquivos de configuração do CA Endeavor® SCM RAM” na página 57.
- O CA Endeavor® SCM RAM possui um executável de alocação, FEL.SFELPROC(CRANDVRA), que pode ser customizado. Para obter mais informações, consulte “CRANDVRA, o executável de alocação inicial de RAM do CA Endeavor® SCM” na página 58.
- O CA Endeavor® SCM RAM suporta a execução de ações CA Endeavor® SCM no modo em lote. As ações em lote requerem um arquivo de configuração, FEL.#CUST.PARMLIB(CRABCFG), uma JCL de estrutura básica, FEL.#CUST.CNTL(CRABATCA), e um cartão de tarefa padrão opcional, FEL.#CUST.CNTL(CRABJOBC), que devem ser customizados. Para obter mais informações, consulte “Ações em Lote do CA Endeavor® SCM RAM” na página 59.

Submissão em Lote com RAMs de Amostra

As informações nesta seção descrevem como configurar CARMA com as seguintes especificações:

- Inicialização do servidor: método de envio em lote, que requer inicializadores do JES
- RAM: RAMs de amostra, que não devem ser usados para propósitos de produção

Esta etapa de customização pode ser omitida se você desejar usar um dos outros cenários com diferentes especificações.

Criar os Conjuntos de Dados VSAM

Customize e envie as seguintes tarefas JCL para definir e preencher os conjuntos de dados VSAM relacionados ao CARMA. Para instruções de customização, consulte a documentação no membro. Os conjuntos de dados VSAM existentes são substituídos.

Para obter detalhes adicionais sobre esta etapa, consulte “Conjuntos de Dados CARMA VSAM” na página 54 e “Repository Access Managers (RAMs) CARMA” na página 55.

CARMA

- FEL.#CUST.JCL(CRA\$VDEF)
- FEL.#CUST.JCL(CRA\$VMSG)
- FEL.#CUST.JCL(CRA\$VSTR)

RAMs de Amostra

- FEL.#CUST.JCL(CRA#VPDS)

Customizar CRASRV.properties

O servidor RSE usa as configurações em `/etc/zexpl/CRASRV.properties` para iniciar e conectar a um servidor CARMA. É possível editar o arquivo com o comando do TSO **OEDIT**. Para que as mudanças entrem em vigor, a tarefa iniciada pelo RSED deve ser reiniciada.

Ao usar locais de arquivos padrão, a única mudança necessária é mudar o valor da diretiva `clist.dsname` para `FEL.#CUST.CNTL(CRASUBMT)`. Para obter informações adicionais sobre as diferentes diretivas, consulte “CRASRV.properties, a Interface RSE para CARMA” na página 46.

```
clist.dsname='FEL.#CUST.CNTL(CRASUBMT)'
```

Figura 9. CRASRV.properties: Envio em lote com RAMs de amostra

Customizar CRASUBMT

A CLIST `FEL.#CUST.CNTL(CRASUBMT)` e a JCL integrada enviam um servidor CARMA. As mudanças entram em vigor para todos os servidores CARMA que são iniciados após a atualização.

Para instruções de customização, consulte a documentação no membro. Para obter mais informações sobre o método de inicialização de envio em lote, consulte “CRASUB*, a Inicialização do Servidor de Envio em Lote” na página 52.

```

PROC 8 CRAPRM1 CRAPRM2 CRAPRM3 CRAPRM4 CRAPRM5 CRAPRM6 CRAPRM7 CRAPRM8
SUBMIT * END($$)
//CRA&PORT JOB CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1)
//*
//RUN      EXEC PGM=IKJEFT01,DYNAMNBR=125,REGION=0M,TIME=NOLIMIT
//*
//* DD utilizado pelo RAM
//CRARAM1 DD DISP=SHR,DSN=FEL.#CUST.CRARAM1          * PDS RAM
//*
//* DD usado pelo servidor CARMA (CRASERV)
//STEPLIB DD DISP=SHR,DSN=FEL.SFELLOAD
//CRADEF  DD DISP=SHR,DSN=FEL.#CUST.CRADEF
//CRAMSG  DD DISP=SHR,DSN=FEL.#CUST.CRAMSG
//CRASTRS DD DISP=SHR,DSN=FEL.#CUST.CRASTRS
//CARMALOG DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//*
//* DD usado pelo ISPF (ISPSTART)
//ISPPROF DD DISP=(NEW,DELETE,DELETE),UNIT=SYSALLDA,
//          LRECL=80,RECFM=FB,SPACE=(TRK,(1,1,5))
//ISPCTL0 DD DISP=(NEW,DELETE,DELETE),UNIT=SYSALLDA,
//          LRECL=80,RECFM=FB,SPACE=(TRK,(5,5))
//ISPCTL1 DD DISP=(NEW,DELETE,DELETE),UNIT=SYSALLDA,
//          LRECL=80,RECFM=FB,SPACE=(TRK,(5,5))
//ISPMLIB DD DISP=SHR,DSN=ISP.SISPMENU
//ISPPLIB DD DISP=SHR,DSN=ISP.SISPPENU
//ISPSLIB DD DISP=SHR,DSN=ISP.SISPSENU
//ISPTLIB DD DISP=SHR,DSN=ISP.SISPTENU
//*
//* DD usado pelo TSO (IKJEFT01)
//SYSPROC DD DISP=SHR,DSN=#hlq.SFELPROC          * CRAALLOC
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
%CRALLOC ISPSTART PGM(CRASERV) PARM(&CRAPRM1 &CRAPRM2 &STR(+
&CRAPRM3 &STR(+
&CRAPRM4 &STR(+
&CRAPRM5 &STR(+
&CRAPRM6 &STR(+
&CRAPRM7 &STR(+
&CRAPRM8 &STR(+
)
$$
EXIT CODE(0)

```

Figura 10. CRASUBMT: Envio em lote com RAMs de amostra

Customização da RAM adicional (opcional)

As RAMs customizadas possuem componentes adicionais que podem ser customizados, se necessário.

- O processamento da inicialização do CARMA possui uma saída opcional do usuário. Para obter mais informações, consulte a saída do usuário do CARMA (opcional).
- A inicialização da RAM customizada possui um executável de alocação, FEL.SFELPROC(CRAALLOC), que pode ser customizado. Para obter informações adicionais, consulte o CRAALLOC, o executável de alocação da RAM customizada.

Detalhes da Configuração do CARMA

Os diferentes cenários de configuração que são documentados nesta publicação compartilham vários arquivos de configuração do CARMA. Os detalhes desses arquivos de configuração são documentados aqui, e são mencionados em vários cenários.

CRASRV.properties, a Interface RSE para CARMA

O servidor CARMA fornece uma API padrão para outros produtos que usam sistemas host para acessar um ou mais Software Configuration Managers (SCMs). No entanto, ele não fornece métodos para comunicação direta com um computador cliente. Para esta comunicação, 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.

CRASRV.properties está localizado em /etc/zexpl/, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEL.SFELSAMP(FELSETUP). Para obter mais detalhes, consulte “Configuração da Customização” na página 15. É possível editar o arquivo com o comando do TSO **OEDIT**.

Nota: Para que as mudanças entrem em vigor, a tarefa iniciada pelo RSED deve ser reiniciada.

```
| # CRASRV.properties - CARMA configuration options
| #
| clist.dsname=''
| crastart.configuration.file=crastart.conf
| #port.start=0
| #port.range=100
| #user.exit='FEL.SFELSAMP(CRAEXIT) '
| #startup.script.name=carma.startup.rex
| #crastart.stub=CRASTART
| #crastart.syslog=Partial
| #crastart.timeout=420
| #crastart.steplib=FEL.SFELLPA
| #crastart.tasklib=TASKLIB
```

Figura 11. CRASRV.properties – Arquivo de Configuração CARMA

clist.dsname

Define o método de inicialização para o servidor CARMA. Para obter detalhes adicionais sobre os diferentes métodos de inicialização, consulte “Selecionar o Método de Inicialização do Servidor e o RAM Ativo” na página 34.

- *CRASTART indica que o servidor CARMA deve ser iniciado como uma subtarefa dentro do RSE utilizando CRASTART. Se você especificar *CRASTART, também deverá especificar as diretivas crastart.* ou usar seus valores padrão.
- *ISPF indica que o servidor CARMA deve ser iniciado usando o TSO/ISPF Client Gateway do ISPF. Esse método de inicialização está reprovado.
- Qualquer outro valor define o local de CRASUBMT CLIST, utilizando convenções de nomenclatura ao estilo TSO. Com aspas simples (') o nome do conjunto de dados é uma referência absoluta, sem aspas simples (') o nome do conjunto de dados é prefixado com o ID do

usuário do cliente, não com o prefixo do TSO. Esse último requer que todos os usuários do CARMA mantenham seu próprio CRASUBMT CLIST.

O padrão é uma sequência de caracteres nula para indicar que o CARMA não está configurado.

crastart.configuration.file

Especifica o nome do arquivo de configuração CRAFT. O padrão é `crastart.conf`. Esse arquivo especifica as alocações de conjuntos de dados e chamadas de programas que são necessárias para iniciar um servidor CARMA. Esta diretiva será usada apenas se a diretiva `clist.dsname` tiver `*CRAFT` como valor. O nome do arquivo pode ser especificado de várias maneiras:

- Sequência nula, que significa que a variável não é especificada. O valor padrão é utilizado.
- Apenas um nome do arquivo, que é o método padrão. O CARMA procura no diretório de configuração (`/etc/zexpl`, por padrão) para localizar o arquivo.
- Caminho relativo, que é o diretório e nome do arquivo, sem uma barra inicial (`/`). O CARMA inclui um diretório de configuração (`/etc/zexpl`, por padrão) no caminho fornecido para torná-lo um caminho absoluto.
- Caminho absoluto, que é o diretório e nome do arquivo, com uma barra inicial (`/`). O CARMA usa o local do arquivo especificado.

#port.start

Quando o valor de `port.start` for 0 (zero), o CARMA usará uma porta efêmera para comunicação entre o CARMA e o servidor RSE. Neste cenário, o TCP/IP designa um número da porta livre aleatório. Quando o valor de `port.start` for diferente de zero, ele será interpretado como o ponto de início de um intervalo de portas usado para comunicação entre o CARMA e o servidor RSE, neste caso, a variável `port.range` também deve ser definida. A porta padrão é 0. Para especificar o início do intervalo de portas, remova o comentário e customize. A comunicação nesta porta é limitada ao sistema host.

Nota: Antes de selecionar uma porta, verifique se ela está disponível no sistema usando os comandos **NETSTAT** e **NETSTAT PORTL**. Para obter mais informações, consulte "Portas TCP/IP reservadas" no *Referência de Configuração do Host (SC27-8578)*.

#port.range

Intervalo de portas, começando em `port.start`, que é usado para comunicação do CARMA, se `port.start` for diferente de zero. O padrão é 100. Para especificar o tamanho do intervalo de portas, remova o comentário e customize. Por exemplo, quando `port.start` for 5227 e `port.range` for 100, a porta 5227 até a 5326 (ambas inclusivas) poderão ser usadas pelo CARMA. Cada conexão do CARMA usa uma porta exclusivamente, portanto, a especificação de um intervalo de portas limita o número máximo de sessões do CARMA simultâneas.

#user.exit

Define o código especificado pelo usuário a ser executado durante a inicialização do CARMA. Remova o comentário e especifique o nome do conjunto de dados do código a ser executado.

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. Este último requer que todos os usuários do CARMA mantenham seu próprio código de saída.

Uma saída de usuário de amostra é fornecida como FEL.SFELSAMP(CRAEXIT). Essa amostra também documenta os argumentos de inicialização transmitidos para a saída do usuário. Para obter mais informações, consulte a saída do usuário do CARMA (opcional).

startup.script.name

Define o script de inicialização de CARMA. O padrão é carma.startup.rex. Este executável REXX aciona a inicialização de um servidor CARMA. O nome do arquivo pode ser especificado de várias maneiras:

- Sequência nula, que significa que a variável não é especificada. Neste caso, o valor padrão é usado.
- Apenas um nome do arquivo, que é o método padrão. O CARMA procura nos diretórios na variável de ambiente PATH para localizar o arquivo. O diretório que contém os executáveis do Developer for z Systems (/usr/lpp/ibm/rdz/bin, por padrão) é incluído automaticamente na variável de ambiente PATH.
- Caminho relativo, que é o diretório e nome do arquivo, sem uma barra inicial (/). O CARMA inclui um diretório de configuração (/etc/zexpl/, por padrão) no caminho fornecido para torná-lo um caminho absoluto.
- Caminho absoluto, que é o diretório e nome do arquivo, com uma barra inicial (/). O CARMA usa o local do arquivo especificado.

#crastart.stub

Stub do z/OS UNIX para chamar CRASTART. O padrão é CRASTART. Esse stub disponibiliza o módulo de carregamento CRASTART baseado em MVS para os processos do z/OS UNIX. Para especificar um caminho específico, remova o comentário e customize. Esta diretiva será usada apenas se a diretiva clist.dsname tiver *CRASTART como valor. O nome do arquivo pode ser especificado de várias maneiras:

- Sequência nula, que significa que a variável não é especificada. O valor padrão é utilizado.
- Apenas um nome do arquivo, que é o método padrão. O CARMA procura nos diretórios na variável de ambiente PATH para localizar o arquivo. O diretório que contém os executáveis do Developer for z Systems (/usr/lpp/ibm/rdz/bin, por padrão) é incluído automaticamente na variável de ambiente PATH.
- Caminho relativo, que é o diretório e nome do arquivo, sem uma barra inicial (/). O CARMA inclui um diretório de configuração (/etc/zexpl/, por padrão) no caminho fornecido para torná-lo um caminho absoluto.
- Caminho absoluto, que é o diretório e nome do arquivo, com uma barra inicial (/). O CARMA usa o local do arquivo especificado.

#crastart.syslog

Especifica quantas informações são gravadas no log do sistema, enquanto CRASTART inicia um servidor CARMA. O padrão é Partial. Os valores válidos são listados na tabela a seguir.

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

Para especificar o nível de detalhe necessário para mensagens de log do sistema, remova o comentário e customize. Esta diretiva será usada apenas se a diretiva `clist.dsname` tiver `*CRASTART` 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). Para especificar o valor de tempo limite necessário, remova o comentário e customize. Esta diretiva será usada apenas se a diretiva `clist.dsname` tiver `*CRASTART` como valor.

Nota: O encerramento de forma anormal 522 do sistema para o módulo CRASERV ocorrerá se o parâmetro JWT no membro parmlib SMFPRMxx estiver configurado para um valor inferior a `crastart.timeout` em `CRASRV.properties`. Esta ocorrência não impacta as operações do CARMA, porque o servidor é reiniciado automaticamente, se necessário.

#crastart.steplib

O local do módulo CRASTART quando acessado por meio da diretiva STEPLIB em `rdz.env`. O padrão é `FEL.SFELLPA`. Se o módulo CRASTART não puder fazer parte do LPA ou LINKLIST, remova o comentário e customize esta diretiva. Podem ocorrer problemas no controle de programa e no APF se o módulo CRASTART não estiver no LPA. Esta diretiva será usada apenas se a diretiva `clist.dsname` tiver `*CRASTART` como valor.

#crastart.tasklib

Nome alternativo para o nome TASKLIB DD em `crastart.conf`. O padrão é TASKLIB. Se o TASKLIB do nome DD tiver um significado especial para seu SCM ou RAM e não puder ser usado como substituição de STEPLIB, remova o comentário e customize esta diretiva. Esta diretiva será usada apenas se a diretiva `clist.dsname` tiver `*CRASTART` como valor.

crastart*.conf, A Inicialização do Servidor CRASTART

O RSE inicia o módulo de carregamento CRASTART, que usa as definições em `crastart*.conf` para criar um ambiente válido para executar comandos em lote TSO e ISPF. O Rational Developer for z Systems usa esse ambiente para executar o servidor CARMA, CRASERV.

`crastart*.conf` está localizado em `/etc/zexpl/`, a menos que tenha especificado um local diferente quando customizou e enviou a tarefa `FEL.SFELSAMP(FELSETUP)`. Para obter mais detalhes, consulte “Configuração da Customização” na página 15. É possível editar o arquivo com o comando do TSO **0EDIT**.

Nota: As mudanças entram em vigor para todos os servidores CARMA que são iniciados após a atualização.

O Developer for z Systems fornece diversos arquivos de configuração `crastart*.conf`. Cada um desses arquivos de amostra é pré-configurado para um cenário de customização específico:

- `crastart.endevor.conf` está configurado para inicialização de CRASTART com CA Endevor® SCM RAM.
- `crastart.conf` está configurado para inicialização de CRASTART com RAMs de amostra.

A função do arquivo `crastart*.conf` tem conceito semelhante ao de um fluxo de tarefas JCL, mas é mais restritiva.

- As amostras a seguir mostram formatos de linha válidos:
 - * comment
 - ddname=dsn1,dsn2,dsn3 * comment
 - ddname=SYSOUT(c) * comment
 - ddname=DUMMY * comment
 - -COMMAND=<any bpxwdyn command> * comment
 - PROGRAM = proname parms * comment

Nota: O comando **BPXWDYN** está documentado em *Using REXX and z/OS UNIX System Services* (SA22-7806) e permite construções de alocação complexas.

- Toda a entrada é alterada para maiúsculas.
- Continuações de linha não são suportadas.
- Não existe limitação quanto ao comprimento da linha.
- São permitidos um ou mais espaços em branco em torno do sinal de igual (=).
- Alocações DD devem preceder a instrução PROGRAM relacionada.
- Os nomes de DD alocados aqui são liberados no final da execução do programa. Eles não são acumulados.
- Os nomes de DD alocados pelos programas chamados não são liberados.
- Conjuntos de dados múltiplos podem ser concatenados para um nome DD. Os nomes de conjuntos de dados devem ser separados por uma vírgula (,), e a concatenação é procurada na ordem listada.
- Todas as alocações de conjuntos de dados são feitas com DISP=SHR, exceto para alocações feitas usando -COMMAND.
- Os dados sequenciais não são suportados. Todos os dados devem estar em arquivos catalogados.
- As variáveis podem ser usadas apenas no lado direito do sinal de igual (=).
- As variáveis a seguir são suportadas:

&CRAUSER.	ID do usuário de cliente
&CRADATE.	Data atual no formato Daaaadd (7 caracteres do calendário juliano)
&CRATIME.	Horário atual em formato Thhmmss (hora min seg)
&CRAPRM1.	Número da porta
Símbolo do sistema	Qualquer símbolo do sistema SYS1.PARMLIB(IEASYMxx)
-<ddname>	Um hífen (-) seguido por um nome de DD definido anteriormente age como uma referência de retrocesso *.ddname na JCL. O DD original deve ser alocado com o uso da instrução -COMMAND.

Nota: Não há nenhuma variável para o prefixo TSO, porque o TSO não está ativo quando o arquivo de configuração é interpretado. Se você precisar do prefixo TSO ou de outra variável que não esteja disponível, consulte "(Opcional) Executável de Alocação Customizado" na página 64.

Figura 12 na página 51 mostra uma base crastart*.conf que inclui serviços ISPF.

```

* DD utilizado pelo RAM

* DD utilizado pelo servidor CARMA (CRASERV)
TASKLIB = FEL.SFELLOAD
CRADEF = FEL.#CUST.CRADEF
CRAMSG = FEL.#CUST.CRAMSG
CRASTRS = FEL.#CUST.CRASTRS
CARMALOG = SYSOUT(H)
SYSPRINT= SYSOUT(H)

* DD utilizado pelo ISPF (ISPSTART)
-COMMAND=ALLOC FI(ISPCTL0) NEW DELETE DSORG(PS) RECFM(F,B) LRECL(80)
  BLKSIZE(32720) SPACE(5,5) TRACKS UNIT(SYSALLDA)
-COMMAND=ALLOC FI(ISPCTL1) NEW DELETE DSORG(PS) RECFM(F,B) LRECL(80)
  BLKSIZE(32720) SPACE(5,5) TRACKS UNIT(SYSALLDA)
-COMMAND=ALLOC FI(ISPPROF) NEW DELETE DSORG(PO) RECFM(F,B) LRECL(80)
  BLKSIZE(32720) SPACE(5,5) TRACKS UNIT(SYSALLDA) DIR(5)
ISPTABL = -ISPPROF
ISPTLIB = -ISPPROF,ISP.SISPTENU
ISPMLIB = ISP.SISPMENU
ISPPLIB = ISP.SISPPENU
ISPSLIB = ISP.SISPSENU

* DD utilizado pelo TSO (IKJEFT01)
SYSPROC = #hlq.SFELPROC
SYSTSIN = DUMMY
SYSTSPRT = SYSOUT(H)

PROGRAM=IKJEFT01 %CRAALLOC ISPSTART PGM(CRASERV) PARM(&CRAPRM1.
  &CRAPRM2. &CRAPRM3. &CRAPRM4. &CRAPRM5. &CRAPRM6. &CRAPRM7.
  &CRAPRM8. )

```

Figura 12. *crastart*.conf: Inicialização do servidor CARMA usando CRASTART*

Nota:

- Devido às limitações de largura da página, algumas linhas na amostra foram agrupadas sobre a linha seguinte. Todas as linhas que começam com uma endentação devem ser incluídas no final da linha anterior.
- Se você alterar a linha PROGRAM, assegure-se de que haja pelo menos um espaço em branco antes de fechar os parênteses(“”) da instrução PARM() para simplificar o processamento da sequência.
- 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.
- O nome de DD TASKLIB age como STEPLIB na JCL. Seu nome DD deve corresponder ao valor especificado para crastart.tasklib em CRASRV.properties, que é descrito em “CRASRV.properties, a Interface RSE para CARMA” na página 46.
- Regras regulares de APF se aplicam para alocações de TASKLIB. As bibliotecas perdem sua autorização de APF quando uma biblioteca autorizado não APF faz parte da concatenação.
- O encerramento anormal do sistema 522 para o módulo CRASERV ocorrerá se o parâmetro JWT no membro parmlib SMFPRMxx estiver configurado como um valor inferior ao valor crastart.timeout em CRASRV.properties. O encerramento anormal do sistema não impacta operações do CARMA, porque o servidor será reiniciado automaticamente, se necessário.
- Detalhes da inicialização do servidor CARMA são mostrados em rsecomm.log quando o servidor é encerrado. Para obter mais informações sobre como configurar o nível de detalhes de rsecomm.log, consulte o *IBM Explorer for z/OS Host Configuration Guide (SC27-8437)*.

Coletando Arquivos de Log CRASTART

CRASTART cria um ambiente TSO como um processo-filho de RSE, que é executado em um espaço de endereço separado. Ações não triviais podem ser necessárias para manter a saída do CARMA enviada para SYSOUT(*), que complica a coleta de arquivos de log. Esta dificuldade pode ser resolvida gravando os arquivos de log em um conjunto de dados específico do usuário, conforme mostrado na seguinte alocação de amostra:

```
-COMMAND=ALLOC FI(CARMALOG) MOD CATALOG DSORG(PS) RECFM(F,B) LRECL(133)
  BLKSIZE(27930) SPACE(5,5) TRACKS UNIT(SYSALLDA)
  DA(&CRAUSER..&SYSNAME..CRA.CARMALOG)
```

Nota:

- Devido às limitações de largura da página, algumas linhas na amostra foram agrupadas sobre a linha seguinte. Todas as linhas que começam com uma endentação devem ser incluídas no final da linha anterior.
- Para poder criar arquivos de log específicos do usuário, este arquivo de log deve ser alocado usando a instrução -COMMAND.
- Também é possível alocar os conjuntos de dados do log em um executável de alocação se você precisar de mais flexibilidade; por exemplo, envie o log apenas para um conjunto de dados para usuários específicos. Para obter mais informações sobre alocação de executáveis, consulte “(Opcional) Executável de Alocação Customizado” na página 64.

Se estiver gravando arquivos de log em SYSOUT, lembre-se de que o SYSOUT alocado por processos do z/OS UNIX é tratado como saída especial no JES. Isso é semelhante ao SYSOUT alocado pelas transações APPC.

- Enquanto o servidor CARMA ainda estiver ativo, a saída pode ser vista usando o comando **DA** no SDSF. A tarefa terá o ID do usuário seguido por um número de um dígito aleatório como nome da tarefa e um ID da tarefa STC. O usuário é o proprietário da tarefa.
- Se a saída foi gravada em uma classe de saída HOLD, quando o servidor CARMA for encerrado, devido à inatividade ou ao usuário encerrar a conexão, a saída pode ser vista usando os comandos **APPC ON** e **H ALL** no SDSF. O nome, o ID e o proprietário da tarefa permanecem os mesmos. Cada DD é mostrado como um arquivo de spool separado, sem nenhuma indicação de qual é esse DD.
- O Monitor de Tarefas JES também pode mostrar a saída se SEARCHALL=ON estiver ativo em FEJJCNFG e a saída residir no spool em uma classe de saída HOLD. Para obter mais informações sobre a diretiva SEARCHALL, consulte o *IBM Explorer for z/OS Host Configuration Guide (SC27-8437)*.

CRASUB*, a Inicialização do Servidor de Envio em Lote

O RSE inicia CLIST CRASUB* que, por sua vez, envia uma JCL integrada para criar um ambiente válido para executar comandos TSO e ISPF em lote. O Rational Developer for z Systems usa esse ambiente para executar o servidor CARMA, CRASERV.

CRASUB* está localizado em FEL.#CUST.CNTL, a menos que tenha especificado um local diferente quando customizou e enviou a tarefa FEL.SFELSAMP(FELSETUP). Para obter mais detalhes, consulte “Configuração da Customização” na página 15.

Nota: As mudanças entram em vigor para todos os servidores CARMA que são iniciados após a atualização.

O Developer for z Systems fornece várias tarefas de JCL CRASUB*. Cada um desses arquivos de amostra é pré-configurado para um cenário de customização específico:

- O CRASUBCA é configurado para inicialização em lote com o CA Endevor® SCM RAM.
- O CRASUBMT é configurado para inicialização em lote com RAMs de amostra.

Figura 13 mostra uma base CRASUB* que inclui serviços ISPF.

```
PROC 8 CRAPRM1 CRAPRM2 CRAPRM3 CRAPRM4 CRAPRM5 CRAPRM6 CRAPRM7 CRAPRM8
/* SET CRAPRM2=420
SUBMIT * END($$)
//CRA&PORT JOB CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1)
//*
//RUN      EXEC PGM=IKJEFT01,DYNAMNBR=125,REGION=0M,TIME=NOLIMIT
//*
//* DD usado pelo RAM
//*
//* DD usado pelo servidor CARMA (CRASERV)
//STEPLIB DD DISP=SHR,DSN=FEL.SFELLOAD
//CRADEF  DD DISP=SHR,DSN=FEL.#CUST.CRADEF
//CRAMSG  DD DISP=SHR,DSN=FEL.#CUST.CRAMSG
//CRASTRS DD DISP=SHR,DSN=FEL.#CUST.CRASTRS
//CARMALOG DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//*
//* DD usado pelo ISPF (ISPSTART)
//ISPPROF DD DISP=(NEW,DELETE,DELETE),UNIT=SYSALLDA,
//          LRECL=80,RECFM=FB,SPACE=(TRK,(1,1,5))
//ISPCTL0 DD DISP=(NEW,DELETE,DELETE),UNIT=SYSALLDA,
//          LRECL=80,RECFM=FB,SPACE=(TRK,(5,5))
//ISPCTL1 DD DISP=(NEW,DELETE,DELETE),UNIT=SYSALLDA,
//          LRECL=80,RECFM=FB,SPACE=(TRK,(5,5))
//ISPMLIB DD DISP=SHR,DSN=ISP.SISPMENU
//ISPPLIB DD DISP=SHR,DSN=ISP.SISPPENU
//ISPSLIB DD DISP=SHR,DSN=ISP.SISPSENU
//ISPTLIB DD DISP=SHR,DSN=ISP.SISPTENU
//*
//* DD usado pelo TSO (IKJEFT01)
//SYSPROC DD DISP=SHR,DSN=#FEL.SFELPROC          * CRAALLOC
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
%CRAALLOC ISPSTART PGM(CRASERV) PARM(&CRAPRM1 &CRAPRM2 &STR(+)
&CRAPRM3 &STR(+)
&CRAPRM4 &STR(+)
&CRAPRM5 &STR(+)
&CRAPRM6 &STR(+)
&CRAPRM7 &STR(+)
&CRAPRM8 &STR(+)
)
$$
EXIT CODE(0)
```

Figura 13. CRASUB*: Inicialização do CARMA usando o envio em lote

Nota:

- Se você alterar os dados do SYSTSIN, assegure-se de que haja pelo menos um espaço em branco antes de fechar os parênteses (“)”) da instrução PARM() para simplificar o processamento da sequência.
- 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, é possível mudar o valor de tempo limite do CARMA removendo o comentário e modificando a linha SET CRAPRM2=420 no CRASUB* CLIST. O valor de tempo limite é o número de segundos que o CARMA aguarda o próximo comando a partir do cliente. A configuração de um valor 0 resultará no valor de tempo limite padrão, atualmente, 420 segundos (7 minutos).
- Os detalhes do processo de inicialização do CARMA são mostrados em rsecomm.log quando o servidor é encerrado. Para obter mais informações sobre como configurar o nível de detalhes de rsecomm.log, consulte o *IBM Explorer for z/OS Host Configuration Guide (SC27-8437)*.

Conjuntos de Dados CARMA VSAM

O servidor CARMA requer acesso READ a três conjuntos de dados VSAM. Os membros de amostra para criar e preencher esses conjuntos de dados VSAM estão localizados em FEL.#CUST.JCL, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEL.SFELSAMP(FELSETUP). Para obter mais detalhes, consulte “Configuração da Customização” na página 15.

Nota:

- Se precisar mesclar as definições para uma RAM (customizada) em uma configuração VSAM existente, consulte a tarefa de amostra FEL.#CUST.JCL(CRA#UADD). Essa tarefa deve ser customizada e submetida para cada arquivo CARMA VSAM alterado. Para obter mais informações sobre a estrutura de registro usada por arquivos CARMA VSAM diferentes, consulte o *Common Access Repository Manager Developer's Guide (SC23-7660)*.
- Use a tarefa de amostra FEL.#CUST.JCL(CRA#UQRY) para extrair as definições ativas de um VSAM para um conjunto de dados sequencial.

CRADEF, o Conjunto de Dados de Configuração

Este conjunto de dados VSAM descreve as funções que são suportadas pelos RAMs definidos. Os desenvolvedores do RAM requerem acesso UPDATE a este conjunto de dados. O conjunto de dados pode ser criado por uma destas tarefas de amostra:

- CRA\$VCAD preenche o conjunto de dados com dados do CA Endevor® SCM RAM.
- CRA\$VDEF preenche o conjunto de dados com dados do RAM de amostra.

As tarefas de amostra mencionadas podem ser usadas para desativar um RAM definido durante a criação do VSAM. Fazer isso permite criar uma configuração do CARMA customizada usando um único arquivo de entrada principal, que pode ser um fornecido pela IBM ou customizado pelos desenvolvedores do RAM.

CRAMSG, o Conjunto de Dados de Mensagem

Esse conjunto de dados VSAM contém mensagens emitidas pelo próprio servidor CARMA. O conjunto de dados pode ser criado por uma destas tarefas de amostra:

- CRA\$VMSG preenche o conjunto de dados com dados do servidor genérico.

CRASTRS, o Conjunto de Dados de Sequência Customizado

Este conjunto de dados VSAM contém as mensagens que são emitidas pelos RAMs definidos. Os desenvolvedores do RAM requerem acesso UPDATE a este conjunto de dados. O conjunto de dados pode ser criado por uma destas tarefas de amostra:

- CRA\$VCAS preenche o conjunto de dados com dados do CA Endevor® SCM RAM.
- CRA\$VSTR preenche o conjunto de dados com dados do RAM de amostra.

Repository Access Managers (RAMs) CARMA

Um Repository Access Manager (RAM) é uma Interface de Programação de Aplicativos (API) para um Software Configuration Manager (SCM) baseado em z/OS. Por sua vez, o Developer for z Systems ou aplicativos gravados pelo usuário podem iniciar um servidor CARMA, que carrega os RAMs e fornece uma interface padrão para acessar qualquer SCM suportado.

O servidor CARMA deve estar apto a localizar os módulos de carregamento de RAM, por meio de LINKLIST ou STEPLIB/TASKLIB.

Os módulos de carregamento de RAM CRAR* que são fornecidos pelo Developer for z Systems estão localizados em FEL.SFELLOAD e o código-fonte de amostra e as tarefas de compilação estão localizados em FEL.SFELSAMP, a menos que você tenha usado um qualificador de alto nível diferente durante a instalação do SMP/E do Developer for z Systems.

As seções a seguir têm notas de customização para os RAMs que estão disponíveis com Developer for z Systems. Os membros de amostra referenciados estão localizados em FEL.#CUST.*, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa de amostra FEL.SFELSAMP(FELSETUP). Para obter mais detalhes, consulte “Configuração da Customização” na página 15.

Para obter um conhecimento detalhado do CARMA e obter informações adicionais sobre os RAMs de amostra e o código-fonte de amostra fornecido, consulte *Common Access Repository Manager Developer's Guide* (SC23-7660).

CA Endeavor® SCM RAM

- O CA Endeavor® SCM RAM é um RAM de tipo de produção.
- O CA Endeavor® SCM RAM fornece aos clientes Developer for z Systems acesso direto a elementos do CA Endeavor® SCM.
- O nome do módulo de carregamento é CRARNVDR.
- O CA Endeavor® SCM RAM possui muitas configurações adicionais em comparação com uma configuração convencional do CARMA. Use uma das configurações pré-configuradas que suportam o CA Endeavor® SCM RAM como um ponto de início e customize-a de acordo com suas necessidades.
- O CA Endeavor® SCM RAM tem diversos arquivos de configuração que podem ser customizados. Para obter mais informações, consulte “CRACFG, CRASCL, CRASHOW e CRATMAP, os arquivos de configuração do CA Endeavor® SCM RAM” na página 57.
- O CA Endeavor® SCM RAM possui um executável de alocação, FEL.SFELPROC(CRANDVRA), que pode ser customizado. Consulte “CRANDVRA, o executável de alocação inicial de RAM do CA Endeavor® SCM” na página 58 para obter informações adicionais.
- O CA Endeavor® SCM RAM suporta a execução de ações do CA Endeavor® SCM no modo em lote, no segundo plano. Para obter mais informações, consulte “Ações em Lote do CA Endeavor® SCM RAM” na página 59.

RAM de Pacotes do CA Endeavor® SCM

- O RAM de pacotes do CA Endeavor® SCM é um RAM de tipo de produção.
- O RAM de pacotes CA Endeavor® SCM fornece aos clientes Developer for z Systems acesso direto a pacotes CA Endeavor® SCM.
- O nome do módulo de carregamento é CRARPKGS.

- O RAM de pacotes do CA Endevor® SCM não possui configurações customizáveis e deve ser usado em conjunto com o CA Endevor® SCM RAM.

PDS RAM

- O PDS RAM é um RAM de amostra. Não use em um ambiente de produção.
- O PDS RAM fornece uma lista de conjuntos de dados semelhante a **MVS Arquivos -> Meus Conjuntos de Dados** na visualização Sistemas Remotos.
- O nome do módulo de carregamento é CRARPDS.
- O PDS RAM requer que os serviços ISPF estejam disponíveis.
- O PDS RAM requer que um conjunto de dados VSAM adicional seja alocado para DD CRARAM1. Esse conjunto de dados VSAM pode ser alocado e preparado com a tarefa de amostra FEL.#CUST.JCL(CRA#VPDS). Para instruções de customização, consulte a documentação no membro.
- Código-fonte e tarefas de compilação estão disponíveis em FEL.SFELSAMP. Para obter informações adicionais, consulte *Common Access Repository Manager Developer's Guide* (SC23-7660).

RAM de Base

- O RAM de esqueleto é um RAM de amostra. Não use em um ambiente de produção.
- O RAM de base fornece uma estrutura funcional que pode ser utilizada como ponto de início para o desenvolvimento de seu próprio RAM.
- O nome do módulo de carregamento é CRARTEST.
- Código-fonte e tarefas de compilação estão disponíveis em FEL.SFELSAMP. Para obter informações adicionais, consulte *Common Access Repository Manager Developer's Guide* (SC23-7660).

RAM do SCLM

- O SCLM RAM é um RAM de amostra. Não use em um ambiente de produção.
- O SCLM RAM fornece uma entrada básica para SCLM, o Software Configuration Manager do ISPF. Esse RAM não é ativado por padrão.
- O nome do módulo de carregamento é CRARSCLM.
- O SCLM RAM requer que os serviços ISPF estejam disponíveis.
- O SCLM RAM requer que um conjunto de dados VSAM adicional seja alocado para DD CRARAM2. Este conjunto de dados VSAM pode ser alocado e preparado com a tarefa de amostra FEL.#CUST.JCL(CRA#VSLM). Para instruções de customização, consulte a documentação no membro.
- O SCLM RAM requer que os vários conjuntos de dados específicos do usuário existam. Customize FEL.#CUST.JCL(CRA#ASLM) para alocar esses conjuntos de dados. Para instruções de customização, consulte a documentação no membro. Cada usuário deve submeter CRA#ASLM uma vez antes de usar o CARMA com SCLM RAM. Se isso não for feito, o resultado será um erro de alocação.
- O SCLM RAM não é ativado por padrão. Para ativar o RAM, ele deve ser definido nos conjuntos de dados VSAM CARMA referidos pelos DDs CRADEF e CRASTRS. Use a tarefa de amostra FEL.#CUST.JCL(CRA#UADD) para mesclar FEL.SFELVSM2(CRA0SLMD) em CRADEF e FEL.SFELVSM2(CRA0SLMS) em CRASTRS. Para instruções de customização, consulte a documentação no membro.
- Código-fonte e tarefas de compilação estão disponíveis em FEL.SFELSAMP. Para obter informações adicionais, consulte *Common Access Repository Manager Developer's Guide* (SC23-7660).

CRACFG, CRASCL, CRASHOW e CRATMAP, os arquivos de configuração do CA Endeavor® SCM RAM

Os componentes CARMA específicos do RAM do CA Endeavor® SCM podem ser customizados, independentemente do método de inicialização do servidor escolhido. Os membros de amostra referenciados abaixo estão localizados em FEL.#CUST.PARMLIB, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEL.SFELSAMP(FELSETUP). Para obter mais detalhes, consulte “Configuração da Customização” na página 15.

CRACFG, interação do CA Endeavor® SCM RAM com o SCM

O CRACFG especifica como o CA Endeavor® SCM RAM interage com o CA Endeavor® SCM. Consulte a documentação no membro para as instruções de customização se desejar alterar os padrões.

```
# ENTRY-STAGE-COPY-MODE = RETRIEVE-ADD
# ALTERNATIVE-ALLOC = YES
# PACKAGE-EDITING-OPTION = {READONLY | DISABLED}
# SCL-REQUIRED = YES
# SCL-DATASET-TEMPLATE = FEL.#CUST.PARMLIB(CRASCL)
# DYNAMIC-VB-DATASET-ALLOC = YES
# DATASET-ALLOC-OVERRIDE = SPACE(5,30) TRACK UNIT(SYSALLDA)
```

Figura 14. CRACFG - interação do CA Endeavor® SCM RAM com o SCM

CRASCL, SCL de modelo de RAM do CA Endeavor® SCM

CRASCL é um SCL (Software Control Language) de modelo que pode limitar quais ações e opções são permitidas quando Pacotes do CA Endeavor® SCM são processados.

Quando usado, apenas ações e opções explicitamente listadas no SCL de modelo têm permissão para serem especificadas no Editor de Pacotes do Developer for z Systems. Consulte a documentação no membro para as instruções de customização se desejar alterar os padrões.

Filtros Padrão CRASHOW, CA Endeavor® SCM RAM

O CRASHOW define os filtros padrão para os ambientes, sistemas e assim por diante do CA Endeavor®. Consulte a documentação no membro para as instruções de customização se desejar alterar os padrões.

```
ENV=*
TOENV=
STGID=*
TOSTGID=
SYS=*
SUBSYS=*
ELEM=*
TOELEM=
TYPE=*
#FILTER-DEP=YES
```

Figura 15. Filtros Padrão CRASHOW - CA Endeavor® SCM RAM

Nota: FILTER-DEP não é uma variável do CA Endeavor® SCM comum, mas uma variável específica do Developer for z Systems que controla varreduras de dependência para elementos com referências de área de cobertura para outros locais de repositório do CA Endeavor® SCM.

CRATMAP, os Mapeamentos de Extensões de Arquivos do CA Endevor® SCM RAM

O CRATMAP substitui o tipo do CA Endevor® SCM para mapeamentos de extensões de arquivos. Se desejar alterar os padrões, consulte as instruções de customização na documentação no membro.

```
# *      = cbl
# COBOL  = cbl
# COPY   = cpy
# ASM    = asm
# MACRO   = asm
# PROCESS = jcl
```

Figura 16. CRATMAP: Filtros padrão do CA Endevor® SCM RAM

CRANDVRA e CRADYNDA, os executáveis de alocação de RAM do CA Endevor® SCM

Os componentes CARMA específicos do RAM do CA Endevor® SCM podem ser customizados, independentemente do método de inicialização do servidor escolhido.

>É possível customizar uma cópia desses executáveis REXX de alocação se determinados padrões, como o nome do conjunto de dados, não corresponderem aos padrões do seu site. Os executáveis estão localizados em FEL.SFELPROC, a menos que você tenha usado um qualificador de alto nível diferente durante a instalação do SMP/E do Developer for z Systems.

Para instruções de customização, consulte a documentação no membro. Para obter mais informações sobre alocação de executáveis, consulte “(Opcional) Executável de Alocação Customizado” na página 64.

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 fazer isso, deve-se atualizar a referência para SFELPROC no SYSEXEC DD de seu método de inicialização do CARMA escolhido para corresponder ao nome do seu novo conjunto de dados.

CRANDVRA, o executável de alocação inicial de RAM do CA Endevor® SCM

O envio em lote e o método de inicialização CRASTART chamam o executável CRANDVRA REXX para alocar conjuntos de dados específicos do usuário usados pelo CA Endevor® SCM RAM. As alocações são feitas em um executável separado, porque um executável permite mais flexibilidade do que o que é possível na JCL CRASUBCA de envio em lote e no arquivo de configuração CRASTART crastart.endevor.conf. O executável de alocação também é responsável por chamar a saída opcional do usuário.

DD	Nome do conjunto de dados	Autenticação
DEPEND	&SYSPREF..&SYSUID.. &SYSNAME..CRA\$NDVR.DEPEND	Permanente
BROWSE	&SYSPREF..&SYSUID.. &SYSNAME..CRA\$NDVR.BROWSE	Temporário
BROWSEV	&SYSPREF..&SYSUID.. &SYSNAME..CRA\$NDVR.BROWSEV	Temporário

DD	Nome do conjunto de dados	Autenticação
ENHCEDIT	&SYSPREF..&SYSUID.. &SYSNAME..CRA\$NDVR.ENHCEDIT	Temporário
ENHCEDITV	&SYSPREF..&SYSUID.. &SYSNAME..CRA\$NDVR.ENHCEDITV	Temporário
C1PRINT	&SYSPREF..&SYSUID.. &SYSNAME..CRA\$NDVR.LISTING	Temporário
SPCLLIST	&SYSPREF..&SYSUID.. &SYSNAME..CRA\$NDVR.SPCLLIST	Temporário
PKGSCLS	&SYSPREF..&SYSUID.. &SYSNAME..CRA\$NDVR.PKGSCLS	Temporário
CRABJCLO	&SYSPREF..&SYSUID.. &SYSNAME..CRA\$NDVR.CRABJCLO	Temporário
CRAPARM	&SYSPREF..&SYSUID.. &SYSNAME..CRA\$NDVR.CRAPARM	Temporário

CRADYNDA, o executável de realocação de RAM do CA Endeavor® SCM

O CA Endeavor® SCM pode funcionar com conjuntos de dados bloqueados de variável com vários comprimentos de registro e requer que o Developer for z Systems use alocações idênticas para determinadas ações. Ambos, o envio de lote e o método de inicialização CRASTART, chamam o executável REXX CRADYNDA para alocar um conjunto de dados de trabalho com os atributos corretos.

Ações em Lote do CA Endeavor® SCM RAM

Normalmente, ações do CA Endeavor® SCM, como “Gerar Elemento”, são executadas “online”, no espaço de endereço do servidor CARMA. Este comportamento causa problemas se os procedimentos do CA Endeavor® SCM chamarem o TSO, porque o TSO já está ativo e isso significa que os DDs necessários, tais como SYSTSIN e SYSTSPRT, estão em uso.

Para resolver esse problema, o CA Endeavor® SCM RAM suporta “ações em lote”. Quando as ações em lote são ativadas, o CA Endeavor® SCM RAM envia uma tarefa em lote customizável para executar ações como “Gerar Elemento”. Usar uma tarefa em lote resulta na alocação de DDs, como SYSTSIN e SYSTSPRT, por procedimentos do CA Endeavor® SCM, porque a JCL enviada não requer que o TSO esteja ativo.

As ações de lote de RAM do CA Endeavor® SCM são equivalentes ao Developer for z Systems de ações do CA Endeavor® SCM de plano de fundo.

Quando uma solicitação é emitida para executar uma ação suportada por ações em lote, o CA Endeavor® SCM RAM verifica a existência do CRABCFG DD, no CRASUBCA ou crastart.endeavor.conf, e verifica se a configuração atrás deste DD é válida. Se CRABCFG existir e a configuração for válida, a ação será executada em lote. Se CRABCFG não existir, a ação será executada online. Os clientes do Developer for z Systems têm o recurso para substituir esse comportamento.

Por exemplo:

```
/* uncomment CRABCFG and CRABSKEL to use batch actions
/*CRABCFG DD DISP=SHR,DSN=FEL.#CUST.PARMLIB(CRABCFG)
/*CRABSKEL DD DISP=SHR,DSN=FEL.#CUST.CNTL
```

Nota:

- O ambiente livre do TSO está disponível apenas para ações selecionadas do CA Endeavor® SCM. As ações em lote não suportam um ambiente livre de TSO fora desse escopo.
- O arquivo de configuração CRABCFG documenta quais ações do CA Endeavor® SCM são suportadas.
- Uma tarefa de amostra funcional, FEL.#CUST.CNTL(CRABATCA), é fornecida para executar ações em lote, mas a intenção de ações em lote é que essa amostra seja customizada para iniciar os procedimentos atuais do CA Endeavor® SCM.
- Certifique-se de que existem inicializadores JES suficientes disponíveis na classe usada para submeter JCLs de ação em lote.
- Ao usar o JES em um ambiente SYSPLEX, assegure-se de que a tarefa seja executada no sistema atual, ou que as informações de conclusão sejam roteadas de volta para o sistema que está hospedando o Developer for z Systems, para que o CA Endeavor® SCM RAM possa verificar o status.
- O cliente do Developer for z Systems pode fornecer um cartão JOB customizado e instruções JCL adicionais para a JCL da ação em lote antes do envio.

CRABCFG, a Configuração da Ação em Lote do CA Endeavor® SCM RAM

CRABCFG define variáveis de configuração relacionadas a ações em lote do CA Endeavor® SCM RAM.

CRABCFG está localizado em FEL.#CUST.PARMLIB, a menos que tenha especificado um local diferente quando customizou e enviou a tarefa FEL.SFELSAMP(FELSETUP). Para obter mais detalhes, consulte “Configuração da Customização” na página 15.

Consulte o seguinte arquivo de amostra CRABCFG, que deve ser customizado para corresponder ao ambiente do sistema. As linhas de comentário começam com um sinal de número (#) ao usar uma página de códigos dos EUA. Os comentários sob uma diretiva e seu valor designado são suportados. Os espaços em torno do sinal de igual (=) são suportados. Continuações de linha não são suportadas.

Nota: As mudanças entram em vigor para todos os servidores CARMA que são iniciados após a atualização.

```
# Location of batch action JCL
SKELETON-DD = CRABSKEL
#
# batch action JCL members within SKELETON-DD
DEFAULT-JOBCARD = CRABJOBC
ADD-ELEMENT     = CRABATCA
DELETE-ELEMENT  = CRABATCA
GENERATE-ELEMENT = CRABATCA
MOVE-ELEMENT    = CRABATCA
RETRIEVE-ELEMENT = CRABATCA
PRINT-ELEMENT   = CRABATCA
PRINT-MEMBER    = CRABATCA
SIGNIN-ELEMENT  = CRABATCA
TRANSFER-ELEMENT = CRABATCA
#
# Command substitution key within batch action JCL
BSTIPT01-KEY = <CRA_BSTIPT01>
```

Figura 17. CRABCFG: Configuração de ação em lote do CA Endeavor® SCM RAM

SKELETON-DD

Nome da instrução DD que referencia um ou mais conjuntos de dados PDS(E) que contêm as JCLs do esqueleto da ação em lote. O valor de

amostra é CRABSKEL. Pode ser alterado se necessário. Este DD deve ser definido para o servidor CARMA no CRASUBCA ou crastart.endevor.conf.

DEFAULT-JOBCARD

Nome do membro que retém um cartão JOB padrão. Se não for indeferido por um cartão JOB específico do usuário armazenado no cliente Developer for z Systems, esse cartão JOB padrão será usado para substituir a chave <JOBCARD> em um JCL de estrutura básica. Pode ser alterado se necessário.

GENERATE-ELEMENT e Outras Ações do CA Endevor® SCM

Os nomes de chave representam ações do CA Endevor® SCM que são suportadas pela ação em lote e não podem ser alterados. O valor designado a cada valor é o nome do membro da JCL de esqueleto relacionado. O valor de amostra é CRABATCA para todas as chaves. Pode ser alterado se necessário.

BSTIPT01-KEY

A chave de substituição para a sequência de comando CA Endevor® SCM real. O valor de amostra é <CRA_BSTIPT01>. Pode ser alterado se necessário. A primeira ocorrência, mas não em um comentário, desta chave de substituição na JCL de esqueleto é substituída pela sequência de caracteres de comando que instrui o CA Endevor® SCM a executar a ação solicitada no elemento solicitado.

CRABATCA, a JCL de Ação em Lote do CA Endevor® SCM RAM

CRABATCA é um JCL de esqueleto de amostra usado pelas ações em lote. Para alterar os padrões, consulte as instruções de customização na documentação no membro.

CRABATCA está localizado em FEL.#CUST.CNTL, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEL.SFELSAMP(FELSETUP). Para obter mais detalhes, consulte “Configuração da Customização” na página 15.

As mudanças ficam ativas para todas as novas chamadas. A reinicialização do servidor não é necessária.

```
//<JOBCARD>
//*
//CRABATCA EXEC PGM=NDVRC1,DYNAMNBR=1500,REGION=4096K,PARM='C1BM3000'
//STEPLIB DD DISP=SHR,DSN=CA.NDVR.CSIQAUTU
// DD DISP=SHR,DSN=CA.NDVR.CSIQAUTH
//CONLIB DD DISP=SHR,DSN=CA.NDVR.CSIQLOAD
//C1MSG1 DD SYSOUT=*
//C1MSG2 DD SYSOUT=*
//C1PRINT DD SYSOUT=*,DCB=(RECFM=FBA,LRECL=133)
//SYSOUT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYMDUMP DD DUMMY
//SYSIN DD DUMMY
//BSTIPT01 DD *
SET STOPRC 16 .
<CRA_BSTIPT01>
/*
```

Figura 18. CRABATCA: JCL de ação em lote do CA Endevor® SCM RAM

CRABJOBC, o Cartão JOB de Ação em Lote do CA Endeavor® SCM RAM

CRABJOBC é um cartão JOB padrão de amostra usado para JCL de esqueleto de ação em lote que especifica a chave <JOB CARD>. Para alterar os padrões, consulte as instruções de customização na documentação no membro.

CRABJOBC está localizado em FEL.#CUST.CNTL, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEL.SFELSAMP(FELSETUP). Para obter mais detalhes, consulte “Configuração da Customização” na página 15.

As mudanças ficam ativas para todas as novas chamadas. A reinicialização do servidor não é necessária.

```
I //<USERID>B JOB CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1)
I //*PROCS JCLLIB ORDER=(COBOL.V5R1M0.SIGYPROC,CBC.SCCNPRC)
```

Figura 19. CRABJOBC: Cartão JOB de ação em lote do CA Endeavor® SCM RAM

CRAALLOC, o executável de alocação customizado de RAM

O envio em lote e o método de inicialização CRASTART chamam o executável CRAALLOC REXX para alocar conjuntos de dados específicos do usuário que podem ser usados por uma RAM gravada pelo usuário. As alocações são feitas em um executável separado, porque um executável permite mais flexibilidade do que é possível no envio em lote do arquivo de configuração CRASUBMT JCL e CRASTART crastart.conf. O executável de alocação também é responsável por chamar a saída opcional do usuário.

DD	Nome do conjunto de dados	Tipo
CRAPARM	&SYSPREF.&SYSUID.. &SYSNAME..CRA\$CUST.CRAPARM	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. CRAALLOC está localizado em FEL.SFELPROC, a menos que você tenha usado um qualificador de alto nível diferente durante a instalação do SMP/E do Rational Developer for z Systems.

Para instruções de customização, consulte a documentação no membro. Para obter informações adicionais sobre executáveis de alocação, consulte (Opcional) Executável de alocação customizada.

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 fazer isso, deve-se atualizar a referência para SFELPROC no SYSEXEC DD de seu método de inicialização do CARMA escolhido para corresponder ao nome do seu novo conjunto de dados.

Códigos de Retorno CARMA

O CARMA pode relatar vários códigos de erro ao cliente ou nos logs do sistema host. Os detalhes fornecidos com o erro e as informações na Tabela 12 na página 63 podem ajudá-lo a localizar o erro e trabalhar para uma resolução.

Tabela 12. Códigos de Retorno CARMA

Intervalo de erro	Tipo de erro
4-99	Erros genéricos do CARMA
100-199	Erros genéricos de RAM
200-399	Erros CRASERV (servidor CARMA)
400-499	Erros RSE (CARMA miner)
500-899	Erros específicos do RAM
900-999	Erros do TSO e TCP/IP

Alguns códigos de retorno comuns são os seguintes:

- 220: O servidor CARMA é encerrado devido ao tempo limite de inatividade. Isso não é um erro.
- 990: O servidor CARMA não pode conectar-se à porta na qual o extrator CARMA está recebendo.

(Opcional) Suportando Múltiplos RAMs

CARMA possui o recurso para definir vários RAMs e executá-los simultaneamente. No entanto, como há apenas um servidor CARMA ativo para um usuário, mesmo quando houver diversos RAMs, podem ser necessárias algumas mudanças na configuração para 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, identifica todos os RAMs definidos e envia as informações para o cliente CARMA. O usuário pode então selecionar um ou mais RAMs, que são carregados no servidor CARMA.

Como os RAMs estão ativos como plug-ins do servidor CARMA, certifique-se de que todos os pré-requisitos, como alocações do conjunto de dados para cada um dos RAMs, estejam disponíveis no espaço de endereço do servidor CARMA. Este requisito pode precisar de mudanças nas amostras de configuração do CARMA, como CRASUBMT ou crastart.conf, que estão incluídas no Developer for z Systems.

Exemplo

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

Definições para o CA Endevor® SCM RAM:

- FEL.SFELVSM2(CRA0VCAD): definições de CRADEF
- FEL.SFELVSM2(CRA0VCAS): definições de CRASTRS
- /etc/zexpl/crastart.endevor.conf: arquivo de configuração CRASTART

Definições para o PDS RAM:

- FEL.SFELVSM2(CRA0VDEF): definições de CRADEF
- FEL.SFELVSM2(CRA0VSTR): definições de CRASTRS
- FEL.#CUST.CRARAM1: definições de 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 que são específicos do PDS RAM dos membros SFELVSM2. Estes membros contêm definições para todos os RAMs de amostra, não apenas para o PDS RAM.
2. Mescle esses dados com os membros do CA Endeavor® SCM RAM SFELVSM2.
3. Crie uma lista de pré-requisitos específicos do PDS RAM:
 - DD CRARAM1, vinculado ao FEL.#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. O CRAMSG VSAM é específico para o servidor CARMA e, assim, idêntico para ambos os RAMs.
2. Inclua uma definição CRARAM1 para crastart.endevor.conf:
CRARAM1 = FEL.#CUST.CRARAM1
3. Verifique a instrução do PROGRAMA em crastart.endevor.conf para garantir que ele é capaz de fornecer o ambiente necessário por ambos os RAMs.
4. 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 Endeavor®.
 - %CRANDVRA: executável de pré-alocação do CA Endeavor® SCM RAM, localizado em FEL.SFELPROC, que aloca conjuntos de dados de trabalho temporários específicos do usuário.
 - NDVRC1: Backend do Endeavor®, que possui um mecanismo integrado para executar os comandos TSO e ISPF.
 - PGM(CRASERV): Comando para iniciar um servidor CARMA, em 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® é ativo em um ambiente ISPF, o que significa que o ambiente TSO requerido pelo PDS RAM também está disponível.

(Opcional) Executável de Alocação Customizado

Todos os métodos de inicialização do servidor CARMA possuem limitações relacionadas à alocação do conjunto de dados. Por exemplo, a substituição do prefixo TSO não está disponível na JCL ou CRASTART.

No entanto, ao criar um executável chamado após o início do TSO ou do ISPF e antes do início do CARMA, é possível usar todo o intervalo de variáveis e serviços disponíveis no TSO ou ISPF para fazer as alocações necessárias.

O Rational Developer for z Systems usa um executável de alocação em cada uma das configurações pré-configuradas descritas anteriormente nesse capítulo.

FEL.SFELPROC(CRANDVRA), o executável de alocação para CA Endevor® SCM RAM e FEL.SFELPROC(CRAALLOC), o executável de alocação para RAMs customizadas. O executável aloca conjuntos de dados temporários catalogados que tiveram o prefixo TSO do usuário como qualificador de alto nível. O executável de alocação também é responsável por chamar a saída opcional do usuário.

As instruções de customização são documentadas no executável. A mudança do executável de alocação é suportada, mas não aconselhada, pois as customizações devem ser refeitas quando o serviço PTF atualiza o executável. Se possível, use a saída do usuário CARMA, em vez disso, que é descrita em“(Opcional) Saída do Usuário CARMA”.

Nota:

- Ao atualizar o executável de alocação, assegure-se de não destruir as alocações feitas anteriormente no processo de inicialização do CARMA pelo CRASTART ou seu JCL de inicialização.
- A saída gerada pelo executável de alocação é mostrada no DD SYSTSPRT do servidor CARMA.

Ao atualizar o executável de alocação, assegure-se de não destruir as alocações feitas anteriormente no processo de inicialização do CARMA pelo CRASTART ou seu JCL de inicialização.

As amostras a seguir mostram como iniciar um executável de alocação que requer apenas o TSO.

crastart*.conf

```
SYSPROC = my.exec.library  
PROGRAM = IKJEFT01 %myexec ISPSTART PGM(CRASERV) PARM(&CRAPRM1. &CRAPRM2. )
```

CRASUB*

```
//SYSPROC DD DISP=SHR,DSN=my.exec.library  
//SYSTSIN DD *  
%myexec ISPSTART PGM(CRASERV) PARM(&CRAPRM1. &CRAPRM2. )  
//*
```

(Opcional) Saída do Usuário CARMA

O CARMA suporta a chamada de saída de um usuário para permitir inicialização especializada durante a inicialização e a limpeza especializada durante o encerramento do servidor CARMA. O uso de uma saída do usuário reduz a necessidade de alterar o executável de alocação, que é mantido pelo serviço PTF.

A saída do usuário é chamada pelo executável de alocação e é executada duas vezes. A chamada de inicialização é após a alocação dos conjuntos de dados temporários e antes de o servidor CARMA ser chamado. A chamada de limpeza é após o servidor CARMA terminar e antes de os arquivos temporários serem removidos. Se a primeira chamada terminar com código de retorno 99 ou superior, a inicialização do CARMA será interrompida. Isso quer dizer que nem o servidor CARMA nem a segunda chamada da saída desse usuário será executada.

Uma saída do usuário de amostra é fornecida como FEL.SFELSAMP(CRAEXIT), a menos que você tenha usado um qualificador de alto nível diferente durante a instalação do SMP/E do Developer for z Systems. Essa saída de amostra do usuário documenta em detalhe os argumentos da inicialização transmitidos para a saída do usuário:

Argumento de inicialização	Descrição
(STARTUP) (ENDING)	Indicador sobre se a chamada de saída é anterior ou posterior à chamada do servidor CARMA.
EXIT_RC=rc	Código de retorno da chamada anterior da saída. rc é sempre 0 durante a chamada (STARTUP).
CARMA_RC=rc	Código de retorno da chamada do servidor CARMA. rc é sempre 0 durante a chamada (STARTUP).
...	Comando de inicialização do servidor CARMA e argumentos de inicialização. Por exemplo, ISPSTART PGM(CRASERV) PARM(1312 420 EXIT=CRAEXIT CLIENT=9.0.1)

A saída gerada pela saída do usuário é mostrada no DD SYSTSPRT do servidor CARMA.

(Opcional) IRXJCL versus CRAXJCL

Se o servidor CARMA for iniciado usando o TSO (IKJEFTxx), poderão ocorrer problemas se seus RAMs chamarem serviços que, por sua vez, chamam a interface em lote IRXJCL REXX. O problema pode ocorrer quando os processadores chamados pelo RAM foram executados anteriormente sem o TSO, ou apenas no TSO online e alocaram 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 tentar alocar SYSTSIN ou SYSTSPRT, que é necessário para IRXJCL, porque o TSO em lote necessário para o 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. Em seguida, chama IRXJCL para executar o trabalho real.

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 são executados fora do TSO/CARMA, as alocações de SYSTSIN e SYSTSPRT são criadas por 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 por IRXJCL é gravada 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 em formato de origem, porque você deve customizá-lo para especificar as alocações específicas a serem usadas para SYSTSPRT. A alocação para SYSTSIN geralmente deve ser um conjunto de dados fictício.

O código-fonte do assembler de amostra e uma tarefa de compilação/ligação de amostra estão disponíveis como FEL.#CUST.ASM(CRAXJCL) e FEL.#CUST.JCL(CRA#CIRX), a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEL.SFELSAMP(FELSETUP). Para obter mais detalhes, consulte “Configuração da Customização” na página 15.

Customize o código-fonte do assembler CRAXJCL, conforme necessário, usando a documentação no membro. Em seguida, customize e envie a JCL CRA#CIRX para criar um módulo de carregamento CRAXJCL. Para instruções de customização, consulte a documentação no membro.

Se necessário, é possível renomear IRXJCL para algo diferente. Ajuste a origem CRAXJCL para chamar este novo nome para IRXJCL e compilá-lo, e, em seguida, renomear o módulo de carregamento CRAXJCL para IRXJCL. Esta configuração deve ser mais fácil do que alterar todas as suas chamadas para IRXJCL.

Capítulo 4. SCLM Developer Toolkit

O SCLM Developer Toolkit fornece as ferramentas necessárias para estender os recursos do SCLM para o cliente. O próprio SCLM é um gerenciador de código-fonte baseado em sistema host que é incluído como parte do ISPF.

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

Requisitos e Lista de Verificação

Você precisará de assistência de um administrador de SCLM e, opcionalmente, de um administrador de segurança para concluir esta tarefa de customização, que requer os seguintes recursos e tarefas de customização especiais:

- Atualizações de APF e LINKLIST
- Defina os tradutores de idiomas SCLM para suporte do Java EE
- Defina os tipos de SCLM para suporte de Java EE
- (Opcional) Regra de segurança para permitir que os usuários atualizem para um SCLM VSAM
- (Opcional) Instalação do Ant

Para começar a usar o SCLM Developer Toolkit em seu site, você deve executar as seguintes tarefas. A menos que especificado o contrário, todas as tarefas são obrigatórias.

1. Verifique e ajuste os pré-requisitos e atualizações de PARMLIB. Para obter detalhes, consulte “Pré-requisitos” na página 70.
2. Customize os arquivos de configuração do Developer for z Systems. Para obter detalhes, consulte:
 - “Atualizações do ISPF.conf para SCLMDT” na página 70
 - “Atualizações do rdz.env para SCLMDT” na página 71
3. Defina, opcionalmente, o suporte à conversão de nomes longos/curtos. Para obter detalhes, consulte “(Opcional) Tradução de Nome Longo/Abreviado” na página 72.
4. Opcionalmente, instale e customize o Ant para usar o suporte de construção do Java EE. Para obter detalhes, consulte “(Opcional) Instalar e Customizar Ant” na página 74.
5. Atualize o SCLM para definir partes específicas do SCLMDT. Para obter detalhes, consulte “atualizações SCLM para SCLMDT” na página 75.
6. Opcionalmente, configure a automação para limpar, periodicamente, a área de trabalho SCLMDT. Para obter detalhes, consulte “Remover Arquivos Antigos de WORKAREA e /tmp” na página 76.

Pré-requisitos

Para obter uma lista completa dos requisitos de software do Developer for z Systems, incluindo pré-requisitos e correquisitos, consulte a guia **Requisitos do sistema** na página do produto Developer for z Systems (<http://www-03.ibm.com/software/products/en/developerforsystemz/>) que também documenta as especificações de Ant necessárias para construções do Java EE no SCLM Developer Toolkit.

Atenção: O SCLM Developer Toolkit usa o Gateway ISPF do Legacy, o que significa que o z/OS 1.8 ou posterior é necessário. Isso também significa que se o Gateway ISPF Interativo for usado para o serviço de Comandos TSO, os Gateways ISPF de Legacy e Interativo deverão ser configurados.

Conforme descrito em “Alterações PARMLIB” na página 16, o SCLM Developer Toolkit requer customização adicional de configurações do sistema. Estas mudanças incluem os seguintes itens:

- (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. Os dois métodos requerem atenção adicional:

- O SDSF deve ser ordenado, instalado e configurado separadamente. .
- As configurações padrão para o comando TSO **OUTPUT** permitem que um usuário recupere apenas as saídas de tarefas que começam com esse ID do usuário específico. Para usar totalmente o recurso **OUTPUT**, a saída TSO/E de amostra IKJEFF53 talvez precise ser modificada para que um usuário possa recuperar a saída de tarefa pertencente ao usuário, mas ela não começa com o ID do usuário desse usuário. Para obter informações adicionais sobre essa saída, consulte *TSO/E Customization* (SA22-7783).

Os usuários requerem permissão READ, WRITE e EXECUTE para os diretórios /tmp/ e /var/zexpl/WORKAREA/ do z/OS UNIX. O diretório WORKAREA/ está localizado em /var/zexpl/, a menos que você tenha especificado um local diferente quando customizou o IBM Explorer for z/OS.

Atualizações do ISPF.conf para SCLMDT

O SCLM Developer Toolkit usa os esqueletos padrão do ISPF/SCLM, portanto, certifique-se de que a biblioteca de esqueletos ISP.SISPLIB esteja alocada para a concatenação ISPSLIB em ISPF.conf. O uso do conjunto de dados ISP.SISPSENU é opcional.

ISPF.conf está localizado em /etc/zexpl/, a menos que você tenha especificado um local diferente quando customizou o IBM Explorer for z/OS. É possível editar o arquivo com o comando do TSO **EDIT**.

Nota: As mudanças entram em vigor para todos os clientes que se conectam ao sistema 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 (,). .

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

* OPTIONAL:
*allocjob = ISP.SISPSAMP(ISPZISP2)
*ISPF_timeout = 900
```

Figura 20. 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. Para obter mais detalhes, consulte "Customizando o ambiente do TSO" no *IBM Explorer for z/OS Host Configuration Reference (SC27-8438)*.
- 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=hlq.USERSKEL,ISP.SISPSLIB
```

Atualizações do `rdz.env` para `SCLMDT`

O `SCLM Developer Toolkit` usa algumas diretivas configuradas em `rdz.env` para localizar conjuntos de dados e diretórios.

O `rdz.env` está localizado em `/etc/zexpl/`, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa `FEL.SFELSAMP(FELSETUP)`. Para obter mais detalhes, consulte "Configuração da Customização" na página 15. É possível editar o arquivo com o comando do TSO **EDIT**.

Nota: Para que as mudanças entrem em vigor, reinicie a tarefa iniciada `RSED` do *IBM Explorer for z/OS*.

A amostra de código a seguir mostra as diretivas `SCLMDT` em `rdz.env`, que devem ser customizadas para corresponder ao ambiente do sistema. Para obter mais detalhes sobre como customizar o `rdz.env`, consulte "rdz.env, o arquivo de configuração de ambiente" na página 27.

```
_SCLMDT_CONF_HOME=/var/rdz/scldmt
#STEPLIB=$STEPLIB:FEL.SFELAUTH:FEL.SFELLOAD
#_SCLMDT_TRANTABLE=FEL.#CUST.LSTRANS.FILE
#ANT_HOME=/usr/lpp/Apache/Ant/apache-ant-1.7.1
_SCLMDT_BASE_HOME=$RSE_HOME
_SCLMDT_WORK_HOME=$CGI_ISPHOME
CGI_DTWORk=$_SCLMDT_WORK_HOME
```

Figura 21. Atualizações do `rdz.env` para `SCLMDT`

(Opcional) Tradução de Nome Longo/Abreviado

O SCLM Developer Toolkit fornece a capacidade de armazenar arquivos de nomes longos no SCLM. Os nomes de arquivos longos são arquivos com nomes que possuem mais de 8 caracteres ou são compostos por letras maiúsculas e minúsculas. O armazenamento de nomes de arquivos longos é obtido 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 por meio 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

Para criar o VSAM de conversão de nomes longos/abreviados, customize e envie o membros de amostra FLM02LST na biblioteca de amostra ISP.SISPSAMP do ISPF. As etapas de configuração nesta publicação requerem que o VSAM seja chamado de FEL.#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. Mude todas as referências de FEL.#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 FEL.#CUST.LSTRANS.FILE
SET MAXCC=0
DEFINE CLUSTER(NAME(FEL.#CUST.LSTRANS.FILE) -
              VOLUMES(*****)) -
              RECORDSIZE(58 2048) -
              SHAREOPTIONS(3 3) -
              CYLINDERS(1 1) -
              KEYS(8 0) -
              INDEXED) -
DATA (NAME(FEL.#CUST.LSTRANS.FILE.DATA)) -
INDEX (NAME(FEL.#CUST.LSTRANS.FILE.INDEX))

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

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

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

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

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

```

Figura 22. FLM02LST: JCL de configuração de conversão de nomes longos/abreviados

Nota: Os usuários precisam de autoridade UPDATE para esse conjunto de dados VSAM.

Atualizações de rdz.env para tradução de nome longo/curto

Antes de usar a tradução de nome longo/curto, remova o comentário e configure a variável de ambiente `rdz.env _SCLMDT_TRANTABLE` para corresponder ao nome do VSAM de tradução de nome longo/curto.

`rdz.env` está localizado em `/etc/zexpl/`, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa `FEL.SFELSAMP(FELSETUP)`. Para obter mais detalhes, consulte “Configuração da Customização” na página 15. É possível editar o arquivo com o comando do TSO **0EDIT**.

Nota: Para que as mudanças entrem em vigor, reinicie a tarefa iniciada pelo RSED.

(Opcional) Instalar e Customizar Ant

Esta etapa é requerida somente se você planejar usar o suporte de construção do Java EE no SCLM.

Apache Ant é uma ferramenta de construção Java de software livre e pode ser transferida por download de <http://ant.apache.org/>. Ant consiste de arquivos de texto e de scripts, que são distribuídos no formato ASCII e, dessa forma, requerem uma conversão de ASCII/EBCDIC para executar no z/OS UNIX.

Execute as seguintes etapas para implementar Ant no z/OS e para defini-lo para o Developer for z Systems:

- Faça o download, em formato binário, do arquivo Ant compactado mais recente no sistema de arquivo z/OS UNIX. Faça download da versão .zip do ANT.
- 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 usando 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 deve existir em seu PATH z/OS UNIX local para usar o comando **jar**. Caso contrário, qualifique completamente o comando com a localização de bin Java (por exemplo, `/usr/lpp/java/J6.0/bin/jar -xf apache-ant-1.7.1.zip`).
- Converta todos os arquivos Ant em EBCDIC, customizando opcionalmente e executando o script de amostra `/usr/lpp/ibm/rdz/samples/BWBTRANT`.

Nota: Execute esse script apenas uma vez. Várias execuções corromperão a instalação do Ant.

- Para verificar uma conversão bem-sucedida, localize e abra um arquivo de texto no diretório ANT, como `apache-ant-1.7.1/README`. Se o arquivo for legível, isso indica que a conversão foi bem-sucedida.
- Use o comando **chmod -R 755 *** para permitir que todos os usuários leiam e executem arquivos no diretório ANT.
- Antes de usar Ant, configure as variáveis de ambiente `JAVA_HOME` e `ANT_HOME`.
 - `JAVA_HOME` (em `rse.env`) é necessário para apontar para o diretório inicial Java, por exemplo:
`JAVA_HOME=/usr/lpp/java/IBM/J6.0`
 - `ANT_HOME` (em `rse.env`) é necessário para apontar para o diretório inicial Ant, por exemplo:
`ANT_HOME=/usr/lpp/ibm/rdz/samples/BWBTRANT`

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/ibm/rdz/samples/BWBTRANT`
- `cat ./apache-ant-1.7.1/README`
- `chmod -R 755 *`
- `oedit /etc/rdz.env`

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:

```
export PATH=/usr/lpp/apache/Ant/apache-ant-1.7.1/bin:$PATH
export PATH=/usr/lpp/java/J6.0/bin:$PATH
```

- Para exibir a versão, se instalada com êxito, execute `ant -version`.

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. Para obter informações adicionais sobre as tarefas de customização necessárias, consulte *IBM Rational Developer for z Systems SCLM Developer Toolkit Administrator's Guide* (SC23-9801):

- Defina os tradutores de idiomas para suporte do Java EE
- Defina os tipos de SCLM para suporte de Java EE

Para concluir as tarefas de customização e de definição de projeto, o administrador do SCLM deve saber diversos valores customizáveis do Developer for z Systems, conforme descrito na Tabela 13.

Tabela 13. Lista de Verificação do Administrador de SCLM

Descrição	<ul style="list-style-type: none">• Valor-padrão• Onde encontrar a resposta	Valor
Biblioteca de amostra do Developer for z Systems	<ul style="list-style-type: none">• FEL.SFELSAMV• Instalação SMP/E	
Diretório de amostra do Developer for z Systems	<ul style="list-style-type: none">• /usr/lpp/ibm/rdz/samples• Instalação SMP/E	
Diretório bin Java	<ul style="list-style-type: none">• /usr/lpp/java/J6.0/bin• rse.env - \$JAVA_HOME/bin	
Diretório bin Ant	<ul style="list-style-type: none">• /usr/lpp/apache/Ant/apache-ant-1.7.1/bin• rdz.env - \$ANT_HOME/bin	

Tabela 13. Lista de Verificação do Administrador de SCLM (continuação)

Descrição	<ul style="list-style-type: none"> • Valor-padrão • Onde encontrar a resposta 	Valor
Diretório home WORKAREA	<ul style="list-style-type: none"> • /var/zexpl • rse.env - \$CGI_ISPWORK 	
Diretório home de configuração de projeto do SCLMDT	<ul style="list-style-type: none"> • /var/rdz/sc1mdt • rdz.env - \$_SCLMDT_CONF_HOME 	
VSAM de tradução de nomes longos/abreviados	<ul style="list-style-type: none"> • FEL.#CUST.LSTRANS.FILE • rdz.env - \$_SCLMDT_TRANTABLE 	

Remover Arquivos Antigos de WORKAREA e /tmp

O SCLM Developer Toolkit e o Gateway ISPF do Legacy compartilham o mesmo diretório WORKAREA e /tmp e ambos precisam de uma limpeza periódica. O z/OS UNIX fornece um shell script, skulker, que exclui arquivos com base no diretório no qual eles estão e em sua idade. Combinado com o daemon cron do z/OS UNIX, que executa comandos em datas e horas especificadas, é possível configurar uma ferramenta automatizada que limpa periodicamente os diretórios de destino. Consulte UNIX System Services Command Reference (SA22-7802) para obter mais informações sobre o script skulker e o daemon cron.

Capítulo 5. Depurador Integrado

O componente de host do Developer for z Systems Integrated Debugger permite que clientes da versão 9.0.1 e superior depurem vários aplicativos baseados no Language Environment (LE). O Integrated Debugger requer z/OS 1.10 ou superior.

Consulte a seção “Depurador integrado” do capítulo “Entendimento do Developer for z Systems” na *Host Configuration Reference (SC27-8578)* para obter uma visão geral do fluxo de dados do Depurador integrado.

Requisitos e Lista de Verificação

Você precisa de assistência de um administrador de Segurança, TCP/IP e CICS para concluir essa tarefa de customização, a qual requer os seguintes recursos ou tarefas de customização especiais:

- Inclua um SVC (IPL opcional)
- LPA atualizado para SVC
- Atualização de LINKLIST
- Autorização APF
- Definir tarefa iniciada
- Definir perfis de segurança e listas de acesso
- Reservar portas TCP/IP para comunicação confinada de cliente-host e host
- (Opcional) Atualizar JCL de região CICS
- (Opcional) Atualizar CICS CSD
- (Opcional) Atualizar JCL da região de mensagem de IMS
- (Opcional) Atualizar JCL do espaço de endereço do procedimento armazenado do DB2
- (Opcional) Atualizar procedimentos de construção remota ELAXF*

Para começar a usar o Integrated Debugger em seu site, você deve executar as seguintes tarefas. A menos que especificado o contrário, todas as tarefas são obrigatórias.

1. O Depurador integrado requer que a tarefa iniciada DBGMGR opcional esteja ativa (junto com as tarefas iniciadas RSED e obrigatórias a partir do IBM Explorer for z/OS). Consulte “DBGMGR, tarefa iniciada do gerenciador de depuração” na página 23 para obter a JCL de inicialização DBGMGR.
2. A configuração do Integrated Debugger é gerenciada pelos argumentos de inicialização da tarefa iniciada DBGMGR. Consulte “Parâmetros de configuração do Integrated Debugger” na página 80 para obter detalhes.
3. A tarefa iniciada DBGMGR requer que a biblioteca FEL.SFELAUTH seja autorizada pelo APF. Consulte o “Atualizações parmlib do Integrated Debugger” na página 80.
4. O Integrated Debugger depende de uma chamada do supervisor (SVC) do Developer for z Systems. O módulo de carregamento deve ser carregado na Link Pack Area (LPA) e pode ser dinamicamente ativado. Consulte “Atualizações parmlib do Integrated Debugger” na página 80 para obter detalhes.

5. O Integrated Debugger deve estar acessível em seu aplicativo e requer atualizações STEPLIB ou LINKLIST. Consulte “Atualizações parmlib do Integrated Debugger” na página 80 para obter detalhes.
6. O Integrated Debugger requer que o ID do usuário do aplicativo que está sendo depurado tenha um segmento OMVS válido. Consulte “Atualizações de segurança do Integrated Debugger” na página 82 para obter detalhes.
7. O Integrated Debugger requer algumas permissões de segurança. Consulte “Atualizações de segurança do Integrated Debugger” na página 82 para obter detalhes.
8. A tarefa iniciada DBGMGR requer algumas permissões de segurança. Consulte “Atualizações de segurança do Integrated Debugger” na página 82 para obter detalhes.

A seguinte etapa é requerida somente para transações do CICS de depuração:

1. O Integrated Debugger é capaz de depurar transações CICS. Isto requer que o Integrated Debugger e os Soquetes de IP do CICS são definidos para o CICS. Consulte “Atualizações de CICS do Integrated Debugger” na página 83 para obter detalhes.

O Integrated Debugger é capaz de usar uma saída de usuário do Language Environment (LE) que lê opções de tempo de execução TEST a partir de um conjunto de dados para um determinado módulo. Isso é bom para o código de depuração que está ativo nos subsistemas, como transações do IMS e procedimentos armazenados do DB2, uma vez que não é possível fornecer as opções de tempo de execução TEST dinamicamente para esse código. As etapas a seguir são requeridas apenas ao usar a saída de usuário do LE.

1. Crie a saída de usuário do LE e um ambiente de LE customizado que chama a saída de usuário. Consulte “Atualizações do Language Environment do Integrated Debugger” na página 82 para obter detalhes.
2. Defina o ambiente de LE customizado para sua transação do IMS. Consulte “Atualizações do Depurador Integrado IMS” na página 84 para obter detalhes.
3. Defina o ambiente de LE customizado para seu procedimento armazenado do DB2. Consulte “Atualizações de procedimentos armazenados do DB2 do Integrated Debugger” na página 84 para obter detalhes.

Nota:

- O Integrated Debugger usa o CICS CADP para fornecer opções de tempo de execução TEST para transações do CICS. Consulte sua documentação do CICS TS para obter mais informações sobre o CADP. No entanto, se desejar, também é possível usar o mecanismo de saída de usuário LE.
- O mecanismo de saída de usuário LE também pode ser usado para aplicativos regulares. Para obter detalhes, consulte “Atualizações de ELAXF* do Integrated Debugger” na página 84.

Integrated Debugger e COBOL v4

Para depurar programas gravados em COBOL v3.4 e v4, o Integrated Debugger precisa acessar um conjunto de dados de listagem (PDS ou PDS/E). O nome do conjunto de dados pode ser fornecido pela variável de ambiente AQE_DBG_V4LIST ou DD AQEV4LST. Se nenhuma delas estiver presente, o Integrated Debugger formará o nome do conjunto de dados substituindo o último qualificador do conjunto de dados do executável (por exemplo .LOAD) por .LISTING. Pergunte aos seus desenvolvedores que método é utilizável em seu site.

Integrated Debugger e Outros Depuradores Baseados em Ambientes de Linguagens

Normalmente, somente um depurador baseado em Ambiente de Linguagem (LE), como o Integrated Debugger, pode estar ativo em um determinado aplicativo, região do CICS, procedimento armazenado do DB2 ou transação IMS. Uma boa indicação de que um depurador é baseado em LE é que ele fornece um módulo de carregamento CEEVDBG ou alias que deve estar disponível para o aplicativo.

No entanto, o Depurador integrado pode coexistir com o IBM Debug Tool for z/OS, se o Depurador integrado for carregado primeiro pelo aplicativo.

Integrated Debugger e comunicação criptografada

Se o cliente do Developer for z Systems usar a criptografia para se comunicar com o daemon RSE, então o Debug Engine (baseado em cliente), por padrão, utilizará a criptografia para se comunicar com o Debug Manager (baseado em host). Por padrão, o Mecanismo de Depuração usará os mesmos certificados do cliente do Developer for z Systems.

Isso significa que a tarefa iniciada pelo daemon RSE e a tarefa iniciada pelo Debug Manager devam ser configuradas de forma semelhante em relação à criptografia de comunicação. Os cenários alternativos a seguir estão disponíveis para situações em que as configurações de criptografia para o daemon RSE e o Debug Manager diferirem:

- O Mecanismo de Depuração pode, após confirmação, tentar usar comunicação não criptografada se a configuração da sessão criptografada falhar.
- O Mecanismo de Depuração pode usar outros certificados de autenticação em vez do cliente do Developer for z Systems.

Diferentemente do daemon RSE, o Debug Manager não possui suporte nativo para comunicação criptografada. O Debug Manager depende de um serviço TCP/IP chamado Application Transparent Transport Layer Security (AT-TLS) para comunicação criptografada. Consulte “Configurando o AT-TLS” no *IBM Rational Developer for z Systems Host Reference Guide (SC27-8578)* para obter um guia de configuração passo a passo.

Depurador Integrado e SYSPLEX

O Developer for z Systems não é ciente de SYSPLEX e, portanto, requer que o programa a ser depurado esteja ativo no mesmo sistema ao qual o cliente do Developer for z Systems está conectado. Isso inclui sessões de depuração iniciadas por meio de uma tarefa em lote.

No z/OS 2.1, é possível especificar SYSAFF=* ou SYSTEM=* no cartão de TAREFA para impingir a execução da tarefa no sistema a que foi submetido. Em sistemas mais antigos deve-se especificar explicitamente o nome do sistema ao usar um JESPLEX para unir diversos subsistemas JES em um SYSPLEX.

Parâmetros de configuração do Integrated Debugger

O Depurador Integrado permite configurar as variáveis a seguir na JCL de inicialização DBGMGR. Consulte “DBGMGR, tarefa iniciada do gerenciador de depuração” na página 23 para obter a JCL de inicialização DBGMGR.

- A compensação do fuso horário, padrão EST5DST
- A porta usada para comunicação externa (cliente-host), padrão 5335
- A porta usada para comunicação interna (confinada no host), padrão 5336
- O número do SVC usado pelo Depurador Integrado, padrão 251
- O qualificador de alto nível da biblioteca de carregamento, padrão FEL

Atualizações parmlib do Integrated Debugger

- A tarefa iniciada DBGMGR deve estar ativa antes de os clientes do Developer for z Systems se conectarem ao host. Consulte “Incluir as Tarefas Iniciadas em COMMNDxx” na página 17 para obter detalhes sobre como iniciar o servidor automaticamente durante o tempo de carregamento inicial de programas.
- A tarefa iniciada DBGMGR requer que a biblioteca FEL.SFELAUTH seja autorizada pelo APF. Consulte “Autorizações de APF no PROGxx” na página 19 para obter detalhes.
- O Ambiente de Linguagem (LE) deve poder chamar o Depurador Integrado. Portanto, a biblioteca FEL.SFELAUTH deve ser colocada na LINKLIST, ou na STEPLIB do aplicativo a ser depurado. Consulte “Definições de LINKLIST no PROGxx” na página 20 para obter detalhes.

Nota:

- Ao usar LINKLIST, assegure-se de que FEL.SFELAUTH esteja localizado antes das bibliotecas de outros depuradores baseados em LE que contenham o módulo de carregamento CEEVDBG. Por exemplo, o IBM Debug Tool para z/OS usa bibliotecas hlq.SEQA*.
- Para evitar os conflitos, deve haver apenas um depurador baseado em LE definido em LINKLIST.
- O Integrated Debugger pode coexistir com o IBM Debug Tool for z/OS, se for carregado primeiro pelo aplicativo.
- Exceto para tarefa iniciada pelo Debug Manager, os módulos de carregamento do Depurador integrado em FEL.SFELAUTH não requerem execução autorizada. Os módulos de carregamento residem aqui, de modo que podem ser utilizados em um ambiente autorizado.
- O Integrated Debugger usa o Componente de Ligação do z/OS. Isto significa que SYS1.MIGLIB deva estar em LINKLIST (ou STEPLIB). Consulte “Definições de LINKLIST e LPA de Requisito” na página 22 para obter detalhes.
- O Depurador Integrado usa a API do z/OS Binder. Essa API está disponível desde o z/OS 1.10 como /usr/lib/iewbndd.so e desde o z/OS 1.13 também como SYS1.SIEAMIGE(IEWBNDD). Isso significa que para o z/OS 1.13 e superior, SYS1.SIEAMIGE deve estar na LINKLIST (ou STEPLIB). Consulte “Definições de LINKLIST e LPA de Requisito” na página 22 para obter detalhes.

Nota: Se SYS1.SIEAMIGE não estiver na LINKLIST ou STEPLIB nos sistemas z/OS 1.13 e superior, o Depurador Integrado emitirá a mensagem a seguir e tentará usar /usr/lib/iewbndd.so:

CEE3501S O módulo //IEWBNDD não foi localizado

- O Integrated Debugger requer que uma chamada do supervisor (SVC) do Developer for z Systems esteja definida em seu sistema. O número SVC padrão é 251. O módulo de carregamento relacionado, FEL.SFELLPA(AQESVC03), deve ser carregado em LPA e pode ser ativado dinamicamente. Consulte “Definições SVC no IEASVCxx” na página 18 e “Definições de LPA em LPALSTxx” na página 19 para obter detalhes.

Nota: Versões do Developer for z Systems anteriores à versão 9.1.1 usavam outro SVC, AQESVC01, que não está mais em uso. Ela poderá ser removida se o seu sistema host tiver apenas o Developer for z Systems versão 9.1.1 ou superior.

Atualizações do SVC do Depurador Integrado

O Depurador integrado requer que uma chamada do supervisor (SVC) do Developer for z Systems, FEL.SFELLPA(AQESVC03), esteja definida em seu sistema. Em circunstâncias ideais, esse SVC é ativado ou atualizado durante o IPL do sistema. No entanto, o IPL nem sempre é uma opção, portanto, o Depurador Integrado fará uma instalação ou atualização dinâmica usando a tarefa iniciada do Gerenciador de Depuração.

Durante a inicialização, a tarefa iniciada do Gerenciador de Depuração, DBGMGR, verificará os itens a seguir e, em seguida, executará a ação relacionada, conforme documentado em Tabela 14:

- se o SVC estiver definido
- versão de SVC ativo (se definido) e novo SVC (em SFELLPA)
- o argumento de inicialização SVC=svc_number está especificado, em que svc_number é o número SVC desejado

Tabela 14. Atualização dinâmica de SVC do Depurador Integrado

	SVC definido	SVC não definido
Versão de correspondência de SVC ativo e novo	<ul style="list-style-type: none"> • Nenhuma ação, SVC já está ativa e atual • Argumento de inicialização do SVC é ignorado 	<ul style="list-style-type: none"> • Defina dinamicamente o SVC usando o módulo de carregamento SFELLPA • Argumento de inicialização do SVC é obrigatório
A versão de SVC ativo e novo não corresponde	<ul style="list-style-type: none"> • Atualize o SVC usando o módulo de carregamento SFELLPA • Argumento de inicialização do SVC é ignorado 	<ul style="list-style-type: none"> • Defina dinamicamente o SVC usando o módulo de carregamento SFELLPA • Argumento de inicialização do SVC é obrigatório

Atualizações de TCP/IP do Integrated Debugger

O Depurador Integrado usa 2 portas TCP/IP.

- Porta para comunicação cliente-host (padrão 5335). A comunicação nesta porta pode ser criptografada.
- Porta para comunicação confinada pelo host (padrão 5336).

Atualizações de segurança do Integrated Debugger

O Depurador Integrado requer as seguintes definições de segurança. Consulte Capítulo 9, “Definições de segurança”, na página 103 para obter detalhes.

- Segmento OMVS para o ID do usuário que está em executando o aplicativo que está sendo depurado (isto inclui o ID do usuário da região CICS)
- Tarefa iniciada DBGMR
- Permissão BPX.SERVER para ID do usuário da tarefa iniciada
- Controle de programa para biblioteca de carregamento da tarefa iniciada
- Permissão AQE.AUTHDEBUG.STDPGM para usuários que depuram programas do estado de problema
- Permissão AQE.AUTHDEBUG.AUTHPGM para usuários que estão depurando programas de estado de problema e autorizados

Nota: Para simplificar a migração de uma configuração existente do Developer for z Systems sem o Depurador integrado, a JCL de amostra FEL.SFELSAMP(AQERACF) com comandos RACF é fornecida para definir somente as definições de segurança relacionadas ao Depurador integrado. A

Atualizações do Language Environment do Integrated Debugger

O Integrated Debugger é capaz de usar uma saída de usuário do Language Environment (LE) que lê opções de tempo de execução TEST a partir de um conjunto de dados para um determinado módulo. Isso é bom para o código de depuração que está ativo nos subsistemas, como transações do IMS e procedimentos armazenados do DB2, uma vez que não é possível fornecer as opções de tempo de execução TEST dinamicamente para esse código.

As etapas a seguir descrevem o fluxo do processo:

1. É solicitado que o LE execute um módulo.
2. O LE chama a saída de usuário com o nome do módulo como argumento.
3. A saída de usuário abre o conjunto de dados de opções de tempo de execução TEST e localiza as opções de tempo de execução TEST para o módulo.
4. A saída de usuário retorna as opções de tempo de execução TEST para LE.
5. O LE executa o módulo com as opções de tempo de execução TEST especificadas.

Nota:

- O IBM Debug Tool for z/OS usa uma versão estendida do mecanismo de saída de usuário descrito aqui. Para suportar a coexistência e evitar a duplicação do esforço para manter as opções de tempo de execução TEST, o Integrated Debugger usa os mesmos padrões que Debug Tool para as opções de saída de usuário e o nome do conjunto de dados de opções de tempo de execução TEST.
- A saída de usuário suporta a substituição de variável para o ID do usuário e o nome do módulo no nome do conjunto de dados de opções de tempo de execução TEST.
- O host e o cliente do Developer for z Systems são necessários para manter o conjunto de dados das opções de tempo de execução TEST com a GUI do cliente do Developer for z Systems.
- O Integrated Debugger usa o CICS CADP para fornecer opções de tempo de execução TEST para transações do CICS. Consulte sua documentação do CICSTS para obter mais informações sobre o CADP.

As etapas a seguir são requeridas apenas ao usar a saída de usuário do LE.

- Customize e envie AQED3CXT para criar a saída de usuário. AQED3CXT está em FEL.#CUST.JCL, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEL.SFELSAMP(FELSETUP).
- Customize e envie AQED3CEE para criar os módulos de carregamento do LE customizados que chamam a saída de usuário. AQED3CEE está em FEL.#CUST.JCL, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEL.SFELSAMP(FELSETUP). Por padrão, os módulos de carregamento do LE customizados (CEE*) são colocados em FEL.#CUST.LOAD.

Nota: Por motivos de desempenho, não coloque a biblioteca de carregamento que mantém os módulos de carregamento do LE customizados em LINKLIST. A saída de usuário pode ser chamada para cada módulo de carregamento ativado pelo LE, incluindo o software comprado.

Atualizações de CICS do Integrated Debugger

Para depurar as transações de CICS, o Depurador Integrado requer as seguintes atualizações de CICS.

- Atualizações de parâmetros de inicialização de sistema (SIT) CICS:
 - Especificar DEBUGT00L=YES.
 - Especificar TCPIP=YES.
 - Especificar LLACOPY=YES caso conte com LINKLIST para buscar um módulo de carregamento a partir da concatenação DFHRPL DD .
- Atualizações CICS JCL:
 - Especificar REGION=0M na instrução EXEC.
 - Definir a biblioteca de carregamento FEL.SFELAUTH na instrução DFHRPL DD da região. Se o parâmetro SIT LLACOPY=YES for especificado, a biblioteca também poderá residir em LINKLIST
 - Definir a biblioteca de carregamento SYS1.MIGLIB na instrução DFHRPL DD da região. Se o parâmetro SIT LLACOPY=YES for especificado, a biblioteca também poderá residir em LINKLIST.
 - Para o z/OS 1.13 e superior, defina a biblioteca de carregamento SYS1.SIEAMIGE na instrução DFHRPL DD da região. Se o parâmetro SIT LLACOPY=YES for especificado, a biblioteca também poderá residir em LINKLIST. Consulte as informações da API z/OS Binder no “Atualizações parmlib do Integrated Debugger” na página 80 para obter detalhes adicionais.

Nota:

- O ID do usuário da região CICS requer permissão UPDATE para o perfil CSVLLA.dataset na classe FACILITY para o parâmetro SIT LLACOPY=YES para funcionar conforme projetado.
 - Para depurar programas escritos em COBOL v4, o Depurador Integrado precisa acessar um conjunto de dados de listagem (PDS ou PDS/E). O nome do conjunto de dados pode ser fornecido pela variável de ambiente AQE_DBG_V4LIST ou DD AQEV4LIST. Se nenhuma delas estiver presente, o Depurador Integrado formará o nome do conjunto de dados substituindo o último qualificador do conjunto de dados do executável (por exemplo .LOAD) por .LISTING. Pergunte a seus desenvolvedores que método é utilizável em seu site.
- Atualizações da CSD do CICS:

Defina o depurador para uma região do CICS, conforme documentado na tarefa de atualização da CSD de amostra AQECSD. AQECSD está localizado em FEL.#CUST.JCL, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEL.SFELSAMP(FELSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes.

Nota:

- O Integrated Debugger requer que a comunicação baseada em TCP/IP seja ativada na região do CICS. O z/OS Communication Server IP CICS Sockets é o método preferido para executar isso. Para obter informações sobre a configuração deste recurso, consulte o Guia de Soquetes *Communication Server IP CICS (SC31-8807)*.
- O Integrated Debugger usa o CICS CADP para fornecer opções de tempo de execução TEST para transações do CICS. Para obter mais informações sobre o CADP, consulte a documentação do CICSTS.

Atualizações do Depurador Integrado IMS

Para depurar as transações do IMS, o Depurador Integrado requer as seguintes atualizações do IMS:

- Atualizações JCL da região de mensagens do IMS (DFSMPR):
 - Especificar REGION=0M na instrução EXEC da região.
 - Inclua a biblioteca de carregamento FEL.SFELAUTH na STEPLIB, se ela não estiver na LINKLIST. Não há necessidade de manter a autorização APF para todos os conjuntos de dados STEPLIST, a menos que ela seja necessária para outros produtos, além do Integrated Debugger.
 - Ao usar a saída de usuário do Language Environment (LE), inclua a biblioteca de carregamento que contém os módulos de carregamento da customizada (FEL.#CUST.LOAD padrão) na STEPLIB.

Atualizações de procedimentos armazenados do DB2 do Integrated Debugger

Para depurar procedimentos armazenados do DB2, o Integrated Debugger requer as seguintes atualizações relacionadas do DB2.

- Atualizações para JCL da tarefa iniciada para o espaço de endereço de procedimento armazenado do DB2:
 - Especifique REGION=0M na instrução EXEC.
 - Inclua a biblioteca de carregamento FEL.SFELAUTH na STEPLIB, se ela não estiver na LINKLIST. Não há necessidade de manter a autorização APF para todos os conjuntos de dados STEPLIST, a menos que ela seja necessária para outros produtos, além do Integrated Debugger.
 - Ao usar a saída de usuário do Language Environment (LE), inclua a biblioteca de carregamento que contém os módulos de carregamento da customizada (FEL.#CUST.LOAD padrão) na STEPLIB.

Atualizações de ELAXF* do Integrated Debugger

As seguintes atualizações são necessárias para usar o Depurador Integrado nos procedimentos de construção remotos fornecidos pelo Developer for z Systems. Para obter mais informações sobre a função e a localização desses procedimentos da JCL, consulte Referir-se ao “procedimentos de construção remota do ELAXF*” na página 24.

- Atualizações para os procedimentos JCL ELAXFDCL, ELAXFG0 e ELAXFTS0:
 - Especifique REGION=0M na instrução EXEC.
 - Inclua a biblioteca de carregamento FEL.SFELAUTH na STEPLIB, se ela não estiver na LINKLIST. Não há necessidade de manter a autorização APF para todos os conjuntos de dados STEPLIST, a menos que ela seja necessária para outros produtos, além do Integrated Debugger.
 - Ao usar a saída de usuário do Language Environment (LE), inclua a biblioteca de carregamento que contém os módulos de carregamento da customizada (FEL.#CUST.LOAD padrão) na STEPLIB.

Capítulo 6. Análise de código baseada em host

Semelhante ao cliente do Developer for z Systems, o host do Developer for z Systems suporta a execução das ferramentas de análise de código, que são fornecidas como um produto separado, o IBM Rational Developer for z Systems Host Utilities. Um benefício de executar a análise de código no host é que ele pode ser integrado em seu processamento em lote diário.

As ferramentas de análise de código a seguir estão disponíveis no host:

- Revisão de código: Usando regras com níveis de severidade diferentes, a revisão de código varre o código-fonte e relata violações de regra.
- Cobertura de código: Analise um programa em execução e gere um relatório de linhas que são executadas, comparado com o número total de linhas executáveis.

Requisitos e Lista de Verificação

Não é necessária a assistência de outros administradores para iniciar o uso das ferramentas da análise de código baseada em host em seu site, mas você deverá executar as tarefas a seguir. A menos que especificado o contrário, todas as tarefas são obrigatórias.

1. Instale o IBM Rational Developer for z Systems Host Utilities, conforme documentado no *Program Directory for IBM Rational Developer for z Systems Host Utilities* (GI13-2864). Quando usar os padrões fornecidos, o produto será instalado usando o qualificador de alto nível AKG e o caminho `/usr/lpp/rdzutil` do z/OS UNIX.
2. Crie cópias customizáveis das amostras fornecidas ao customizar e enviar o AKG.SAKGSAMP(AKGSETUP). Essa tarefa realiza as seguintes ações:
 - Crie o AKG.#CUST.PROCLIB e preencha-o com membros de amostra SYS1.PROCLIB.
 - Crie AKG.#CUST.JCL e preencha-o com JCL de configuração de amostra.

Nota: IBM Debug Tool for z/OS é um pré-requisito para o componente de cobertura do Código.

Revisão de Código

A revisão de código varre o código-fonte e relata violações de regra, usando regras com níveis de severidade diferentes. A ferramenta é fornecida com provedores de regra para Cobol e PL/I, mas outros provedores de regra podem ser incluídos.

O Developer for z Systems Host Utilities fornece um procedimento de amostra, AKGCR, para simplificar a chamada dos serviços de revisão de código no modo em lote. O AKGCR está localizado em AKG.#CUST.PROCLIB, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa AKG.SAKGSAMP(AKGSETUP).

Customize o procedimento de amostra, AKG.#CUST.PROCLIB(AKGCR), conforme descrito no membro e copie-o em SYS1.PROCLIB.

Se o procedimento AKGCR não puder ser copiado para uma biblioteca de procedimentos do sistema, solicite que os usuários do Developer for z Systems incluam um cartão JCLLIB logo após o cartão JOB em sua tarefa de chamada.

```
//MYJOB    JOB <job parameters>  
//PROCS    JCLLIB ORDER=(AKG.#CUST.PROCLIB)
```

Modificando o Processamento de Revisão de Código

A Revisão de Código do Developer for z Systems permite que um código de terceiro faça parte do processo de revisão. Por exemplo, é possível fornecer um provedor de regra para analisar o código C/C++, ou aprimorar o provedor de regra do Cobol para reconhecer convenções de codificação específicas do site.

A revisão de código baseada em host é um processo do Eclipse, assim como o cliente do Developer for z Systems. Portanto, os aprimoramentos feitos pela sua equipe de suporte de desenvolvimento para revisão de código no cliente podem ser reutilizados no host.

Os aprimoramentos consistirão de plug-ins do Eclipse ou de recursos do Eclipse. Para ativá-los, você deverá torná-los disponíveis para o código existente, conforme documentado na tarefa de configuração do AKGCRADD. O AKGCRADD está localizado em AKG.#CUST.JCL, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa AKG.SAKGSAMP(AKGSETUP).

Cobertura de Códigos

A cobertura de código analisa um programa em execução e gera um relatório de linhas que são executadas, comparado com o número total de linhas executáveis. Observe que a cobertura do Código configura uma conexão TCP/IP, usando uma porta efêmera, com o IBM Debug Tool for z/OS, ou com o Integrated Debugger Debug Engine de um cliente do Developer for z Systems.

O Developer for z Systems Host Utilities fornece duas maneiras de chamar a cobertura de código em modo em lote, um procedimento JCL de amostra, para processar uma única execução de programa e um conjunto de scripts para iniciar e parar um coletor de cobertura de código ativo que pode processar várias execuções de programa.

Chamada única de cobertura de código

O procedimento de amostra AKGCC fornece um método para iniciar um coletor de cobertura de código, fazer com que ele analise a execução de um único programa, parar o coletor e arquivar os resultados para uso posterior.

O AKGCC está localizado em AKG.#CUST.PROCLIB, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa AKG.SAKGSAMP(AKGSETUP).

Customize o procedimento de amostra, AKG.#CUST.PROCLIB(AKGCC), conforme descrito no membro e copie-o em SYS1.PROCLIB.

Se o procedimento AKGCC não puder ser copiado para uma biblioteca de procedimentos do sistema, solicite que os usuários do Developer for z Systems incluam um cartão JCLLIB logo após o cartão JOB em sua tarefa de chamada.

```
//MYJOB    JOB <job parameters>  
//PROCS    JCLLIB ORDER=(AKG.#CUST.PROCLIB)
```

Várias chamadas de cobertura de código

Dependendo de como o processo de desenvolvimento de software é configurado, a conveniência de fazer com que o procedimento AKGCC cuide de tudo pode não compensar o recurso e o uso de tempo para iniciar um coletor de cobertura de código para cada análise de programa.

Os Utilitários de Host do Developer for z Systems fornecem o script `ccstart` para iniciar um Coletor de cobertura de código que permanece ativo. Esse coletor pode, então, ser usado em várias chamadas de cobertura de código. O script `ccstop` pode ser usado para parar o coletor.

Estes scripts (`ccstart`, `ccstop` e `codecov`) estão localizados em `/usr/lpp/rdzutil/bin/` se os Utilitários de Host do Developer for z Systems estiverem instalados no local padrão. O uso destes scripts está documentado no Developer for z Systems IBM Knowledge Center (http://www-01.ibm.com/support/knowledgecenter/SSQ2R2/rdz_welcome.html).

O seguinte é um cenário de uso genérico:

1. Chame `ccstart` com a opção para usar um número de porta fixa e a opção para redirecionar a saída para um local conhecido.
2. Chame, com a frequência necessária, programas a serem analisados com a opção de inicialização `TEST(,,,TCP/IP&hostip>:<port>)`.
3. Chame `ccstop`.
4. Direcione o cliente do Developer for z Systems para o local de saída para ver os relatórios.

Saída da Cobertura de Código

A saída da cobertura de código deve ser importada em um cliente do Developer for z Systems e, portanto, é gravada em um arquivo do z/OS UNIX. A cobertura de código também pode usar os resultados de uma execução anterior e combiná-los com os resultados da execução atual, resultando em um relatório único que cobre diversos caminhos de código.

Por esses motivos, o Developer for z Systems Host Utilities não tenta remover a saída de uma execução de cobertura de código e a saída acumulará com o passar do tempo.

O z/OS UNIX fornece um shell script, `skulker`, que exclui arquivos de acordo com o diretório no qual eles estão e em seu tempo de existência. Combinado com o daemon `cron` do z/OS UNIX, que executa comandos em datas e horas especificadas, é possível configurar uma ferramenta automatizada que limpa periodicamente os diretórios de destino. Consulte o *UNIX System Services Command Reference* (SA22-7802) para obter mais informações sobre o script `skulker` e o daemon `cron`.

Capítulo 7. Outras tarefas de customização

Esta seção combina várias tarefas opcionais de customização. Para configurar o serviço necessário, siga as instruções na seção apropriada.

Customizações para arquivos de configuração do Developer for z Systems:

- “include.conf, Inclusões forçadas para assistência de conteúdo C/C++”

Customizações relacionadas ao Developer for z Systems em ou para outros produtos:

- “Subprojetos do z/OS UNIX” na página 92
- “Suporte a pré-processador de inclusão” na página 93
- “Suporte xUnit para Enterprise COBOL e PL/I” na página 94
- “Enterprise Service Tools, suporte” na página 95
- “Suporte para linguagem bidirecional do CICS” na página 95
- “Mensagens de diagnósticos IRZ para o Enterprise Service Tools” na página 96
- “FEKRNPLI, pré processador remoto PL/I” na página 97

include.conf, Inclusões forçadas para assistência de conteúdo C/C++

Esta tarefa de customização não requer assistência, recursos especiais ou tarefas de customização especiais.

O assistente de conteúdo para C/C++ pode usar as definições em include.conf para fazer inclusões forçadas de arquivos ou membros especificados. Uma inclusão forçada consiste em um arquivo ou diretório, um conjunto de dados ou um membro do conjunto de dados que será analisado quando uma operação do assistente de conteúdo for executada, independentemente de esse arquivo ou membro ter sido incluído ou não no código de origem usando uma diretiva de pré-processador.

O arquivo deve ser referenciado em rdz.env pelas variáveis include.c ou include.cpp antes de ser usada. Essa referência em rdz.env significa que você pode especificar um arquivo diferente para uso por C e C++. As variáveis em rdz.env estão desativadas por padrão.

A amostra include.conf está localizada em /etc/zexpl/, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEL.SFELSAMP(FELSETUP). Consulte “Configuração da Customização” na página 15 para obter mais detalhes. É possível editar o arquivo com o comando do TSO OEDIT.

As definições devem ser iniciadas na coluna 1. As linhas de comentário são iniciadas com um sinal numérico (#) ao usar uma página de códigos dos EUA. As linhas de dados só poderão ter o nome de um diretório, um arquivo, um conjunto de dados ou um membro. Não são permitidos comentários na mesma linha. Continuações de linha não são suportadas.

```

# Para incluir o arquivo stdio.h do diretório /usr/include, insira:
# /usr/include/stdio.h
#
# Para incluir todos os arquivos do diretório /usr/include e todos os seus
# subdiretórios, insira:
# /usr/include
#
# Remova o comentário e customize a variável FILETYPES para limitar a inclusão
# de curinga do z/OS UNIX em tipos de arquivos selecionados (distinção entre maiúsculas e minúsculas)
# Os tipos de arquivos são especificados em uma lista delimitada por vírgula (sem espaços em branco)
# FILETYPES=H,h,hpp,C,c,cpp,cxx

# Para incluir todos os membros do conjunto de dados CBC.SCLBH.H, insira:
# //CBC.SCLBH.H
#
# Para incluir o membro STDIOSTR do conjunto de dados CBC.SCLBH.H, insira:
# //CBC.SCLBH.H(STDIOSTR)
# A lista de amostra contém alguns arquivos de biblioteca padrão C usados comumente
/usr/include/assert.h
/usr/include/ctype.h
/usr/include/errno.h
/usr/include/float.h
/usr/include/limits.h
/usr/include/locale.h
/usr/include/math.h
/usr/include/setjmp.h
/usr/include/signal.h
/usr/include/stdarg.h
/usr/include/stddef.h
/usr/include/stdio.h
/usr/include/stdlib.h
/usr/include/string.h
/usr/include/time.h

```

Figura 23. include.conf - Inclusões forçadas para assistente de conteúdo C/C++

Subprojetos do z/OS UNIX

Esta tarefa de customização não requer assistência, recursos especiais ou tarefas de customização especiais.

REXEC (Execução Remota) é um serviço TCP/IP que permite que clientes executem um comando no sistema host. SSH (Secure Shell) é um serviço semelhante, mas todas as comunicações são criptografadas. O Developer for z Systems usa qualquer serviço para executar ações remotas (baseadas em host) em subprojetos z/OS UNIX.

Nota:

- O Developer for z Systems utiliza a versão de REXEC do z/OS UNIX, e não a versão do TSO.
- Se o REXEC/SSH não estiver configurado para utilizar a porta padrão, o cliente Developer for z Systems deve definir a porta correta a ser usada por subprojetos do z/OS UNIX. Esta configuração pode ser feita selecionando a página de preferências **Janela > Preferências > Soluções z/OS > Subprojetos do USS > Opções de Ação Remota**. Para saber qual porta é usada, consulte “Configuração de REXEC ou SSH” na página 93.

Configuração de REXEC ou SSH

REXEC e SSH dependem dos serviços fornecidos pelo INETD (Internet Daemon), que é outro serviço TCP/IP. O *Communications Server IP Configuration Guide* (SC31-8775) descreve as etapas necessárias para configurar INETD, REXEC e SSH. Para obter mais detalhes e métodos de configuração alternativos, consulte o white paper *Usando INETD, REXEC e SSH com Developer for System z* (SC14-7301), disponível na biblioteca do Developer for z Systems, <http://www-01.ibm.com/support/docview.wss?uid=swg27038517>.

Uma porta comum usada pelo REXEC é 512. Para verificar a porta que está sendo usada, consulte `/etc/inetd.conf` e `/etc/services`.

- 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`.

Suporte a pré-processador de inclusão

Esta tarefa de customização não requer assistência, recursos especiais ou tarefas de customização especiais.

O Developer for z Systems suporta a interpretação e expansão de instruções include COBOL e PL/I, incluindo instruções include selecionadas de terceiros. O Developer for z Systems também fornece um executável REXX de amostra, FEKRNPLI, que pode ser chamado pelo cliente Developer for z Systems para expandir a origem PL/I, chamando o compilador PL/I.

FEKRNPLI está localizado em `FEL.#CUST.CNTL`, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa `FEL.SFELSAMP(FELSETUP)`. Para obter mais detalhes, consulte Capítulo 2, “Customização Básica”, na página 15.

Customize o executável de amostra `FEL.#CUST.CNTL(FEKRNPLI)`, conforme descrito dentro do membro. Você deve fornecer as seguintes informações:

- `compiler_hlq`: O qualificador de alto nível para o PL/I Compiler

O cliente do Developer for z Systems usa o Serviço de Comando do TSO para executar o `exec`. Isso significa que se o `exec` FEKRNPLI for colocado na concatenação `SYSPROC` ou `SYSEXEC` para o Serviço de Comando do TSO, o usuário não precisará saber o local exato do `exec`. O usuário só precisa saber o nome. Ao usar o Gateway ISPF do Legacy, a concatenação `SYSPROC` ou `SYSEXEC` está definida em `ISPF.conf`. A customização desse arquivo está documentada no *IBM Explorer for z/OS Host Configuration Guide* (SC27-8437).

Suporte xUnit para Enterprise COBOL e PL/I

Esta tarefa de customização não requer assistência, mas requer os seguintes recursos ou tarefas de customização especiais:

- Atualização de LINKLIST
-

As estruturas que ajudam os desenvolvedores a compor código para executar testes de unidade repetidos e de autoverificação são conhecidas coletivamente como xUnit. O Developer for z Systems fornece essa estrutura para teste de unidade de código Enterprise COBOL e PL/I, chamada zUnit.

Para usar a estrutura zUnit, os desenvolvedores precisam de acesso aos módulos de carregamento AZU* e IAZU* na biblioteca de carregamento FEL.SFELLOAD, por meio de STEPLIB ou LINKLIST. O executor de teste zUnit, AZUTSTRN, por sua vez precisa de acesso a várias bibliotecas do sistema, por meio do STEPLIB ou LINKLIST:

- CEE.SCEERUN e CEE.SCEERUN2 (tempo de execução do LE)
- SYS1.CSSLIB (serviços do sistema de chamada)
- SYS1.SIXMLOD1 (kit de ferramentas XML)

O executor de teste do zUnit também precisa de acesso a uma biblioteca de carregamento que contém os diferentes casos de teste. Provavelmente esta biblioteca é exclusiva de um desenvolvedor.

O executor de teste do zUnit, AZUTSTRN, pode ser chamado pelo cliente Developer for z Systems no modo em lote, a partir da linha de comandos do TSO e a partir da linha de comandos do z/OS UNIX.

- O Developer for z Systems fornece um procedimento de amostra, AZUZUNIT, para simplificar a chamada do executor de teste do zUnit no modo em lote. AZUZUNIT está localizado em FEL.#CUST.PROCLIB, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEL.SFELSAMP(FELSETUP). Para obter mais detalhes, consulte “Configuração da Customização” na página 15.

Customize o procedimento de amostra, FEL.#CUST.PROCLIB(AZUZUNIT), conforme descrito dentro do membro, e copie-o para SYS1.PROCLIB.

O nome do procedimento e os nomes das etapas no procedimento correspondem às propriedades padrão que estão incluídas no cliente Developer for z Systems. Se o nome de um procedimento ou o nome de uma etapa em um procedimento for alterado, o arquivo de propriedades correspondente em todos os clientes deverá ser atualizado. Você não deve alterar os nomes de etapa e procedimento.

Se o procedimento AZUZUNIT não puder ser copiado para uma biblioteca de procedimentos do sistema, solicite que os usuários do Developer for z Systems incluam um cartão JCLLIB logo após o cartão JOB em sua tarefa de chamada.

```
//MYJOB    JOB <job parameters>  
//PROCS    JCLLIB ORDER=(FEL.#CUST.PROCLIB)
```

- Para chamar o executor de teste do zUnit a partir do z/OS UNIX (usando o script /usr/lpp/ibm/rdz/bin/zunit), é possível especificar os conjuntos de dados não LINKLIST necessários na diretiva STEPLIB de rdz.env, simplificando assim a configuração para o desenvolvedor.

rdz.env está localizado em /etc/zexpl/, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa

FEL.SFELSAMP(FELSETUP). Para obter mais detalhes, consulte “Configuração da Customização” na página 15. É possível editar o arquivo com o comando do TSO **EDIT**.

O script `zunit` permite que o usuário especifique conjuntos de dados que serão incluídos na diretiva `STEPLIB` usada pelo script.

- Para chamar o executor de teste do zUnit a partir da linha de comandos do TSO usando o executável `FEL.SFELPROC(FEKZUNIT)`, as bibliotecas do sistema devem existir em `LINKLIST`. Se não existirem, os desenvolvedores deverão especificar os nomes dos conjuntos de dados do sistema em cada instância de chamada do executor de teste do zUnit. Também é possível criar um executável wrapper que faça as alocações de **TSOLIB** desses conjuntos de dados para eles. É possível usar o próprio `FEKZUNIT` como um exemplo de como codificar este executável wrapper.

O executor de teste zUnit permite a reformatação automática de relatórios de teste. O Developer for z Systems fornece conversões de amostra (por exemplo, conversão para o formato Ant ou jUnit), que estão localizadas em `/usr/lpp/ibm/rdz/samples/zunit/xsd` e `/usr/lpp/ibm/rdz/samples/zunit/xsl`, se você instalou o Developer for z Systems no local `/usr/lpp/ibm/rdz` padrão.

Enterprise Service Tools, suporte

Esta tarefa de customização não requer assistência, recursos especiais ou tarefas de customização especiais.

O cliente Developer for z Systems possui um componente de geração de códigos chamado Enterprise Service Tools. Dependendo do tipo de código que está sendo gerado, este código depende das funções fornecidas pela instalação do sistema host do Developer for z Systems. A disponibilização destas funções do sistema host é descrita nas seções a seguir:

- “Suporte para linguagem bidirecional do CICS”
- “Mensagens de diagnósticos IRZ para o Enterprise Service Tools” na página 96

Nota: O Enterprise Service Tools inclui diversas ferramentas, como Service Flow Modeler (SFM) e XML Services for the Enterprise.

Suporte para linguagem bidirecional do CICS

Você precisará da assistência de um administrador do CICS para concluir esta tarefa de customização, que requer os seguintes recursos ou tarefas de customização especiais:

- Atualização de JCL da região do CICS
 - Definir um programa para o CICS
-

O componente Developer for z Systems Enterprise Service Tools suporta diferentes formatos de mensagens da interface em árabe e hebraico e a apresentação de dados bidirecionais e edição em todos os editores e visualizações. Em aplicativos do terminal, as telas da esquerda-à-direita e da direita-à-esquerda são suportadas e campos numéricos e campos com orientação oposta à tela.

A funcionalidade e os recursos bidirecionais adicionais incluem:

- O solicitante do serviço Enterprise Service Tools especifica de forma dinâmica os atributos bidirecionais das mensagens de interface.
- O processamento de dados bidirecionais em fluxos de serviços é baseado em atributos bidirecionais, como tipo de texto, orientação do 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 Enterprise Service Tools inclui a conversão dos dados entre os campos em mensagens que possuem atributos bidirecionais diferentes.

Além disso, o código gerado pelo Enterprise Service Tools pode suportar transformação bidi em ambientes que não sejam CICS SFR (Service Flow Runtime). Um exemplo são os aplicativos em lote. É possível criar os geradores do Enterprise Service Tools para incluir chamadas nas rotinas de conversão bidirecional especificando as opções de transformação bidirecional apropriadas nos assistentes de geração do Enterprise Service Tools e vinculando os programas gerados à biblioteca de conversão bidirecional apropriada, FEL.SFELLOAD.

Para ativar o suporte ao idioma Bidirecional do CICS, execute as seguintes tarefas:

1. Coloque os módulos de carregamento FEL.SFELLOAD, FEJBDCMP e FEJBDTRX, 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.

Importante: 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 esses módulos são DLLs e devem residir em uma biblioteca PDSE.

2. Defina FEJBDCMP e FEJBDTRX como os programas para CICS usando o comando CEDA adequado.

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

Mensagens de diagnósticos IRZ para o Enterprise Service Tools

Esta tarefa de customização não requer assistência, mas requer os seguintes recursos ou tarefas de customização especiais:

- Atualização de LINKLIST
 - Atualização de JCL da região do CICS
 - Atualização de CSD da Região CICS
-

O cliente Developer for z Systems possui um componente de geração de códigos chamado Enterprise Service Tools. Para que o código gerado pelo Enterprise Service Tools emita mensagens de erro de diagnóstico, todos os módulos IRZM* e IIRZ* na biblioteca de carregamento FEL.SFELMOD devem ser disponibilizados para o código gerado. O Enterprise Service Tools pode gerar código para os seguintes ambientes:

- CICS
- IMS
- lote MVS

Use as seguintes instruções quando o código gerado for executado em uma transação CICS.

- Inclua todos os módulos IRZM* e IIRZ* em FEL.SFELMOD para o DFHRPL DD da região CICS. Isso deve ser feito com a inclusão do conjunto de dados de instalação na concatenação para que a manutenção aplicada fique disponível automaticamente.
- Customize e submeta a tarefa IRZCSD para atualizar a Definição do Sistema CICS (CSD) para a região CICS. Para instruções de customização, consulte a documentação no membro. IRZCSD está localizado em FEL.#CUST.JCL, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEL.SFELSAMP(FELSETUP). Para obter mais detalhes, consulte “Configuração da Customização” na página 15.

Em todas as outras situações, disponibilize todos os módulos IRZM* e IIRZ* em FEL.SFELMOD por meio de STEPLIB ou LINKLIST. Isso deve ser feito com a inclusão do conjunto de dados de instalação na concatenação para que a manutenção aplicada fique disponível automaticamente.

Se você usar STEPLIB, defina os módulos não disponíveis por meio de LINKLIST na diretiva STEPLIB da tarefa que executa o código.

Se os módulos de carregamento não estiverem disponíveis e o código gerado encontrar um erro, a seguinte mensagem será emitida:

IRZ9999S Falha ao recuperar o texto de uma mensagem de tempo de execução do Ambiente de Linguagem. mensagem. Verifique se o módulo de mensagem de tempo de execução do Ambiente de Linguagem para recurso IRZ está instalado em DFHRPL ou em STEPLIB.

Nota:

- O módulo FEL.SFELMOD(IRZPWSIO) é vinculado estaticamente durante a geração descendente de códigos IMS MPP. Portanto, o módulo não deve estar disponível durante o tempo de execução do código gerado. Ele deve estar disponível apenas durante o tempo de compilação.
- Na versão 9.0.1, FEL.SFELMOD(IRZPWSIO) e o PL/I da amostra FEL.SFELSAMP(IRZPWSH) relacionada incluem o membro movido de Developer for z Systems para IMS Versão 12. As peças são renomeadas para IMS.SDFSRESL(DFSPWSIO) e IMS.SDFSMSPL(DFSPWSH) respectivamente.

FEKRNPLI, pré processador remoto PL/I

Esta tarefa de customização não requer assistência, mas requer os seguintes recursos ou tarefas de customização especiais.

O Developer for z Systems fornece um pré-processador de PL/I de amostra que pode ser chamado pelo cliente do Developer for z Systems.

O membro de amostra FEKRNPLI está localizado em FEL.#CUST.CNTL, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEL.SFELSAMP(FELSETUP). Para obter mais detalhes, consulte “Configuração da Customização” na página 15. Customize o executável do pré-processador PL/I de amostra (REXX) FEL.#CUST.CNTL(FEKRNPLI), conforme descrito no membro e notifique os usuários do Developer for z Systems em que o executável está localizado.

Capítulo 8. 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 as Tarefas Iniciadas

DBGMGR, o gerenciador de depuração

Inicie a tarefa iniciada DBGMGR opcional ou tarefa do usuário. O servidor emite a seguinte mensagem do console na inicialização bem-sucedida, onde `clientport` é o número da porta usada para comunicação externa (cliente-host) e `hostport` é o número de porta usado para comunicação interna (confinado no host).

```
AQECM001I Debug Manager startup complete (clientport/hostport)
```

Se a tarefa terminar com o código de retorno 66, então `FEL.SFELAUTH` não será autorizado pelo APF.

Nota: Inicie DBGMGR antes de continuar com os outros testes IVP relacionados à depuração.

Verificar os Serviços

A instalação do Developer for z Systems fornece vários Installation Verification Programs (IVP) para os serviços básicos e opcionais. Os scripts IVP estão localizados no diretório de instalação que, por padrão, é `/usr/lpp/ibm/rdz/bin/`.

Tabela 15. IVPs para Serviços

<code>fekfivpc</code>	"Conexão com o CARMA" na página 100
<code>fekfivps</code>	"Conexão SCLMDT" na página 100

As tarefas descritas nas seções a seguir requerem que você esteja ativo no z/OS UNIX. Isso pode ser feito emitindo o comando **OMVS** TSO. Para retornar ao TSO, use o comando **exit**.

É necessário um tamanho da região grande para o ID do usuário que executa os IVPs, porque são executadas funções como Java, que requerem uma grande quantidade de memória. Você deve configurar o tamanho da região para 131072 kilobytes (128 megabytes) ou mais.

O seguinte erro de amostra é uma indicação clara de um tamanho da região insuficiente, mas outros erros também podem ocorrer. 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: O z/OS Explorer e as tarefas iniciadas do Developer for z Systems devem estar ativas antes do início do teste IVP.

inicialização do IVP

Todos os comandos de amostra nesta seção requerem que algumas variáveis de ambiente sejam configuradas. Dessa forma, os scripts do IVP ficam disponíveis por meio 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 do RSE, como na amostra a seguir, em que \$ é o prompt do z/OS UNIX:

```
$ pwd
/u/userid
$ cd /etc/zexpl
$ ./ivpinit
-- RSE_CFG set to /etc/zexpl -- based on current location
-- RSE_HOME set to /usr/lpp/ibm/zexpl -- defined in $RSE_CFG/rse.env
-- added product $PATH to PATH
-- PATH=/etc/zexpl:/bin:/usr/lpp/java/J6.0/bin:/usr/lpp/ibm/zexpl/bin:/usr/lpp/i
spf/bin:/bin:/usr/lpp/ibm/rdz/bin:/bin
```

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 (.) refere-se ao diretório atual.

Nota:

- Se **./ivpinit** não for executado antes dos scripts **fekfivp***, o caminho para estes scripts deverá ser especificado quando eles forem chamados, como na amostra a seguir:

```
/usr/lpp/ibm/rdz/bin/fekfivpc
```

Além disso, se **./ivpinit** não for executado primeiro, todos os scripts **fekfivp*** solicitarão o local do diretório que contém os arquivos de configuração customizados

Conexão com o CARMA

Verifique a conexão com o CARMA executando o seguinte comando:

```
fekfivpc
```

O comando deve retornar uma mensagem de êxito:

Nota: Se o IVP falhar, verifique o conteúdo de **/tmp/fekfivpc.log**. Este log documenta a comunicação entre o RSE e o CARMA e pode conter informações que ajudam a localizar a causa raiz da falha.

fekfivpc tem os seguintes parâmetros opcionais não posicionais:

-noram

Por padrão, **fekfivpc** inicia o primeiro RAM que está definido no conjunto de dados CRADEF VSAM. Pode haver situações em que você não deseja testar o RAM; por exemplo, um RAM de terceiros é listado primeiro e requer entrada inesperada. Nesses casos, é possível usar o argumento de inicialização **-noram** para omitir as etapas específicas da RAM do teste IVP.

Conexão SCLMDT

Verifique a conexão com o SCLM Developer Toolkit executando o seguinte comando:

```
fekfivps
```

O comando deve retornar uma mensagem de êxito.

fekfivps tem os seguintes parâmetros opcionais e não posicionais:

-file fekfivps pode produzir grandes quantidade de saída, chegando a centenas de linhas. O parâmetro `-file` envia sua saída para um arquivo, `$TMPDIR/fekfivps.log`, em que `$TMPDIR` é o valor da diretiva `TMPDIR` em `rse.env`, que, por padrão, é `/tmp`.

-debug

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

Capítulo 9. Definições de segurança

Customize e envie a tarefa FELRACF de amostra, que possui comandos RACF de amostra para criar as definições de segurança básica para Developer for z Systems. Customize e envie a tarefa AQERACF de amostra, que possui comandos RACF de amostra para criar as definições de segurança para o Depurador integrado.

FELRACF e AQERACF estão localizados em FEL.#CUST.JCL, a menos que você tenha especificado um local diferente quando customizou e enviou a tarefa FEL.SFELSAMP(FELSETUP). Para obter detalhes adicionais, consulte “Configuração da Customização” na página 15.

Consulte o *RACF Command Language Reference (SA22-7687)*, para obter mais informações sobre comandos RACF.

Requisitos e Lista de Verificação

Para concluir a configuração de segurança, o administrador de segurança deve saber os valores que estão listados na Tabela 16. Esses valores foram definidos durante as etapas anteriores da instalação e customização do Rational Developer for z Systems.

Tabela 16. Variáveis de configuração de segurança

Descrição	<ul style="list-style-type: none">• Valor-padrão• Onde encontrar a resposta	Valor
Developer for z Systems qualificador de alto nível do produto	<ul style="list-style-type: none">• FEL• Instalação SMP/E	
Developer for z Systems qualificador de alto nível de customização	<ul style="list-style-type: none">• FEL.#CUST• FEL.SFELSAMP(FELSETUP), conforme descrito em “Configuração da Customização” na página 15.	
Nome da tarefa iniciada do Depurador Integrado	<ul style="list-style-type: none">• DBGMGR• FEL.#CUST.PROCLIB(DBGMGR), conforme descrito em “Alterações do PROCLIB” na página 23	

A lista a seguir é uma visão geral das ações necessárias para concluir a configuração de segurança básica do Developer for z Systems. Conforme documentado nas seções a seguir, métodos diferentes podem ser usados para preencher esses requisitos, dependendo do nível de segurança necessário.

- “Ativar Configurações e Classes de Segurança” na página 104
- “Definir as Tarefas Iniciadas do Developer for z Systems” na página 104
- “Definir o Debug Manager como um servidor z/OS UNIX seguro” na página 105
- “Definir as bibliotecas controladas pelo programa MVS para o Debug Manager” na página 105

- “Definir acesso ao Integrated Debugger” na página 106
- “Definir os Perfis do Conjunto de Dados” na página 106
- “Verificar as Configurações de Segurança” na página 107

Ativar Configurações e Classes de Segurança

O Developer for z Systems usa uma variedade de mecanismos de segurança para garantir um ambiente do sistema host seguro e controlado para o cliente. Para fazer isto, várias classes e configurações de segurança devem estar ativas, conforme mostrado com os seguintes comandos RACF de amostra:

- Exibir configurações atuais
 - SETROPTS LIST
- Ativar classe de instalação para Depurador Integrado
 - SETROPTS GENERIC(FACILITY)
 - SETROPTS CLASSACT(FACILITY) RACLIST(FACILITY)
- Ativar definições de tarefa iniciada para Depurador Integrado
 - SETROPTS GENERIC(STARTED)
 - RDEFINE STARTED ** STDATA(USER(=MEMBER) GROUP(STCGROUP) TRACE(YES))
 - SETROPTS CLASSACT(STARTED) RACLIST(STARTED)
- Ativar controle de programa para Depurador Integrado
 - 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 obscurece e complica o caminho da procura usado pelo software de segurança. Nesse caso, você deve mesclar as definições * existentes com a ** nova. Use 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 este controle de programa antes de ativá-lo em um sistema de produção.

Definir as Tarefas Iniciadas do Developer for z Systems

Os comandos RACF de amostra a seguir criam a tarefa iniciada DBGMGR, com ID do usuário protegido (STCDBM) e o grupo STCGROUP designado a ele.

- ```
ADDGROUP STCGROUP OMVS(AUTOUID)
DATA('GROUP WITH OMVS SEGMENT FOR STARTED TASKS')
```
- ```
ADDUSER STCDBM DFLTGRP(STCGROUP) NOPASSWORD NAME('DEBUG MANAGER')
OMVS(AUTOUID HOME(/tmp) PROGRAM(/bin/sh) )
DATA('Rational Developer for z Systems')
```
- ```
RDEFINE STARTED DBGMGR.* DATA('DEBUG MANAGER')
STDATA(USER(STCDBM) GROUP(STCGROUP) TRUSTED(NO))
```
- SETROPTS RACLIST(STARTED) REFRESH

**Nota:**

- Assegure-se de que os IDs de usuário das tarefas iniciadas sejam protegidos especificando-se a palavra-chave NOPASSWORD.
- A tarefa iniciada Debug Manager (DBGMGR) é usada somente pelo recurso Depurador integrado.

---

## Definir o Debug Manager como um servidor z/OS UNIX seguro

O Depurador Integrado requer acesso UPDATE ao perfil BPX.SERVER para criar ou excluir o ambiente de segurança para o encadeamento de depuração. Observe que o uso de UID(0) para efetuar bypass desse requisito não é suportado. Essa permissão é necessária apenas quando o recurso Depurador Integrado opcional é usado.

- RDEFINE FACILITY BPX.SERVER UACC(NONE)
- PERMIT BPX.SERVER CLASS(FACILITY) ACCESS(UPDATE) ID(STCDBM)
- SETROPTS RACLIST(FACILITY) REFRESH

**Atenção:** Definir o perfil BPX.SERVER torna o z/OS UNIX um comutador completo da segurança de nível UNIX para a segurança de nível z/OS UNIX, que é mais segura. Esse comutador pode afetar outros aplicativos e operações z/OS UNIX. Teste a segurança antes de ativá-la em um sistema de produção. Para obter informações adicionais sobre os diferentes níveis de segurança, consulte *UNIX System Services Planning* (GA22-7800).

---

## Definir as bibliotecas controladas pelo programa MVS para o Debug Manager

Servidores com autoridade para BPX.SERVER devem executar em um ambiente limpo e controlado por programa. Esse requisito significa que todos os programas chamados pelo Debug Manager 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 Debug Manager usa bibliotecas do sistema, tempo de execução do Language Environment e a biblioteca de carregamento do Developer for z Systems (ISP.SISPLoad).

- RALTER PROGRAM \*\* UACC(READ) ADDMEM('SYS1.LINKLIB'//NOPADCHK)
- RALTER PROGRAM \*\* UACC(READ) ADDMEM('SYS1.CSSLIB'//NOPADCHK)
- RALTER PROGRAM \*\* UACC(READ) ADDMEM('CEE.SCEERUN'//NOPADCHK)
- RALTER PROGRAM \*\* UACC(READ) ADDMEM('CEE.SCEERUN2'//NOPADCHK)
- RALTER PROGRAM \*\* UACC(READ) ADDMEM('FEL.SFELAUTH'//NOPADCHK)
- SETROPTS WHEN(PROGRAM) REFRESH

**Nota:** Não utilize o perfil \*\* se já tiver um perfil \* na classe PROGRAM. O perfil obscurece e complica o caminho da procura usado pelo software de segurança. Nesse caso, você deve mesclar as definições \* existentes com a \*\* nova. Use o perfil \*\*, conforme documentado em *Security Server RACF Security Administrator's Guide* (SA22-7683).

As seguintes bibliotecas de pré-requisitos adicionais devem se tornar controladas pelo programa para suportarem o uso de serviços opcionais. Esta lista não inclui conjuntos de dados que são específicos para um produto com o qual o Developer for z Systems interage, como o IBM Explorer for z/OS.

- Biblioteca de tempo de execução alternativa REXX, para SCLM Developer Toolkit
  - REXX.\*.SEAGALT

**Nota:** As bibliotecas que são projetadas para colocação de LPA também requerem autorizações de controle de programa se forem acessadas por meio de LINKLIST ou STEPLIB. Esta publicação documenta o uso das seguintes bibliotecas LPA:

- Biblioteca de tempo de execução REXX, para SCLM Developer Toolkit
  - REXX.\*.SEAGLPA
- Developer for z Systems, para CARMA
  - FEL.SFELLPA

---

## Definir acesso ao Integrated Debugger

Os usuários requerem acesso READ a um dos perfis AQE.AUTHDEBUG.\* listados para poderem usar o Integrated Debugger para depurar programas de estado de problema. Os usuários com permissão para o perfil AQE.AUTHDEBUG.AUTHPGM também têm permissão para depurar programas autorizados por APF. Substitua o item temporário #apf por IDs válidos de usuário ou nomes de grupos RACF para esses usuários com permissão de depurar programas autorizados.

- RDEFINE FACILITY AQE.AUTHDEBUG.STDPGM UACC(NONE)
- PERMIT AQE.AUTHDEBUG.STDPGM CLASS(FACILITY) ACCESS(READ) ID(\*)
- RDEFINE FACILITY AQE.AUTHDEBUG.AUTHPGM UACC(NONE)
- PERMIT AQE.AUTHDEBUG.AUTHPGM CLASS(FACILITY) ACCESS(READ) ID(#apf)
- SETROPTS RACLIST(FACILITY) REFRESH

**Nota:** As versões do IBM Rational Developer for System z anteriores à versão 9.1.1 usavam outro perfil de classe FACILITY, AQE.AUTHDEBUG.WRITEBUFFER, que não está mais em uso. Ela poderá ser removida se o seu sistema host tiver apenas o IBM Rational Developer for System z versão 9.1.1 ou superior.

---

## 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 for z Systems. Substitua o marcador #sysprog por IDs de usuário válidos ou nomes de grupos do RACF. Além disso, solicite do programador de sistema que instalou e configurou o produto os nomes dos conjuntos de dados corretos. FEK é o qualificador de alto nível padrão usado durante a instalação e FEL.#CUST é o qualificador de alto nível padrão para conjuntos de dados criados durante o processo de customização.

- ```
ADDGROUP (FEL) OWNER(IBMUSER) SUPGROUP(SYS1)
DATA('IBM Rational Developer for z Systems - HLQ STUB')
```
- ```
ADDSD 'FEL.*.***' UACC(READ)
DATA('IBM Rational Developer for z Systems')
```
- ```
PERMIT 'FEL.*.***' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
```
- ```
SETROPTS GENERIC(DATASET) REFRESH
```

**Nota:**

- Proteja FEL.SFELAUTH contra atualizações, pois esse conjunto de dados é autorizado pelo APF.
- Os comandos de amostra nesta publicação e na tarefa FELRACF supõem que a Enhanced Generic Naming (EGN) esteja ativa. Quando o EGN está ativo, o qualificador \*\* pode ser usado para representar qualquer número de qualificadores na classe DATASET. Substitua \*\* por \* se a EGN não estiver ativa em seu sistema. Para obter informações adicionais sobre o EGN, consulte *Security Server RACF Security Administrator's Guide (SA22-7683)*.

Alguns dos componentes do Developer for z Systems requerem perfis adicionais do conjunto de dados de segurança. Substitua os itens temporários #sysprog e #ram-developer por IDs de usuário válidos ou nomes do grupo RACF:

- Se a tradução de nome longo/curto do SCLM Developer Toolkit for usada, os usuários precisarão de acesso UPDATE para o VSAM do mapeamento, FEL.#CUST.LSTRANS.FILE.  

```

--
ADDSD 'FEL.#CUST.LSTRANS.**' UACC(UPDATE)
DATA('IBM Rational Developer for z Systems - SCLMDT')
--
PERMIT 'FEL.#CUST.LSTRANS.**' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
--
SETROPTS GENERIC(DATASET) REFRESH

```
- Os desenvolvedores do RAM (Gerenciador de acesso do repositório) do CARMA precisam de acesso UPDATE para os VSAMs do CARMA, FEL.#CUST.CRA\*.  

```

--
ADDSD 'FEL.#CUST.CRA**' UACC(READ)
DATA('IBM Rational Developer for z Systems - CARMA')
--
PERMIT 'FEL.#CUST.CRA**' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
--
PERMIT 'FEL.#CUST.CRA**' CLASS(DATASET) ACCESS(UPDATE) ID(#ram-developer)
--
SETROPTS GENERIC(DATASET) REFRESH

```

---

## Verificar as 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
- Tarefas iniciadas
  - LISTGRP STCGROUP OMVS
  - LISTUSER STCDBM OMVS
  - RLIST STARTED DBGMR.\* ALL STDATA
- Debug Manager como um servidor z/OS UNIX seguro
  - RLIST FACILITY BPX.SERVER ALL
- Bibliotecas controladas pelo programa MVS para Debug Manager
  - RLIST PROGRAM \*\* ALL
- Acesso do Integrated Debugger
  - RLIST FACILITY AQE.\*\* ALL

- |
- |
- |
- Perfis do conjunto de dados
  - LISTGRP FEL
  - LISTDSD PREFIX(FEL) ALL

---

## Capítulo 10. Guia de Migração

---

### Considerações sobre 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. Para obter informações adicionais, consulte as seções relacionadas neste manual.

- Se você for um usuário anterior do IBM Rational Developer for z Systems, salve os arquivos customizados relacionados antes de fazer o upgrade para esta versão do IBM Rational Developer for z Systems .
- Se você planeja executar várias instâncias do Developer for z Systems, leia "Executando diversas instâncias" no *Referência de Configuração do Host* (SC27-8578).
- Se seu cenário de migração abranger mais de duas liberações, você deverá fazer as customizações novamente, como se não houvesse nenhuma liberação mais antiga presente.

### Fazendo Backup de Arquivos Configurados Anteriormente

Se você for um usuário anterior do Developer for z Systems, salve os arquivos customizados relacionados antes de instalar esta versão do IBM Developer for z Systems.

Os arquivos customizáveis do Developer for z Systems podem ser localizados nos seguintes locais:

- Todas as versões anteriores
  - FEK.SFEKSAMP, alguns membros são copiados em FEK.#CUST.\* pela tarefa de amostra FEKSETUP, em que \* é igual a PARMLIB, PROCLIB, JCL, CNTL, ASM e COBOL
  - FEK.SFEKSAMV
  - /usr/lpp/rdz/samples/, alguns arquivos são copiados em /etc/rdz/ e /var/rdz/sclmdt/\* pela tarefa de amostra FEKSETUP, em que \* é igual a CONFIG/, CONFIG/PROJECT/ e CONFIG/script/

As configurações anteriores do Developer for z Systems também documentam mudanças nos arquivos de configuração pertencentes a outros produtos.

- Versões anteriores até a versão 9.0.1
    - SYS1.PARMLIB(IEASCVxx)
  - Todas as versões anteriores
    - SYS1.PARMLIB(BPXPRMxx)
    - SYS1.PARMLIB(COMMNDxx)
    - SYS1.PARMLIB(LPALSTxx)
    - SYS1.PARMLIB(PROGxx)
    - (WLM)
- Designa um ambiente de aplicativo para um procedimento armazenado do DB2.

---

## Migrar da versão 9.5 para a versão 9.5.1

Na versão 9.5.1, as funções relacionadas ao Monitor de tarefas RSE e foram movidas de IBM Rational Developer for z Systems para outro produto, o IBM Explorer for z/OS. O IBM Explorer for z/OS é agora um pré-requisito do IBM Rational Developer for z Systems.

A IBM recomenda fortemente que você refaça todas as customizações usando o novo z/OS Explorer e amostras do Developer for z Systems para garantir que esteja usando os novos nomes do conjunto de dados MVS e caminhos do z/OS UNIX.

Conjuntos de dados SMP/E para z/OS:

- z/OS Explorer é um pré-requisito do SMP/E para z/OS para Developer for z Systems. Isso significa que o z/OS Explorer e o Developer for z Systems devem ser instalados no mesmo CSI.
- Não há mudanças relacionadas ao SMP/E para z/OS para Developer for z Systems Host Utilities.

Conjuntos de dados MVS:

- O z/OS Explorer usa o qualificador de alto nível FEK e os nomes do conjunto de dados SFEK\*. Nenhuma ação de migração é necessária aqui.
- O Developer for z Systems usa um novo qualificador de alto nível, FEL, e novos nomes do conjunto de dados SFEL\*. Verifique suas referências atuais para FEK e atualize-as para usar FEL no lugar, onde for necessário.
  - Definições de segurança nas classes DATASET e PROGRAM
  - PROCLIB para tarefas iniciadas, procedimentos de construção remota e zUnit
  - Configuração para iniciar o CARMA
  - JCL para processar VSAMs do CARMA
  - JCL para processar saída de usuário de emulação de LAN
- O Developer for z Systems Host Utilities usa o qualificador de alto nível AKG e nomes do conjunto de dados SAKG\*. Nenhuma ação de migração é necessária.

Caminhos do z/OS UNIX:

- O IBM Explorer for z/OS usa novos caminhos z/OS UNIX, `/usr/lpp/ibm/zexpl`, `/etc/zexpl` e `/var/zexpl`. Verifique suas referências atuais para os caminhos antigos (`/usr/lpp/rdz`, `/etc/rdz` e `/var/rdz`) e atualize-os para usar os novos, onde for necessário.
  - PROCLIB para tarefas iniciadas
  - Referências JCL para `/etc/rdz`
  - Observe que `/etc/rdz/rsed.envvars` foi substituído por `/etc/zexpl/rse.env`
  - Observe que os logs, direcionar ao cliente e ISPF WORKAREA, foram movidos para `/var/zexpl`
- O Developer for z Systems usa os novos caminhos z/OS UNIX, `/usr/lpp/ibm/rdz` e `/etc/zexpl`. Ele ainda usa o caminho `/var/rdz`. Verifique suas referências atuais para os caminhos antigos (`/usr/lpp/rdz`, `/etc/rdz`) e atualize-os para usar os novos, onde for necessário.
  - JCL para utilizar conversores de saída zUnit de amostra
  - Observe que `/etc/rdz/rsed.envvars` foi substituído por `/etc/zexpl/rdz.env`
  - Observe que SCLMDT permanece em `/var/rdz/sc1mdt`



- O Developer for z Systems Host Utilities usa o caminho `/usr/lpp/rdzutil`. Nenhuma ação de migração é necessária.

## IBM Rational Developer for z Systems, FMID HHOP951

- O local de instalação padrão do SMP/E para z/OS para componentes MVS e z/OS UNIX mudaram de FEK.SFEK\* e de `/usr/lpp/rdz/*` para FEL.SFEL\* e `/usr/lpp/ibm/rdz/*`. O qualificador de alto nível FEK é agora usado pelo IBM Explorer for z/OS.
- O local de customização padrão para componentes MVS e z/OS UNIX mudaram de `/etc/rdz` para `/etc/zexpl`.
- Funções que foram movidas para o IBM Explorer for z/OS
  - O Host Configuration Utility foi movido para IBM Explorer for z/OS
    - Atividades de configuração para Monitor de tarefas JES, RSE e Gateway ISPF do Legacy permanecem
    - Atividades de configuração para CARMA, Depurador integrado e SCLMDT são removidas
  - Monitor de tarefas JES movido para IBM Explorer for z/OS
    - CNTL(FEJTSO)
    - PROCLIB(JMON)
    - SAMPLIB(FEJCNFG)
  - Direcionar ao cliente movido para IBM Explorer for z/OS
    - `/etc/rdz/pushtoclient.properties` se move para `/etc/zexpl/pushtoclient.properties`
    - `/etc/zexpl/pushtoclient.properties` muda `/var/rdz -> /var/zexpl`
  - RSE movido para IBM Explorer for z/OS
    - PROCLIB(RSED)
    - `/etc/rdz/rsed.envvars` move para `/etc/zexpl/rse.env` (renomeado)
    - `/etc/rdz/ISPF.conf` move para `/etc/zexpl/ISPF.conf`
    - `/etc/rdz/rsecomm.properties` move para `/etc/zexpl/rsecomm.properties`
    - `/etc/rdz/ssl.properties` move para `/etc/zexpl/ssl.properties`
    - `/etc/zexpl/rse.env` contém somente definições de customização para funções do z/OS Explorer
  - Ferramentas do sistema movidas para o IBM Explorer for z/OS
    - JCL(FEKRACF) contém somente definições de segurança para funções do z/OS Explorer
    - JCL(FEKSETUP) contém somente definições de customização para funções do z/OS Explorer
    - JCL(FEKSETUP) muda `usr/lpp/rdz -> /usr/lpp/ibm/zexpl`, `/etc/rdz -> /etc/zexpl`, `/var/rdz -> /var/zexpl`
- Funções que permanecem com IBM Rational Developer for z Systems
  - Assistente de conteúdo C/C++
    - `/etc/rdz/include.conf` move para `/etc/zexpl/include.conf`
  - CARMA
    - CNTL(CRASUBCA) muda FEK.\* -> FEL.\*, SFEK\* -> SFEL\*
    - CNTL(CRASUBMT) muda FEK.\* -> FEL.\*, SFEK\* -> SFEL\*
    - JCL(CRA\$VCAD) muda FEK.\* -> FEL.\*, SFEK\* -> SFEL\*
    - JCL(CRA\$VCAS) muda FEK.\* -> FEL.\*, SFEK\* -> SFEL\*

- JCL(CRA\$VDEF) muda FEK.\* -> FEL.\*, SFEK\* -> SFEL\*
- JCL(CRA\$VMSG) muda FEK.\* -> FEL.\*, SFEK\* -> SFEL\*
- JCL(CRA\$VSTR) muda FEK.\* -> FEL.\*, SFEK\* -> SFEL\*
- JCL(CRA#CIRX) muda FEK.\* -> FEL.\*
- JCL(CRA#UADD) muda FEK.\* -> FEL.\*
- JCL(CRA#UQRY) muda FEK.\* -> FEL.\*
- JCL(CRA#VPDS) muda FEK.\* -> FEL.\*
- JCL(CRA#VSLM) muda FEK.\* -> FEL.\*
- /etc/rdz/CRASRV.properties move para /etc/zexpl/CRASRV.properties
- /etc/rdz/crastart.conf move para /etc/zexpl/crastart.conf
- /etc/rdz/crastart.endevor.conf move para /etc/zexpl/crastart.endevor.conf
- /etc/zexpl/crastart.conf muda FEK.\* -> FEL.\*, SFEK\* -> SFEL\*
- /etc/zexpl/crastart.endevor.conf muda FEK.\* -> FEL.\*, SFEK\* -> SFEL\*
- Depurador Integrado
  - JCL(AQED3CEE) muda FEK.\* -> FEL.\*  
 isso atualiza o local padrão onde a saída de usuário LE é armazenada, de modo que os usuários podem precisar atualizar seu uso da saída de usuário LE
  - JCL(AQED3CXT) muda FEK.\* -> FEL.\*
  - JCL(AQERACF) muda FEK.SFEK\* -> FEL.SFEL\*
  - PROCLIB(DBGMGR) muda FEK.SFEK\* -> FEL.SFEL\*
- Procedimentos de construção remota
  - PROCLIB(ELAXFADT) muda FEK -> FEL (LODPRFX e STEPLIB)
  - PROCLIB(ELAXFASM) muda FEK -> FEL (LODPRFX e STEPLIB)
  - PROCLIB(ELAXFCOC) muda FEK -> FEL (LODPRFX e STEPLIB)
  - PROCLIB(ELAXFCOP) muda FEK -> FEL (LODPRFX e STEPLIB)
  - PROCLIB(ELAXFCOT) muda FEK -> FEL (LODPRFX e STEPLIB)
  - PROCLIB(ELAXFCP1) muda FEK -> FEL (LODPRFX e STEPLIB)
  - PROCLIB(ELAXFDCL) muda FEK -> FEL (LODPRFX e STEPLIB)
  - PROCLIB(ELAXFGO) muda FEK -> FEL (LODPRFX e STEPLIB)
  - PROCLIB(ELAXFLNK) muda FEK -> FEL (LODPRFX e STEPLIB)
  - PROCLIB(ELAXFPLP) muda FEK -> FEL (LODPRFX e STEPLIB)
  - PROCLIB(ELAXFPLT) muda FEK -> FEL (LODPRFX e STEPLIB)
  - PROCLIB(ELAXFPL1) muda FEK -> FEL (LODPRFX e STEPLIB)
  - PROCLIB(ELAXFPP1) muda FEK -> FEL (LODPRFX e STEPLIB)
  - PROCLIB(ELAXFTSO) muda FEK -> FEL (LODPRFX e STEPLIB)
  - PROCLIB(ELAXFUOP) muda FEK -> FEL (LODPRFX e STEPLIB)
  - PROCLIB(ELAXFSP) muda FEK.\* -> FEL.\*
  - PROCLIB(ELAXFSQL) muda FEK.\* -> FEL.\*
  - SQL(FEKTEP2) movido para SQL(FELTEP2)
  - SQL(FEKTIAD) movido para SQL(FELTIAD)
- Ferramentas do sistema
  - (nova) JCL(FELRACF) contém definições de segurança para funções do Developer for z Systems

- (nova) JCL(FELSETUP) contém definições de customização para funções do Developer for z Systems
  - (nova) /etc/zexpl/rdz.env contém definições de customização para funções do Developer for z Systems localizadas anteriormente em /etc/rdz/rsed.envvars
  - zUnit
  - PROCLIB(AZUZUNIT) muda FEK -> FEL (LODPRFX & STEPLIB)
- os usuários podem precisar atualizar sua chamada de procedimento para usar o novo caminho do Developer for z Systems, /usr/lpp/ibm/rdz/samples/zunit/\*

## Arquivos configuráveis

Tabela 17 mostra uma visão geral de arquivos do Developer for z Systems que são customizados na versão 9.5.1. As bibliotecas de amostra do Developer for z Systems, FEL.SFELSAMP, FEL.SFELSAMPV e /usr/lpp/ibm/rdz/samples/, contêm mais membros customizáveis que os listados aqui, tais como código-fonte do CARMA de amostra e tarefas para compilá-los.

Nota: a tarefa de amostra FELSETUP copia todos os membros listados em diferentes conjuntos de dados e diretórios, FEL.#CUST.\* e /etc/zexpl/\* padrão.

*Tabela 17. Customizações da versão 9.5.1*

| Membro/Arquivo | Local padrão                               | Propósito                                                                                | Notas sobre Migração                             |
|----------------|--------------------------------------------|------------------------------------------------------------------------------------------|--------------------------------------------------|
| FELSETUP       | FEL.SFELSAMP<br>[FEL.#CUST.JCL]            | JCL para criar conjuntos de dados e diretórios e preenchê-los com arquivos customizáveis | Novo, configuração é necessária                  |
| DBGMR          | FEL.SFELSAMP<br>[FEL.#CUST.JCL]            | JCL para o Debug Manager                                                                 | Usando novos nomes de conjunto de dados/caminhos |
| AQEJCL         | FEL.SFELSAMP<br>[FEL.#CUST.PROCLIB(DBGMR)] | Nome para membro DBGMR                                                                   | Consulte o membro DBGMR                          |
| ELAXF*         | FEL.SFELSAMP<br>[FEL.#CUST.PROCLIB]        | JCL para construções de projetos remotos, etc.                                           | Usando novos nomes de conjunto de dados/caminhos |
| FELRACF        | FEL.SFELSAMP<br>[FEL.#CUST.JCL]            | JCL para definições de segurança                                                         | Novo, configuração é necessária                  |
| AQERACF        | FEL.SFELSAMP<br>[FEL.#CUST.JCL]            | JCL for Debug Manager                                                                    | Usando novos nomes de conjunto de dados/caminhos |
| CRA\$VMSG      | FEL.SFELSAMP<br>[FEL.#CUST.JCL]            | JCL para criação do VSAM de mensagens do CARMA                                           | Usando novos nomes de conjunto de dados/caminhos |

Tabela 17. Customizações da versão 9.5.1 (continuação)

| Membro/Arquivo | Local padrão                        | Propósito                                                                         | Notas sobre Migração                             |
|----------------|-------------------------------------|-----------------------------------------------------------------------------------|--------------------------------------------------|
| CRA\$VDEF      | FEL.SFELSAMP<br>[FEL.#CUST.JCL]     | JCL para criação do VSAM de configuração do CARMA                                 | Usando novos nomes de conjunto de dados/caminhos |
| CRA\$VSTR      | FEL.SFELSAMP<br>[FEL.#CUST.JCL]     | JCL para criação do VSAM de informações customizadas do CARMA                     | Usando novos nomes de conjunto de dados/caminhos |
| CRA\$VCAD      | FEL.SFELSAMP<br>[FEL.#CUST.JCL]     | JCL para criar VSAM da configuração de CARMA para CA Endevor® SCM RAM             | Usando novos nomes de conjunto de dados/caminhos |
| CRA\$VCAS      | FEL.SFELSAMP<br>[FEL.#CUST.JCL]     | JCL para criar VSAM de informações customizadas do CARMA para CA Endevor® SCM RAM | Usando novos nomes de conjunto de dados/caminhos |
| CRASUBMT       | FEL.SFELSAMP<br>[FEL.#CUST.CNTL]    | CLIST de inicialização em lote do CARMA                                           | Usando novos nomes de conjunto de dados/caminhos |
| CRASUBCA       | FEL.SFELSAMP<br>[FEL.#CUST.CNTL]    | CLIST de inicialização em lote do CARMA para CA Endevor® SCM RAM                  | Usando novos nomes de conjunto de dados/caminhos |
| CRACFG         | FEL.SFELSAMP<br>[FEL.#CUST.PARMLIB] | Configuração de interação CARMA para o CA Endevor® SCM RAM                        | Nenhum                                           |
| CRABCFG        | FEL.SFELSAMP<br>[FEL.#CUST.PARMLIB] | Configuração de ações em lote do CARMA para CA Endevor® SCM RAM                   | Nenhum                                           |
| CRABATCA       | FEL.SFELSAMP<br>[FEL.#CUST.CNTL]    | JCL de ação em lote do CARMA para CA Endevor® SCM RAM                             | Nenhum                                           |
| CRASCL         | FEL.SFELSAMP<br>[FEL.#CUST.PARMLIB] | SCL do Modelo para CA Endevor® SCM                                                | Nenhum                                           |

Tabela 17. Customizações da versão 9.5.1 (continuação)

| Membro/Arquivo | Local padrão                        | Propósito                                               | Notas sobre Migração                             |
|----------------|-------------------------------------|---------------------------------------------------------|--------------------------------------------------|
| CRASHOW        | FEL.SFELSAMP<br>[FEL.#CUST.PARMLIB] | Configuração do CARMA para CA Endevor® SCM RAM          | Nenhum                                           |
| CRATMAP        | FEL.SFELSAMP<br>[FEL.#CUST.PARMLIB] | Configuração do CARMA para CA Endevor® SCM RAM          | Nenhum                                           |
| CRANDVRA       | FEL.SFELPROC                        | REXX de alocação do CARMA para CA Endevor® SCM RAM      | Nenhum                                           |
| CRADYND        | FEL.SFELPROC                        | REXX de alocação do CARMA para CA Endevor® SCM RAM      | Nenhum                                           |
| CRAALLOC       | FEL.SFELPROC                        | REXX de alocação CARMA                                  | Nenhum                                           |
| CRA#VSLM       | FEL.SFELSAMP<br>[FEL.#CUST.JCL]     | JCL para criar VSAM de mensagem do SCLM RAM             | Usando novos nomes de conjunto de dados/caminhos |
| CRA#ASLM       | FEL.SFELSAMP<br>[FEL.#CUST.JCL]     | JCL para criar conjuntos de dados do SCLM RAM           | Usando novos nomes de conjunto de dados/caminhos |
| CRA#VPDS       | FEL.SFELSAMP<br>[FEL.#CUST.JCL]     | JCL para criar VSAM de mensagem do PDS RAM              | Usando novos nomes de conjunto de dados/caminhos |
| CRA#UADD       | FEL.SFELSAMP<br>[FEL.#CUST.JCL]     | JCL para mesclagem de definições da RAM                 | Usando novos nomes de conjunto de dados/caminhos |
| CRA#UQRY       | FEL.SFELSAMP<br>[FEL.#CUST.JCL]     | JCL para extração de definições da RAM                  | Usando novos nomes de conjunto de dados/caminhos |
| CRAXJCL        | FEL.SFELSAMP<br>[FEL.#CUST.ASM]     | Código de origem de amostra para substituição de IRXJCL | Nenhum                                           |
| CRA#CIRX       | FEL.SFELSAMP<br>[FEL.#CUST.JCL]     | JCL para compilar CRAXJCL                               | Usando novos nomes de conjunto de dados/caminhos |

Tabela 17. Customizações da versão 9.5.1 (continuação)

| Membro/Arquivo    | Local padrão                               | Propósito                                                                   | Notas sobre Migração                                                                                  |
|-------------------|--------------------------------------------|-----------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------|
| AQECSD            | FEL.SFELSAMP<br>[FEL.#CUST.JCL]            | JCL para definir o Integrated Debugger para regiões do CICS                 | Nenhum                                                                                                |
| AQED3CEE          | FEL.SFELSAMP<br>[FEL.#CUST.JCL]            | JCL para criar módulos de tempo de execução LE customizados                 | Usando novos nomes de conjunto de dados/caminhos                                                      |
| AQED3CXT          | FEL.SFELSAMP<br>[FEL.#CUST.JCL]            | JCL para criar a saída de usuário LE                                        | Usando novos nomes de conjunto de dados/caminhos                                                      |
| FEKTEP2           | FEL.SFELSAMP<br>[FEL.#CUST.SQL]            | Entrada do comando SQL usada pelo ELAXF*                                    | Nenhum                                                                                                |
| FEKTIAD           | FEL.SFELSAMP<br>[FEL.#CUST.JCL]            | Entrada do comando SQL usada pelo ELAXF*                                    | Nenhum                                                                                                |
| AZUZUNIT          | FEL.SFELSAMP<br>[FEL.#CUST.PROCLIB]        | Procedimento JCL para zUnit                                                 | Usando novos nomes de conjunto de dados/caminhos                                                      |
| FEKRNPLI          | FEL.SFELSAMP<br>[FEL.#CUST.CNTL]           | REXX para chamar o compilador PL/I a partir da estrutura de pré-processador | Nenhum                                                                                                |
| IRZCSD            | FEL.SFELSAMP<br>[FEL.#CUST.JCL]            | JCL para definir o Enterprise Service Tools para CICS                       | Nenhum                                                                                                |
| rdz.env           | /usr/lpp/ibm/rdz/samples/<br>[/etc/zexpl/] | Variáveis de ambiente do RSE                                                | As cópias mais antigas devem ser substituídas por esta e as customizações devem ser feitas novamente. |
| CRASRV.properties | /usr/lpp/ibm/rdz/samples/<br>[/etc/zexpl/] | Arquivo de configuração CARMA                                               | Nenhum                                                                                                |
| crastart.conf     | /usr/lpp/ibm/rdz/samples/<br>[/etc/zexpl/] | Arquivo de configuração do CARMA para uso de CRASTART                       | Usando novos nomes de conjunto de dados/caminhos                                                      |

Tabela 17. Customizações da versão 9.5.1 (continuação)

| Membro/Arquivo        | Local padrão                               | Propósito                                                                                  | Notas sobre Migração                             |
|-----------------------|--------------------------------------------|--------------------------------------------------------------------------------------------|--------------------------------------------------|
| crastart.endevor.conf | /usr/lpp/ibm/rdz/samples/<br>[/etc/zexpl/] | Arquivo de configuração do CARMA para uso do CRASTART para CA Endevor <sup>®</sup> SCM RAM | Usando novos nomes de conjunto de dados/caminhos |
| include.conf          | /usr/lpp/ibm/rdz/samples/<br>[/etc/zexpl/] | Inclusões forçadas para assistente de conteúdo C/C++                                       | Nenhum                                           |

Tabela 18 mostra uma visão geral de arquivos customizáveis que são movidos para o z/OS Explorer na versão 9.5.1.

Nota: a tarefa de amostra FEKSETUP copia todos os membros listados para diferentes conjuntos de dados e diretórios, FEK.#CUST.\* e /etc/zexpl/\* padrão.

Tabela 18. Customizações da versão 9.5.1 z/OS Explorer

| Membro/Arquivo | Local padrão                                  | Propósito                                                                                | Notas sobre Migração                                                                                  |
|----------------|-----------------------------------------------|------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------|
| FEKSETUP       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]               | JCL para criar conjuntos de dados e diretórios e preenchê-los com arquivos customizáveis | Atualizado para remover ações de arquivos que não são mais usados e incluir ações nos novos arquivos. |
| JMON           | FEK.SFEKSAMP (FEJJJCL)<br>[FEK.#CUST.PROCLIB] | JCL para JES Job Monitor                                                                 | Nenhum                                                                                                |
| FEJJJCL        | FEK.SFEKSAMP<br>[FEK.#CUST.PROCLIB(JMON)]     | Nome do membro JMON                                                                      | Consulte o membro de JMON                                                                             |
| RSED           | FEK.SFEKSAMP (FEKRSED)<br>[FEK.#CUST.PROCLIB] | JCL para daemon RSE                                                                      | Nenhum                                                                                                |
| FEKRSED        | FEK.SFEKSAMP<br>[FEK.#CUST.PROCLIB(RSED)]     | Nome do membro RSED                                                                      | Consulte o membro de RSED                                                                             |
| FEKRACF        | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]               | JCL para definições de segurança                                                         | Dados específicos do Developer for z Systems removidos                                                |
| FEJJCNFG       | FEK.SFEKSAMP<br>[FEK.#CUST.PARMLIB]           | Arquivo de configuração do JES Job Monitor                                               | Nenhum                                                                                                |
| FEJTSO         | FEK.SFEKSAMP<br>[FEK.#CUST.CNTL]              | JCL para emissões TSO                                                                    | Nenhum                                                                                                |

Tabela 18. Customizações da versão 9.5.1 z/OS Explorer (continuação)

| Membro/Arquivo          | Local padrão                                 | Propósito                                                    | Notas sobre Migração                                                                                           |
|-------------------------|----------------------------------------------|--------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------|
| FEKLOGS                 | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]              | JCL para coletar arquivos de log                             | Usando novos nomes de conjunto de dados/caminhos<br><br>Dados específicos do Developer for z Systems removidos |
| rse.env                 | /usr/lpp/ibm/zexpl/samples/<br>[/etc/zexpl/] | Variáveis de ambiente do RSE                                 | Cópias mais antigas de rsed.envvars devem ser substituídas por esta e as customizações feitas novamente.       |
| ISPF.conf               | /usr/lpp/ibm/zexpl/samples/<br>[/etc/zexpl/] | Arquivo de configuração do Gateway ISPF                      | Usando novos nomes de conjunto de dados/caminhos                                                               |
| ssl.properties          | /usr/lpp/ibm/zexpl/samples/<br>[/etc/zexpl/] | Arquivo de configuração de comunicação criptografada do RSE  | Nenhum                                                                                                         |
| rsecomm.properties      | /usr/lpp/ibm/zexpl/samples/<br>[/etc/zexpl/] | Arquivo de configuração de rastreamento RSE                  | Nenhum                                                                                                         |
| pushtoclient.properties | /usr/lpp/ibm/zexpl/samples/<br>[/etc/zexpl/] | Enviar informações para o arquivo de configuração do cliente | Usando novos nomes de conjunto de dados/caminhos                                                               |

## IBM Rational Developer for z Systems Host Utilities, FMID HAKG951

- O local de instalação padrão do SMP/E para os componentes MVS e z/OS UNIX não foi alterado e, portanto, permanece AKG.\* e /usr/lpp/rdzutil/\*.

### Arquivos Configuráveis

Tabela 19 na página 119 mostra uma visão geral de arquivos que são customizados na versão 9.5.1. As bibliotecas de amostra do Developer for z Systems Host Utilities, AKG.SAKGSAMP e /usr/lpp/rdzutil/samples/, contêm mais membros customizáveis do que os listados aqui, como o script de pós-processamento de revisão de código de amostra.

**Nota:** A tarefa de amostra AKGSETUP copia todos os membros listados em conjuntos de dados diferentes, padrão AKG.#CUST.\*.



Tabela 19. Customizações do Host Utilities Versão 9.0

| Membro ou Arquivo | Local-padrão                        | Finalidade                                                                  | Notas sobre Migração |
|-------------------|-------------------------------------|-----------------------------------------------------------------------------|----------------------|
| AKGSETUP          | AKG.SAKGSAMP<br>[AKG.#CUST.JCL]     | JCL para criar conjuntos de dados e preenchê-los com arquivos customizáveis | Nenhum               |
| AKGCC             | AKG.SAKGSAMP<br>[AKG.#CUST.PROCLIB] | JCL para cobertura de código                                                | Nenhum               |
| AKGCR             | AKG.SAKGSAMP<br>[AKG.#CUST.PROCLIB] | JCL para revisão de código                                                  | Nenhum               |
| AKGCRADD          | AKG.SAKGSAMP<br>[AKG.#CUST.JCL]     | JCL para incluir código de terceiro para revisão de código                  | Nenhum               |

## Migre da versão 9.1 para a versão 9.5

Estas notas são para a migração de uma versão base 9.1 para a versão 9.5. Ela inclui mudanças já documentadas como parte da manutenção da versão 9.1. As mudanças que fazem parte do fluxo de manutenção e, portanto, já estão possivelmente implementadas, estão marcadas com a liberação onde elas foram introduzidas.

É recomendado substituir um `rsed.envvars` existente (por padrão em `/etc/rdz`) pela amostra fornecida recentemente (por padrão em `/usr/lpp/rdz/samples`) e refazer as customizações.

## IBM Rational Developer for z Systems, FMID HHOP950

- O local de instalação padrão do SMP/E para os componentes MVS e z/OS UNIX não foi alterado e permanece FEK.\* e `/usr/lpp/rdz/*`.
- A partir da versão 9.5
  - O Application Deployment Manager foi removido do produto. As seguintes definições foram removidas:
    - SFEKLOAD(ADN\*)
    - FEK.#CUST.JCL(ADN\*)
    - FEK.#CUST.COBOLE(ADN\*)
    - FEK.#CUST.ADNREPF0 [VSAM criado por ADNVCRD]
    - FEK.#CUST.ADNMANF0 [VSAM criado por ADNVMFST]
    - FEK.#CUST.ADN\* perfis de segurança do conjunto de dados [executado por FEKRACF]
    - Definições de CICS CSD [executadas por ADNCSDAR, ADNCSDRS, ADNCSDTX, ADNCSDWS, ADNVMFST, AQECSD]
    - `usr/lpp/rdz/cicsts`
  - CARMA: Os membros customizáveis a seguir foram alterados:
    - CRACFG define nova opção, DYNAMIC-VB-DATASET-ALLOC
    - CRACFG define nova opção, DATASET-ALLOC-OVERRIDE
  - Enterprise Service Tools: novos membros customizáveis

- IRZCSD: atualização CSD da região do CICS
- Integrated Debugger: A depuração de transações CICS agora requer IP CICS Sockets.
- PL/I: novos membros customizáveis
  - FEKRNPLI: REXX pré-processador de PL/I
- Procedimentos de construção remotos: Os seguintes membros customizáveis foram alterados:
  - ELAXFDCL: incluir SFEKAUTH no STEPLIB para Integrated Debugger
  - ELAXFTSO: incluir SFEKAUTH no STEPLIB para Integrated Debugger
- RSE: Novos comandos do operador foram incluídos:
  - F rsed,APPL=SEND 'message'[USER={ userid | (userid,userid,...)}]
  - F rsed,APPL=SEND 'message'[PID={ pid | (pid,pid,...)}]
  - F rsed,APPL=SEND CLEARALL
- RSE: Diretivas opcionais foram incluídas em rsed.envvars:
  - CGI\_CEATSO
  - CGI\_CEATSO\_KEEPLIVE
  - GSK\_PROTOCOL\_SSLV3
  - GSK\_V3\_CIPHER\_SPECS
- RSE: Diretivas opcionais foram removidas de rsed.envvars:
  - DSTORE\_SSL\_ALGORITHM
- RSE: Diretivas necessárias foram marcadas como não editáveis no rsed.envvars:
  - \_RSE\_HOST\_CODEPAGE
- RSE: Suporte incluído para envio de mensagens para clientes usando um comando do operador, comando do z/OS UNIX ou TSO SEND
- RSE: Suporte incluído para Gateway ISPF Interativo
- zUnit: Os seguintes membros customizáveis foram alterados:
  - AZUZUNIT: incluído suporte de Cobertura de Código
- Desde a versão 9.1.1.1
  - Feedback de Erro: FEK.SFEKPROC(FEKFERRF) movido para /usr/lpp/rdz/bin/fekferrf.rex
  - CARMA: Novos membros customizáveis foram incluídos:
    - CRASCL: SCL de modelo do CARMA de amostra para CA Endevor® SCM.
  - CARMA: Os membros customizáveis a seguir foram alterados:
    - CRACFG define nova opção, PACKAGE-EDITING-OPTION
    - CRACFG define nova opção, SCL-REQUIRED
    - CRACFG define nova opção, SCL-DATASET-TEMPLATE
  - Integrated Debugger: Os membros customizáveis a seguir foram alterados:
    - AQED3CXT novo nível de depuração para saída de usuário de LE
    - AQED3CEE assegure-se de que a saída de usuário correta seja incluída no tempo de execução de LE
  - RSE: módulo de LPA opcional tornando-se necessário
    - FEKDSI deve agora estar disponível através de LPA/LINKLIST/STEPLIB
  - RSE: Novos comandos do operador foram incluídos:
    - F rsed,APPL=DEBUG {PROCESS | P},{STORAGE | STOR}[PID=pid]
    - F rsed,APPL=DEBUG {DAEMON | D},{STORAGE | STOR}

- RSE: registro do produto necessário durante a compra, como parte do código do produto 5697-CDT, IBM Enterprise COBOL Suite for z/OS
- SYS1.PARMLIB: Atualizando um novo membro PARMLIB:
  - IFAPRDxx: registro do produto
- Desde a versão 9.1.1.0
  - CARMA: Os arquivos CRADEF e CRASTRS VSAM do CA Endevor® SCM RAM foram atualizados.
  - CARMA: Os membros customizáveis a seguir foram alterados:
    - CRACFG define nova opção, ALTERNATIVE-ALLOC
    - CRABCFG define uma nova ação, TRANSFER-ELEMENT
    - CRASUBCA incluiu APIHJC
    - crastart.endevor.conf incluiu APIHJC
  - Integrated Debugger: Suporte de coexistência para o IBM Debug Tool for z/OS
  - Integrated Debugger: Os membros customizáveis a seguir foram alterados:
    - AQECSD (atualização do CICS CSD) define novos recursos
    - AQERACF define novos perfis AQE.\*\*
    - FEKRACF define novos perfis AQE.\*\*
    - Integrated Debugger: Definições obrigatórias do CICS CSD foram incluídas
      - Filas de dados transientes AQEM e AQED
      - Programa AQEW3Z3, AQEW3Z6, AQEL3Z3, AQEL3Z6, AQEL6Z3, AQEL6Z6
    - Integrated Debugger: Algumas definições do CICS CSD foram removidas
      - Filas de dados transientes CIGZ e CIBM
      - Programa AQEEV006
  - Integrated Debugger: Os membros customizáveis a seguir foram alterados:
    - AQESTC [DBGMGR] a tarefa iniciada agora tem SFEKLPa em STEPLIB
  - Integrated Debugger: SVC requerido foi incluído
    - Mudança de parmlib IEASVCxx para AQESVC03 (pode ser carregada dinamicamente)
  - Integrated Debugger: SVC opcional foi removido
    - Mudança de parmlib IEASVCxx para AQESVC01
  - Integrated Debugger: Perfis necessários do RACF foram incluídos
    - AQE.AUTHDEBUG.STDPGM
    - AQE.AUTHDEBUG.AUTHPGM
  - Integrated Debugger: Perfis opcionais do RACF foram removidos
    - AQE.AUTHDEBUG.WRITEBUFFER
  - Coleção de logs: Clientes podem coletar logs de hosts
  - RSE: Suporte à passphrase incluído
  - RSE: Diretivas opcionais foram incluídas em rsed.envvars:
    - search.server.limit.scanned\_objects
    - search,server.limit.errcount
  - RSE: Diretivas opcionais foram removidas de rsed.envvars:
    - enable.saf.check
    - RSE\_DSICALL
    - search.server.limit.datasets

## Arquivos Configuráveis

O Tabela 20 mostra uma visão geral de arquivos que são customizados na versão 9.5. As bibliotecas de amostra do Developer for z Systems, FEK.SFEKSAMP, FEK.SFEKSAMPV e /usr/lpp/rdz/samples/, contêm mais membros customizáveis do que os listados aqui, como o código-fonte do CARMA de amostra e as tarefas para compilá-los.

**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 20. Customizações da versão 9.5

| Membro/Arquivo | Local padrão                                 | Propósito                                                                                | Notas de migração                                                                                     |
|----------------|----------------------------------------------|------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------|
| FEKSETUP       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]              | JCL para criar conjuntos de dados e diretórios e preenchê-los com arquivos customizáveis | Atualizado para remover ações de arquivos que não são mais usados e incluir ações nos novos arquivos. |
| JMON           | FEK.SFEKSAMP(FEJJJCL)<br>[FEK.#CUST.PROCLIB] | JCL para JES Job Monitor                                                                 | Nenhum                                                                                                |
| FEJJJCL        | FEK.SFEKSAMP<br>[FEK.#CUST.PROCLIB(JMON)]    | Nome do membro JMON                                                                      | Consulte o membro de JMON                                                                             |
| RSED           | FEK.SFEKSAMP(FEKRSED)<br>[FEK.#CUST.PROCLIB] | JCL para daemon RSE                                                                      | Nenhum                                                                                                |
| FEKRSED        | FEK.SFEKSAMP<br>[FEK.#CUST.PROCLIB(RSED)]    | Nome do membro RSED                                                                      | Consulte o membro de RSED                                                                             |
| DBGMGR         | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]              | JCL para o Debug Manager                                                                 | SFEKLPA incluído em STEPLIB                                                                           |
| AQEJCL         | FEK.SFEKSAMP<br>[FEK.#CUST.PROCLIB(DBGMGR)]  | Nome para membro DBGMGR                                                                  | Consulte o membro DBGMBR                                                                              |
| ELAXF*         | FEK.SFEKSAMP<br>[FEK.#CUST.PROCLIB]          | JCL para construções de projetos remotos, etc.                                           | SFEKAUTH incluído em STEPLIB                                                                          |
| FEKRACF        | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]              | JCL para definições de segurança                                                         | Novos perfis AQE.**                                                                                   |
| AQERACF        | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]              | JCL for Debug Manager                                                                    | Novos perfis AQE.**                                                                                   |
| FEKPBITS       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]              | JCL para alterar permissões de acesso para arquivos de log                               | Nenhum                                                                                                |

Tabela 20. Customizações da versão 9.5 (continuação)

| Membro/Arquivo | Local padrão                        | Propósito                                                                         | Notas de migração               |
|----------------|-------------------------------------|-----------------------------------------------------------------------------------|---------------------------------|
| FEJJCNGF       | FEK.SFEKSAMP<br>[FEK.#CUST.PARMLIB] | Arquivo de configuração do JES Job Monitor                                        | Nenhum                          |
| FEJTSO         | FEK.SFEKSAMP<br>[FEK.#CUST.CNTL]    | JCL para emissões TSO                                                             | Nenhum                          |
| CRA\$VMSG      | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para criação do VSAM de mensagens do CARMA                                    | Nenhum                          |
| CRA\$VDEF      | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para criação do VSAM de configuração do CARMA                                 | Nenhum                          |
| CRA\$VSTR      | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para criação do VSAM de informações customizadas do CARMA                     | Nenhum                          |
| CRA\$VCAD      | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para criar VSAM da configuração de CARMA para CA Endevor® SCM RAM             | A entrada VSAM foi alterada     |
| CRA\$VCAS      | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para criar VSAM de informações customizadas do CARMA para CA Endevor® SCM RAM | A entrada VSAM foi alterada     |
| CRASUBMT       | FEK.SFEKSAMP<br>[FEK.#CUST.CNTL]    | CLIST de inicialização em lote do CARMA                                           | Nenhum                          |
| CRASUBCA       | FEK.SFEKSAMP<br>[FEK.#CUST.CNTL]    | CLIST de inicialização em lote do CARMA para CA Endevor® SCM RAM                  | Incluído DD APIHJC e novos DD's |

Tabela 20. Customizações da versão 9.5 (continuação)

| Membro/Arquivo | Local padrão                        | Propósito                                                        | Notas de migração               |
|----------------|-------------------------------------|------------------------------------------------------------------|---------------------------------|
| CRACFG         | FEK.SFEKSAMP<br>[FEK.#CUST.PARMLIB] | Configuração de interação CARMA para o CA Endeavor® SCM RAM      | Diretivas opcionais incluídas   |
| CRABCFG        | FEK.SFEKSAMP<br>[FEK.#CUST.PARMLIB] | Configuração de ações em lote do CARMA para CA Endeavor® SCM RAM | Diretivas opcionais incluídas   |
| CRABATCA       | FEK.SFEKSAMP<br>[FEK.#CUST.CNTL]    | JCL de ação em lote do CARMA para CA Endeavor® SCM RAM           | Nenhum                          |
| CRASCL         | FEK.SFEKSAMP<br>[FEK.#CUST.PARMLIB] | SCL do Modelo para CA Endeavor® SCM                              | Novo, a configuração é opcional |
| CRASHOW        | FEK.SFEKSAMP<br>[FEK.#CUST.PARMLIB] | Configuração do CARMA para CA Endeavor® SCM RAM                  | Nenhum                          |
| CRATMAP        | FEK.SFEKSAMP<br>[FEK.#CUST.PARMLIB] | Configuração do CARMA para CA Endeavor® SCM RAM                  | Nenhum                          |
| CRANDVRA       | FEK.SFEKPROC                        | REXX de alocação do CARMA para CA Endeavor® SCM RAM              | Alocar novo DD                  |
| CRADYNDA       | FEK.SFEKPROC                        | REXX de alocação do CARMA para CA Endeavor® SCM RAM              | Novo, a configuração é opcional |
| CRAALLOC       | FEK.SFEKPROC                        | REXX de alocação CARMA                                           | Nenhum                          |
| CRA#VSLM       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para criar VSAM de mensagem do SCLM RAM                      | Nenhum                          |
| CRA#ASLM       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para criar conjuntos de dados do SCLM RAM                    | Nenhum                          |

Tabela 20. Customizações da versão 9.5 (continuação)

| Membro/Arquivo | Local padrão                        | Propósito                                                   | Notas de migração                                              |
|----------------|-------------------------------------|-------------------------------------------------------------|----------------------------------------------------------------|
| CRA#VPDS       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para criar VSAM de mensagem do PDS RAM                  | Nenhum                                                         |
| CRA#UADD       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para mesclagem de definições da RAM                     | Nenhum                                                         |
| CRA#UQRY       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para extração de definições da RAM                      | Nenhum                                                         |
| CRAXJCL        | FEK.SFEKSAMP<br>[FEK.#CUST.ASM]     | Código de origem de amostra para substituição de IRXJCL     | Nenhum                                                         |
| CRA#CIRX       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para compilar CRAXJCL                                   | Nenhum                                                         |
| AQECSD         | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para definir o Integrated Debugger para regiões do CICS | Definir novos recursos                                         |
| AQED3CEE       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para criar módulos de tempo de execução LE customizados | Atualizado para assegurar que a saída de LE correta seja usada |
| AQED3CXT       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para criar a saída de usuário LE                        | Novo nível de depuração para saída de LE                       |
| FEKTEP2        | FEK.SFEKSAMP<br>[FEK.#CUST.SQL]     | Entrada do comando SQL usada pelo ELAXF*                    | Nenhum                                                         |
| FEKTIAD        | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | Entrada do comando SQL usada pelo ELAXF*                    | Nenhum                                                         |
| AZUZUNIT       | FEK.SFEKSAMP<br>[FEK.#CUST.PROCLIB] | Procedimento JCL para zUnit                                 | SFEKAUTH incluído em STEPLIB                                   |

Tabela 20. Customizações da versão 9.5 (continuação)

| Membro/Arquivo        | Local padrão                         | Propósito                                                                      | Notas de migração                                                                                     |
|-----------------------|--------------------------------------|--------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------|
| FEKRNPLI              | FEK.SFEKSAMP<br>[FEK.#CUST.CNTL]     | REXX para chamar o compilador PL/I a partir da estrutura de pré-processador    | Nenhum                                                                                                |
| IRZCSD                | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]      | JCL para definir o Enterprise Service Tools para CICS                          | Novo, a configuração é opcional                                                                       |
| FEKLOGS               | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]      | JCL para coletar arquivos de log                                               | Nenhum                                                                                                |
| rsed.envvars          | /usr/lpp/rdz/samples/<br>[/etc/rdz/] | Variáveis de ambiente do RSE                                                   | As cópias mais antigas devem ser substituídas por esta e as customizações devem ser feitas novamente. |
| ISPF.conf             | /usr/lpp/rdz/samples/<br>[/etc/rdz/] | Arquivo de configuração de Gateway do Cliente TSO/ISPF                         | Nenhum                                                                                                |
| CRASRV.properties     | /usr/lpp/rdz/samples/<br>[/etc/rdz/] | Arquivo de configuração CARMA                                                  | Nenhum                                                                                                |
| crastart.conf         | /usr/lpp/rdz/samples/<br>[/etc/rdz/] | Arquivo de configuração do CARMA para uso de CRASTART                          | Nenhum                                                                                                |
| crastart.endevor.conf | /usr/lpp/rdz/samples/<br>[/etc/rdz/] | Arquivo de configuração do CARMA para uso do CRASTART para CA Endevor® SCM RAM | Incluído DD APIHJC e novos DD's                                                                       |
| include.conf          | /usr/lpp/rdz/samples/<br>[/etc/rdz/] | Inclusões forçadas para assistente de conteúdo C/C++                           | Nenhum                                                                                                |



Tabela 20. Customizações da versão 9.5 (continuação)

| Membro/Arquivo          | Local padrão                         | Propósito                                                    | Notas de migração |
|-------------------------|--------------------------------------|--------------------------------------------------------------|-------------------|
| ssl.properties          | /usr/lpp/rdz/samples/<br>[/etc/rdz/] | Arquivo de configuração de comunicação criptografada do RSE  | Nenhum            |
| rsecomm.properties      | /usr/lpp/rdz/samples/<br>[/etc/rdz/] | Arquivo de configuração de rastreamento RSE                  | Nenhum            |
| pushtoclient.properties | /usr/lpp/rdz/samples/<br>[/etc/rdz/] | Enviar informações para o arquivo de configuração do cliente | Nenhum            |

## Utilitários de Host do IBM Rational Developer for z Systems, FMID HAKG950

- O local de instalação padrão do SMP/E para os componentes MVS e z/OS UNIX não foi alterado e, portanto, permanece AKG.\* e /usr/lpp/rdzutil/\*.
- Desde a versão 9.1.1.1
  - Cobertura de Código: registro do produto necessário durante a compra, como parte do código do produto 5697-CDT, IBM Enterprise COBOL Suite for z/OS
  - Revisão de Código: registro do produto necessário durante a compra, como parte do código do produto 5697-CDT, IBM Enterprise COBOL Suite for z/OS
- Desde a versão 9.1.1.0
  - Revisão de código: novas opções para o procedimento AKGCR
    - SYSLIB: substitui PROPERTY que requeria dados importados do cliente
    - LIST: lista de conjuntos de dados/membros e seu tipo de arquivo (substitui PDS, MEMBERS e EXTMAP)
    - JUNIT: novo formato de saída

### Arquivos Configuráveis

O Tabela 21 na página 128 mostra uma visão geral de arquivos que são customizados na versão 9.5. As bibliotecas de amostra do Developer for z Systems Host Utilities, AKG.SAKGSAMP e /usr/lpp/rdzutil/samples/, contêm mais membros customizáveis do que os listados aqui, como o script de pós-processamento de revisão de código de amostra.

**Nota:** A tarefa de amostra AKGSETUP copia todos os membros listados em conjuntos de dados diferentes, padrão AKG.#CUST.\*.

Tabela 21. Customizações do Host Utilities Versão 9.0

| Membro ou Arquivo | Local padrão                        | Propósito                                                                   | Notas de migração              |
|-------------------|-------------------------------------|-----------------------------------------------------------------------------|--------------------------------|
| AKGSETUP          | AKG.SAKGSAMP<br>[AKG.#CUST.JCL]     | JCL para criar conjuntos de dados e preenchê-los com arquivos customizáveis | Nenhum                         |
| AKGCC             | AKG.SAKGSAMP<br>[AKG.#CUST.PROCLIB] | JCL para cobertura de código                                                | Nenhum                         |
| AKGCR             | AKG.SAKGSAMP<br>[AKG.#CUST.PROCLIB] | JCL para revisão de código                                                  | Novos DDs SYSLIB, LIST e JUNIT |
| AKGCRADD          | AKG.SAKGSAMP<br>[AKG.#CUST.JCL]     | JCL para incluir código de terceiro para revisão de código                  | Nenhum                         |

## Notas de migração da versão 9.1

As notas de migração a seguir são específicas do IBM Rational Developer for z Systems versão 9.1. Essas notas são válidas para migração de IBM Rational Developer for z Systems versão 9.1.0 para a versão 9.1.1, e são inclusões para as notas de migração da versão 9.1.0 existentes.

Todas as mudanças listadas são válidas desde a versão 9.1.1.

### IBM Rational Developer for z Systems, FMID HHOP910

- CARMA: Os arquivos CRADEF e CRASTRS VSAM do CA Endevor® SCM RAM foram atualizados.
- CARMA: Os membros customizáveis a seguir foram alterados:
  - CRACFG define nova opção, ALTERNATIVE-ALLOC
  - CRABCFG define uma nova ação, TRANSFER-ELEMENT
  - CRASUBCA incluiu APIHJC
  - crastart.endevor.conf incluiu APIHJC
- Integrated Debugger: Suporte de coexistência para o IBM Debug Tool for z/OS
- Integrated Debugger: Os membros customizáveis a seguir foram alterados:
  - A tarefa DBGMGR iniciada não utiliza mais o número SVC
  - AQECSD (atualização do CICS CSD) define novos recursos
  - AQERACF define novos perfis AQE.\*\*
  - FEKRACF define novos perfis AQE.\*\*
  - Integrated Debugger: Definições obrigatórias do CICS CSD foram incluídas
    - Filas de dados transientes AQEM e AQED
    - Programa AQEW3Z3, AQEW3Z6, AQEL3Z3, AQEL3Z6, AQEL6Z3, AQEL6Z6
  - Integrated Debugger: Algumas definições do CICS CSD foram removidas
    - Filas de dados transientes CIGZ e CIBM
    - Programa AQEEV006
- Integrated Debugger: SVC requerido foi incluído
  - Mudança de parmlib IEASVCxx para AQESVC03

- Integrated Debugger: SVC opcional foi removido
  - Mudança de parmlib IEASVCxx para AQESVC01
- Integrated Debugger: Perfis necessários do RACF foram incluídos
  - AQE.AUTHDEBUG.STDPGM
  - AQE.AUTHDEBUG.AUTHPGM
- Integrated Debugger: Perfis opcionais do RACF foram removidos
  - AQE.AUTHDEBUG.WRITEBUFFER
- RSE: Suporte à passphrase incluído
- RSE: Diretivas opcionais foram incluídas em rsed.envvars:
  - search.server.limit.scanned\_objects
  - search.server.limit.errcount
- RSE: Diretivas opcionais foram removidas de rsed.envvars:
  - enable.saf.check
  - RSE\_DSICALL
  - search.server.limit.datasets

## Utilitários de Host IBM Rational Developer for z Systems, FMID HAKG910

- Revisão de código: novas opções para o procedimento AKGCR
  - SYSLIB: substitui PROPERTY que requeria dados importados do cliente
  - LIST: lista de conjuntos de dados/membros e seu tipo de arquivo (substitui PDS, MEMBERS e EXTMAP)
  - JUNIT: novo formato de saída

---

## Migrar da versão 9.0 para a versão 9.1

Estas notas se referem a uma migração da versão base 9.0 para a versão 9.1. Ela inclui as mudanças já documentadas como parte da manutenção da versão 9.0. As mudanças que fazem parte do fluxo de manutenção e, portanto, já estão possivelmente implementadas, estão marcadas com a liberação onde elas foram introduzidas.

É recomendado substituir um rsed.envvars existente (por padrão em /etc/rdz) pela amostra fornecida recentemente (por padrão em /usr/lpp/rdz/samples) e refazer as customizações.

## IBM Rational Developer for z Systems, FMID HHOP910

- O local de instalação padrão do SMP/E para os componentes MVS e z/OS UNIX não foi alterado e permanece FEK.\* e /usr/lpp/rdz/\*.
- CARMA: Os membros customizáveis a seguir foram alterados:
  - CRASUBMT
  - CRASUBCA
- Customização: A JCL FEKSETUP agora processa os novos membros:
  - AQED3CEE: copiado para FEK.#CUST.JCL(AQED3CEE)
  - AQED3CXT: copiado para FEK.#CUST.JCL(AQED3CXT)
  - FEKPBITS: copiado para FEK.#CUST.JCL(FEKPBITS)
- RSE: Novas diretivas opcionais foram incluídas no rsed.envvars:
  - (\_RSE\_JAVAOPTS) -Denable.dDVIPA

- (\_RSE\_JAVAOPTS) -Dlog.file.mode
- (\_RSE\_JAVAOPTS) -Dlog.secure.mode
- (\_RSE\_JAVAOPTS) -Denable.saf.check
- (\_RSE\_JAVAOPTS) -Dkeep.stats.copy.local
- (\_RSE\_JAVAOPTS) -Ddebug.miner.localhost
- (\_RSE\_JAVAOPTS) -DDSTORE\_USE\_THREADED\_MINERS
- RSE: Novos comandos do operador foram incluídos:
  - F rsed,APPL=LOG {USER | AUDIT | NOSERVER | OWNER}
- RSE: Novas mensagens do console foram incluídas:
  - FEK220I = Logs do host são gravados em {0}
  - FEK221E = {0} foi interrompido por causa de {1}
  - FEK301E = {0} (uid:{1}) não possui o diretório de {2} (file\_owner uid:{3})
  - FEK302E = O solicitante, {0}, do comando LOGS não tem autoridade para acessar o perfil de {1}
  - FEK303E = O link simbólico, {0}, não pode ser usado como um diretório de log
  - FEK304W = Inválido {0}, {1}, foi especificado. O modo padrão, {3}, é usado no lugar.
  - FEK305E = O ID, {0}, não tem privilégios apropriados para acessar {1}.
- Segurança: O suporte para arquivos de log seguros foi incluído:
  - FEKPBITS: script para alterar permissões para a infraestrutura de log existente
  - Novo comportamento: logs do servidor são, agora, colocados em \$daemon.log/server
  - Novo comportamento: logs são criados com permissões de acesso restrito.
- Segurança: O suporte para novos perfis de segurança foi incluído:
  - FEK.CMD.LOGS.\*\*
  - FEK.REJECT\*.UPDATES.system.group
- desde a versão 9.0.1.1
  - RSE: Novas diretivas opcionais foram incluídas no rsed.envvars:
    - (\_RSE\_JAVAOPTS) -Ddebug.miner.autoreconnect
- desde a versão 9.0.1
  - CARMA: O arquivo CRADEF VSAM para o CA Endevor® SCM RAM foi atualizado.
  - CARMA: O módulo de carregamento CRASTART, que reside no LPA, foi atualizado, requerendo uma atualização do LPA.
  - CARMA: Incluído suporte para executar uma saída do usuário durante a inicialização do CARMA.
  - CARMA: Suporte incluído para argumentos de inicialização de processamento de RAMs.
  - CARMA: Novos membros customizáveis foram incluídos:
    - CRAEXIT: Saída de amostra do usuário do CARMA.
    - CRAALLOC: Executável de alocação para chamadas customizadas do RAM CARMA.
    - CRACFG: Arquivo de configuração de uso CA Endevor® SCM RAM.
  - CARMA: Os membros customizáveis a seguir foram alterados:
    - CRASRV.properties

- crastart.conf
- crastart.endevor.conf
- CRASUBMT
- CRASUBCA
- CRANDVRA
- CARMA: Instruções DD adicionais foram incluídas ao CA Endevor® SCM RAM no crastart.endevor.conf e CRASUBCA:
  - CRAPARM, que é alocado pelo CRANDVRA
  - CRACFG
- CARMA: Instruções DD adicionais estão incluídas em não-“CA Endevor® SCM RAM” no crastart.conf e CRASUBMT:
  - CRAPARM, que é alocado pelo CRAALLOC
- Customização: A JCL FEKSETUP agora processa os novos membros:
  - CRACFG: copiado para FEK.#CUST.PARMLIB(CRACFG)
  - AQEJCL: copiado para FEK.#CUST.PROCLIB(DBGMGR)
  - AQECSD: copiado para FEK.#CUST.JCL(AQECSD)
- Integrated Debugger: Novo serviço opcional
  - Atualizações parmlib IEASVCxx, LPALSTxx e PROGxx (APF e LINKLIST)
  - DBGMGR: tarefa JCL iniciada
  - AQECSD: amostra JCL para atualizar CICS CSD
  - AQERACF: amostra JCL para tornar a configuração de segurança apenas para o Integrated Debugger
- RSE: Membros PROCLIB atualizados
  - ELAXFGO
- RSE: Novas diretivas opcionais foram incluídas ao rsecomm.properties:
  - USER
- RSE: Novos comandos do operador
  - F rsed,APPL=TRACE {USER | SERVER | CLEAR}
- RSE: Novas diretivas opcionais foram incluídas ao rsed.envvars:
  - (\_RSE\_JAVAOPTS) -Dsearch.server.limit.timeout
  - (\_RSE\_JAVAOPTS) -Dkeep.all.logs
  - (\_RSE\_JAVAOPTS) -Daudit.users
  - RSE\_UBLD\_DD
  - RSE\_UBLD\_STEPLIB
- RSE: Novas mensagens do console foram incluídas:
  - FEK910I = FEK107E = Disco quase cheio em {0}, {1} existente removido
- zUnit: Novos argumentos de inicialização opcionais foram incluídos:
  - CLOCALE / -l

**Nota:** Para simplificar a migração de uma configuração existente do Developer for z Systems sem o Depurador Integrado ou os , a JCL de amostra FEK.SFEKSAMP(AQERACF) com comandos RACF é fornecido para definir apenas as definições de segurança relacionadas ao Depurador Integrado.

## Arquivos Configuráveis

O Tabela 22 na página 132 mostra uma visão geral dos arquivos customizados na versão 9.1.0. As bibliotecas de amostra do Developer for z Systems, FEK.SFEKSAMP,

FEK.SFEKSAMV e /usr/lpp/rdz/samples/, contêm mais membros customizáveis do que os listados aqui, como o código-fonte do CARMA de amostra e as tarefas para compilá-los.

**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 22. Customizações da versão 9.1.0*

| Membro/Arquivo | Local padrão                                  | Propósito                                                                                | Notas de migração                                                                                     |
|----------------|-----------------------------------------------|------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------|
| FEKSETUP       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]               | JCL para criar conjuntos de dados e diretórios e preenchê-los com arquivos customizáveis | Atualizado para remover ações de arquivos que não são mais usados e incluir ações nos novos arquivos. |
| JMON           | FEK.SFEKSAMP (FEJJJCL)<br>[FEK.#CUST.PROCLIB] | JCL para JES Job Monitor                                                                 | Nenhum                                                                                                |
| FEJJJCL        | FEK.SFEKSAMP<br>[FEK.#CUST.PROCLIB(JMON)]     | Nome do membro JMON                                                                      | Consulte o membro de JMON                                                                             |
| RSED           | FEK.SFEKSAMP (FEKRSED)<br>[FEK.#CUST.PROCLIB] | JCL para daemon RSE                                                                      | Nenhum                                                                                                |
| FEKRSED        | FEK.SFEKSAMP<br>[FEK.#CUST.PROCLIB(RSED)]     | Nome do membro RSED                                                                      | Consulte o membro de RSED                                                                             |
| DBGMGR         | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]               | JCL para o Debug Manager                                                                 | Novo, a customização é opcional                                                                       |
| AQEJCL         | FEK.SFEKSAMP<br>[FEK.#CUST.PROCLIB(DBGMGR)]   | Nome para membro DBGMGR                                                                  | Consulte o membro DBGMBR                                                                              |
| ELAXF*         | FEK.SFEKSAMP<br>[FEK.#CUST.PROCLIB]           | JCL para construções de projetos remotos, etc.                                           | ELAXFGO inclui SFEKAUTH no STEPLIB                                                                    |
| FEKRACF        | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]               | JCL para definições de segurança                                                         | Tarefa nova, iniciada DBGMGR                                                                          |
| AQERACF        | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]               | JCL for Debug Manager                                                                    | Novo, a customização é opcional                                                                       |
| FEKPBITS       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]               | JCL para alterar permissões de acesso para arquivos de log                               | Novo, a customização é opcional                                                                       |

Tabela 22. Customizações da versão 9.1.0 (continuação)

| Membro/Arquivo | Local padrão                        | Propósito                                                                         | Notas de migração                                                                        |
|----------------|-------------------------------------|-----------------------------------------------------------------------------------|------------------------------------------------------------------------------------------|
| FEJJCNFG       | FEK.SFEKSAMP<br>[FEK.#CUST.PARMLIB] | Arquivo de configuração do JES Job Monitor                                        | Nenhum                                                                                   |
| FEJTSO         | FEK.SFEKSAMP<br>[FEK.#CUST.CNTL]    | JCL para emissões TSO                                                             | Nenhum                                                                                   |
| CRA\$VMSG      | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para criação do VSAM de mensagens do CARMA                                    | Nenhum                                                                                   |
| CRA\$VDEF      | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para criação do VSAM de configuração do CARMA                                 | Nenhum                                                                                   |
| CRA\$VSTR      | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para criação do VSAM de informações customizadas do CARMA                     | Nenhum                                                                                   |
| CRA\$VCAD      | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para criar VSAM da configuração de CARMA para CA Endevor® SCM RAM             | A entrada VSAM foi alterada                                                              |
| CRA\$VCAS      | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para criar VSAM de informações customizadas do CARMA para CA Endevor® SCM RAM | A entrada VSAM foi alterada                                                              |
| CRASUBMT       | FEK.SFEKSAMP<br>[FEK.#CUST.CNTL]    | CLIST de inicialização em lote do CARMA                                           | Incluído suporte para 8 argumentos de inicialização, DD CRAPARM e executável de alocação |

Tabela 22. Customizações da versão 9.1.0 (continuação)

| Membro/Arquivo | Local padrão                        | Propósito                                                        | Notas de migração                                                            |
|----------------|-------------------------------------|------------------------------------------------------------------|------------------------------------------------------------------------------|
| CRASUBCA       | FEK.SFEKSAMP<br>[FEK.#CUST.CNTL]    | CLIST de inicialização em lote do CARMA para CA Endevor® SCM RAM | Incluído suporte para 8 argumentos de inicialização, DD CRAPARM e novos DD's |
| CRACFG         | FEK.SFEKSAMP<br>[FEK.#CUST.PARMLIB] | Configuração de interação CARMA para o CA Endevor® SCM RAM       | Novo, a configuração é opcional                                              |
| CRABCFG        | FEK.SFEKSAMP<br>[FEK.#CUST.PARMLIB] | Configuração de ações em lote do CARMA para CA Endevor® SCM RAM  | Nenhum                                                                       |
| CRABATCA       | FEK.SFEKSAMP<br>[FEK.#CUST.CNTL]    | JCL de ação em lote do CARMA para CA Endevor® SCM RAM            | Nenhum                                                                       |
| CRASHOW        | FEK.SFEKSAMP<br>[FEK.#CUST.PARMLIB] | Configuração do CARMA para CA Endevor® SCM RAM                   | Nenhum                                                                       |
| CRATMAP        | FEK.SFEKSAMP<br>[FEK.#CUST.PARMLIB] | Configuração do CARMA para CA Endevor® SCM RAM                   | Nenhum                                                                       |
| CRANDVRA       | FEK.SFEKPROC                        | REXX de alocação do CARMA para CA Endevor® SCM RAM               | Incluído suporte para saída de usuário e alocar novo DD                      |
| CRAALLOC       | FEK.SFEKPROC                        | REXX de alocação CARMA                                           | Novo, a configuração é opcional                                              |
| CRA#VSLM       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para criar VSAM de mensagem do SCLM RAM                      | Nenhum                                                                       |
| CRA#ASLM       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para criar conjuntos de dados do SCLM RAM                    | Nenhum                                                                       |



Tabela 22. Customizações da versão 9.1.0 (continuação)

| Membro/Arquivo | Local padrão                    | Propósito                                                          | Notas de migração               |
|----------------|---------------------------------|--------------------------------------------------------------------|---------------------------------|
| CRA#VPDS       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL] | JCL para criar VSAM de mensagem do PDS RAM                         | Nenhum                          |
| CRA#UADD       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL] | JCL para mesclagem de definições da RAM                            | Nenhum                          |
| CRA#UQRY       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL] | JCL para extração de definições da RAM                             | Nenhum                          |
| CRAXJCL        | FEK.SFEKSAMP<br>[FEK.#CUST.ASM] | Código de origem de amostra para substituição de IRXJCL            | Nenhum                          |
| CRA#CIRX       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL] | JCL para compilar CRAXJCL                                          | Nenhum                          |
| AQECSD         | FEK.SFEKSAMP<br>[FEK.#CUST.JCL] | JCL para definir o Integrated Debugger para regiões do CICS        | Novo, a configuração é opcional |
| AQED3CEE       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL] | JCL para criar módulos de tempo de execução LE customizados        | Novo, a configuração é opcional |
| AQED3CXT       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL] | JCL para criar a saída de usuário LE                               | Novo, a configuração é opcional |
| ADNCSDRS       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL] | JCL para definir servidor CRD RESTful para região do CICS primária | Nenhum                          |
| ADNCSDTX       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL] | JCL para definir IDs de transação alternativos para região do CICS | Nenhum                          |
| ADNTXNC        | FEK.SFEKSAMP<br>[FEK.#CUST.JCL] | JCL para criação de IDs de transação alternativa                   | Nenhum                          |

Tabela 22. Customizações da versão 9.1.0 (continuação)

| Membro/Arquivo | Local padrão                        | Propósito                                                                      | Notas de migração |
|----------------|-------------------------------------|--------------------------------------------------------------------------------|-------------------|
| ADNMSGHC       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para compilar ADNMSGHS                                                     | Nenhum            |
| ADNMSGHS       | FEK.SFEKSAMP<br>[FEK.#CUST.COBOL]   | Código de origem de amostra para o Manipulador de Mensagem de Pipeline         | Nenhum            |
| ADNVCRD        | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para criar o repositório do CRD                                            | Nenhum            |
| ADNCSDWS       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para definir o servidor CRD de serviço da Web para região do CICS primária | Nenhum            |
| ADNCSDAR       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para definir o servidor CRD para regiões do CICS não primárias             | Nenhum            |
| ADNJSPAU       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para atualizar os padrões do CRD                                           | Nenhum            |
| ADNVMFST       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para criar e definir o repositório do Manifesto                            | Nenhum            |
| FEKTEP2        | FEK.SFEKSAMP<br>[FEK.#CUST.SQL]     | Entrada do comando SQL usada pelo ELAXF*                                       | Nenhum            |
| FEKTIAD        | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | Entrada do comando SQL usada pelo ELAXF*                                       | Nenhum            |
| AZUZUNIT       | FEK.SFEKSAMP<br>[FEK.#CUST.PROCLIB] | Procedimento JCL para zUnit                                                    | Nenhum            |

Tabela 22. Customizações da versão 9.1.0 (continuação)

| Membro/Arquivo        | Local padrão                         | Propósito                                                                      | Notas de migração                                                                                     |
|-----------------------|--------------------------------------|--------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------|
| FEKRNPLI              | FEK.SFEKSAMP<br>[FEK.#CUST.CNTL]     | REXX para chamar o compilador PL/I a partir da estrutura de pré-processador    | Nenhum                                                                                                |
| FEKLOGS               | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]      | JCL para coletar arquivos de log                                               | Nenhum                                                                                                |
| rsed.envvars          | /usr/lpp/rdz/samples/<br>[/etc/rdz/] | Variáveis de ambiente do RSE                                                   | As cópias mais antigas devem ser substituídas por esta e as customizações devem ser feitas novamente. |
| ISPF.conf             | /usr/lpp/rdz/samples/<br>[/etc/rdz/] | Arquivo de configuração de Gateway do Cliente TSO/ISPF                         | Nenhum                                                                                                |
| CRASRV.properties     | /usr/lpp/rdz/samples/<br>[/etc/rdz/] | Arquivo de configuração CARMA                                                  | Incluído suporte para saída de usuário                                                                |
| crastart.conf         | /usr/lpp/rdz/samples/<br>[/etc/rdz/] | Arquivo de configuração do CARMA para uso de CRASTART                          | Incluído suporte para 8 argumentos de inicialização, DD CRAPARM e executável de alocação              |
| crastart.endevor.conf | /usr/lpp/rdz/samples/<br>[/etc/rdz/] | Arquivo de configuração do CARMA para uso do CRASTART para CA Endevor® SCM RAM | Incluído suporte para 8 argumentos de inicialização, DD CRAPARM e novos DD's                          |
| include.conf          | /usr/lpp/rdz/samples/<br>[/etc/rdz/] | Inclusões forçadas para assistente de conteúdo C/C++                           | Nenhum                                                                                                |

Tabela 22. Customizações da versão 9.1.0 (continuação)

| Membro/Arquivo          | Local padrão                         | Propósito                                                    | Notas de migração                        |
|-------------------------|--------------------------------------|--------------------------------------------------------------|------------------------------------------|
| ssl.properties          | /usr/lpp/rdz/samples/<br>[/etc/rdz/] | Arquivo de configuração de comunicação criptografada do RSE  | Nenhum                                   |
| rsecomm.properties      | /usr/lpp/rdz/samples/<br>[/etc/rdz/] | Arquivo de configuração de rastreo RSE                       | Incluído suporte para palavra-chave USER |
| pushtoclient.properties | /usr/lpp/rdz/samples/<br>[/etc/rdz/] | Enviar informações para o arquivo de configuração do cliente | Nenhum                                   |

## Utilitários de Host IBM Rational Developer for z Systems, FMID HAKG910

- O local de instalação padrão do SMP/E para os componentes MVS e z/OS UNIX não foi alterado e, portanto, permanece AKG.\* e /usr/lpp/rdzutil/\*.
- Revisão de código: Instruções DD adicionais são incluídas a AKGCR
  - BEXPORT
  - BIMPORT

### Arquivos Configuráveis

Tabela 23 mostra uma visão geral dos arquivos que são customizados na versão 9.1. As bibliotecas de amostra do Developer for z Systems Host Utilities, AKG.SAKGSAMP e /usr/lpp/rdzutil/samples/, contêm mais membros customizáveis do que os listados aqui, como o script de pós-processamento de revisão de código de amostra.

**Nota:** A tarefa de amostra AKGSETUP copia todos os membros listados em conjuntos de dados diferentes, padrão AKG.#CUST.\*.

Tabela 23. Customizações do Host Utilities Versão 9.0

| Membro ou Arquivo | Local padrão                        | Propósito                                                                   | Notas de migração          |
|-------------------|-------------------------------------|-----------------------------------------------------------------------------|----------------------------|
| AKGSETUP          | AKG.SAKGSAMP<br>[AKG.#CUST.JCL]     | JCL para criar conjuntos de dados e preenchê-los com arquivos customizáveis | Nenhum                     |
| AKGCC             | AKG.SAKGSAMP<br>[AKG.#CUST.PROCLIB] | JCL para cobertura de código                                                | Nenhum                     |
| AKGCR             | AKG.SAKGSAMP<br>[AKG.#CUST.PROCLIB] | JCL para revisão de código                                                  | Novo DDs BIMPORT e BEXPORT |

Tabela 23. Customizações do Host Utilities Versão 9.0 (continuação)

| Membro ou Arquivo | Local padrão                    | Propósito                                                  | Notas de migração |
|-------------------|---------------------------------|------------------------------------------------------------|-------------------|
| AKGCRADD          | AKG.SAKGSAMP<br>[AKG.#CUST.JCL] | JCL para incluir código de terceiro para revisão de código | Nenhum            |

## Migrar da Versão 8.5 para a Versão 9.0

Estas notas são para a migração da versão base 8.5 para a versão 9.0. Elas incluem mudanças que já estão documentadas como parte da manutenção da versão 8.5. As mudanças que fazem parte do fluxo de manutenção e, portanto, já estão possivelmente implementadas, estão marcadas com a liberação onde elas foram introduzidas.

### IBM Rational Developer for z Systems, FMID HHOP900

- O local da instalação padrão do SMP/E para componentes MVS e z/OS UNIX não foi alterado e, portanto, permanece como FEK.\* e /usr/lpp/rdz/\*.
- CARMA: Os arquivos CRADEF e CRASTRS VSAM para CA Endeavor® SCM RAM devem ser atualizados para usarem o novo suporte para ações em lote CA Endeavor® SCM customizáveis (desde a versão 8.5.1).
- CARMA: Suporte incluído para desativar uma RAM durante a criação de RADEF VSAM (desde a versão 8.5.1).
- CARMA: Suporte incluído para referências de arquivo não absoluto em CRASRV.properties (desde a versão 8.5.1).
- CARMA: Novos membros de amostra foram incluídos:
  - CRABJOBC: Cartão JOB padrão para ações de lote CA Endeavor® SCM (desde a versão 8.5.1).
- CARMA: Os membros customizáveis a seguir foram alterados:
  - CRASRV.properties (desde a versão 8.5.1)
  - carma.startup.rex (desde a versão 8.5.1)
  - CRA\$VCAD (desde a versão 8.5.1)
  - CRA\$VDEF (desde a versão 8.5.1)
  - CRABATCA (desde a versão 8.5.1)
  - CRABCFG (desde a versão 8.5.1)
  - CRANDVRA (desde a versão 8.5.1)
- CARMA: Instruções DD adicionais foram incluídas ao CA Endeavor® SCM RAM no crastart.endevor.conf e CRASUBCA:
  - CRABJCLO, que é alocado por CRANDVRA (desde a versão 8.5.1)
  - ENHCEDIT, que é alocado por CRANDVRA (desde a versão 8.5.1)
- Customização: A JCL FEKSETUP agora processa os novos membros:
  - CRABJOBC: copiado para a FEK.#CUST.CNTL(CRABJOBC) (desde a versão 8.5.1)
  - ELAXFSP: copiado para FEK.#CUST.PROCLIB(ELAXFSP) (desde a versão 9.0)
  - ELAXFSQL: copiado para FEK.#CUST.PROCLIB(ELAXFSQL) (desde a versão 9.0)
  - FEKTEP2: copiado para FEK.#CUST.SQL(FEKTEP2) (desde a versão 9.0)
  - FEKTIAD: copiado para FEK.#CUST.SQL(FEKTEP2) (desde a versão 9.0)

- Integração do Analisador de Falhas: o suporte para o FAI foi descontinuado. Essa mudança é incompatível com clientes antigos que ainda usam o FAI.
- Monitor de Tarefas JES - Novos comandos do operador foram incluídos na tarefa iniciada JMON:
  - MODIFY USERS (desde a versão 8.5.1)
  - MODIFY -T{N | E | I | V} (desde a versão 8.5.1)
  - MODIFY -M{N | E | W | I | V} (desde a versão 8.5.1)
  - MODIFY TRACE {N | E | I | V} (desde a versão 9.0)
  - MODIFY MESSAGE {N | E | W | I | V} (desde a versão 9.0)
- JES Job Monitor - Novas diretivas opcionais foram incluídas no FEJJCENFG:
  - LOOPBACK\_ONLY (desde a versão 9.0)
- JES Job Monitor - Diretivas opcionais foram removidas do FEJJCENFG:
  - \_BPXK\_SETIBMOPT\_TRANSPORT (desde a versão 9.0)
- Determinação de problema: A JCL FEKLOGS agora suporta a especificação de diversos IDs de usuário para reunir logs de usuário (desde a versão 8.5.1).
- Determinação de problema: A JCL FEKLOGS agora usa DD REFORMAT para coletar logs reformatados para uma determinação mais rápida de problema (desde a versão 8.5.1).
- Determinação de problema: Os membros customizáveis a seguir foram alterados:
  - FEKLOGS (desde a versão 8.5.1)
- RSE - Os novos comandos do operador foram incluídos à tarefa iniciada RSED:
  - MODIFY DISPLAY OWNER,DATASET=dataset (desde a versão 9.0)
  - MODIFY DEBUG GC,PID=pid (desde a versão 9.0)
- RSE: Novas diretivas não customizáveis foram incluídas ao rsed.envvars:
  - \_CMDSERV\_BASE\_HOME (desde a versão 8.5.1)
  - \_CMDSERV\_CONF\_HOME (desde a versão 8.5.1)
  - \_CMDSERV\_WORK\_HOME (desde a versão 8.5.1)
  - RSE\_DSN\_SFELLOAD (desde a versão 9.0)
  - (\_RSE\_JAVAOPTS) -Dlock.info.timeout (desde a versão 9.0)
  - (\_RSE\_JAVAOPTS) -DDSTORE\_INITIAL\_SIZE (desde a versão 9.0)
  - (\_RSE\_JAVAOPTS) -DDSTORE\_MAX\_FREE (desde a versão 9.0)
- RSE: Novas diretivas necessárias foram incluídas ao rsed.envvars:
  - RSE\_HLQ (desde a versão 9.0)
- RSE: Novas diretivas opcionais foram incluídas ao rsed.envvars:
  - (\_RSE\_JAVAOPTS) -DRSE\_DSICALL (desde a versão 8.5.1)
  - (\_RSE\_JAVAOPTS) -DDISABLE\_REMOTE\_INDEX\_SEARCH (desde a versão 8.5.1)
  - (\_RSE\_JAVAOPTS) -DDISABLE\_TEXT\_SEARCH (desde a versão 9.0)
  - (\_RSE\_JAVAOPTS) -Dsearch.server.limit.hits (desde a versão 9.0)
  - (\_RSE\_JAVAOPTS) -Dsearch.server.limit.datasets (desde a versão 9.0)
  - (\_RSE\_JAVAOPTS) -Dsearch.server.limit.lines (desde a versão 9.0)
  - (\_RSE\_JAVAOPTS) -DDSTORE\_SSL\_ALGORITHM (desde a versão 9.0)
- RSE: O valor padrão para diretivas não customizáveis em rsed.envvars foi alterado:
  - (\_RSE\_JAVAOPTS) -DSPIRIT\_EXPIRY\_TIME (desde a versão 9.0)
- RSE: O valor padrão para diretivas opcionais em rsed.envvars mudou:

- (\_RSE\_JAVAOPTS) -Xms (desde a versão 8.5.1)
- (\_RSE\_JAVAOPTS) -Xmx (desde a versão 8.5.1)
- (\_RSE\_JAVAOPTS) -Dmaximum.clients (desde a versão 8.5.1)
- (\_RSE\_JAVAOPTS) -Dmaximum.threads (desde a versão 8.5.1)
- CGI\_ISPPREF (desde a versão 9.0)
- Segurança: O suporte para novos perfis de segurança foi incluído:
  - FEK.USR.\*\* (desde a versão 8.5.1)

## Arquivos Configuráveis

A Tabela 24 mostra uma visão geral de arquivos que são customizados na versão 9.0. As bibliotecas de amostra do Developer for z Systems, FEK.SFEKSAMP, FEK.SFEKSAMV e /usr/lpp/rdz/samples/, contêm mais membros customizáveis do que os listados aqui, como o código-fonte do CARMA de amostra e as tarefas para compilá-los.

Os seguintes membros e arquivos não são mais customizáveis ou não são mais usados:

- Tarefa iniciada LOCKD
- Procedimento armazenado ELAXMSAM de amostra do DB2
- Procedimento armazenado ELAXMJCL de amostra da JCL para DB2

**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 24. Customizações da versão 9.0*

| Membro/Arquivo | Local padrão                                 | Propósito                                                                                | Notas de migração                                                                                     |
|----------------|----------------------------------------------|------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------|
| FEKSETUP       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]              | JCL para criar conjuntos de dados e diretórios e preenchê-los com arquivos customizáveis | Atualizado para remover ações de arquivos que não são mais usados e incluir ações nos novos arquivos. |
| JMON           | FEK.SFEKSAMP(FEJJJCL)<br>[FEK.#CUST.PROCLIB] | JCL para JES Job Monitor                                                                 | Nenhum                                                                                                |
| FEJJJCL        | FEK.SFEKSAMP<br>[FEK.#CUST.PROCLIB(JMON)]    | Nome do membro JMON                                                                      | Consulte o membro de JMON                                                                             |
| RSED           | FEK.SFEKSAMP(FEKRSED)<br>[FEK.#CUST.PROCLIB] | JCL para daemon RSE                                                                      | Nenhum                                                                                                |
| FEKRSED        | FEK.SFEKSAMP<br>[FEK.#CUST.PROCLIB(RSED)]    | Nome do membro RSED                                                                      | Consulte o membro de RSED                                                                             |

Tabela 24. Customizações da versão 9.0 (continuação)

| Membro/Arquivo | Local padrão                        | Propósito                                                             | Notas de migração                                                                                  |
|----------------|-------------------------------------|-----------------------------------------------------------------------|----------------------------------------------------------------------------------------------------|
| ELAXF*         | FEK.SFEKSAMP<br>[FEK.#CUST.PROCLIB] | JCL para construções de projetos remotos, etc.                        | ELAXFSP e ELAXFSQL são novos, ELAXFCOC e ELAXFCP1 foram atualizados para suportar o Cobol Versão 5 |
| FEKRACF        | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para definições de segurança                                      | Nenhum                                                                                             |
| FEJJCNGF       | FEK.SFEKSAMP<br>[FEK.#CUST.PARMLIB] | Arquivo de configuração do JES Job Monitor                            | Novas diretivas opcionais foram incluídas. As diretivas opcionais existentes foram removidas.      |
| FEJTSO         | FEK.SFEKSAMP<br>[FEK.#CUST.CNTL]    | JCL para emissões TSO                                                 | Nenhum                                                                                             |
| CRA\$VMSG      | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para criação do VSAM de mensagens do CARMA                        | Nenhum                                                                                             |
| CRA\$VDEF      | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para criação do VSAM de configuração do CARMA                     | Incluído suporte para excluir RAMs                                                                 |
| CRA\$VSTR      | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para criação do VSAM de informações customizadas do CARMA         | Nenhum                                                                                             |
| CRA\$VCAD      | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para criar VSAM da configuração de CARMA para CA Endevor® SCM RAM | Incluído suporte para excluir RAMs e a entrada do VSAM foi alterada                                |



Tabela 24. Customizações da versão 9.0 (continuação)

| Membro/Arquivo | Local padrão                        | Propósito                                                                          | Notas de migração                           |
|----------------|-------------------------------------|------------------------------------------------------------------------------------|---------------------------------------------|
| CRA\$VCAS      | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para criar VSAM de informações customizadas do CARMA para CA Endeavor® SCM RAM | A entrada VSAM foi alterada                 |
| CRASUBMT       | FEK.SFEKSAMP<br>[FEK.#CUST.CNTL]    | CLIST de inicialização em lote do CARMA                                            | Nenhum                                      |
| CRASUBCA       | FEK.SFEKSAMP<br>[FEK.#CUST.CNTL]    | CLIST de inicialização em lote do CARMA para CA Endeavor® SCM RAM                  | Nenhum                                      |
| CRABCFG        | FEK.SFEKSAMP<br>[FEK.#CUST.PARMLIB] | Configuração de ações em lote do CARMA para CA Endeavor® SCM RAM                   | Novas diretivas incluídas                   |
| CRBATCA        | FEK.SFEKSAMP<br>[FEK.#CUST.CNTL]    | JCL de ação em lote do CARMA para CA Endeavor® SCM RAM                             | Incluído suporte para o cartão JOB variável |
| CRASHOW        | FEK.SFEKSAMP<br>[FEK.#CUST.PARMLIB] | Configuração do CARMA para CA Endeavor® SCM RAM                                    | Nenhum                                      |
| CRATMAP        | FEK.SFEKSAMP<br>[FEK.#CUST.PARMLIB] | Configuração do CARMA para CA Endeavor® SCM RAM                                    | Nenhum                                      |
| CRANDVRA       | FEK.SFEKPROC                        | REXX de alocação do CARMA para CA Endeavor® SCM RAM                                | Incluídas novas alocações de DD             |

Tabela 24. Customizações da versão 9.0 (continuação)

| Membro/Arquivo | Local padrão                    | Propósito                                                          | Notas de migração |
|----------------|---------------------------------|--------------------------------------------------------------------|-------------------|
| CRA#VSLM       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL] | JCL para criar VSAM de mensagem do SCLM RAM                        | Nenhum            |
| CRA#ASLM       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL] | JCL para criar conjuntos de dados do SCLM RAM                      | Nenhum            |
| CRA#VPDS       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL] | JCL para criar VSAM de mensagem do PDS RAM                         | Nenhum            |
| CRA#UADD       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL] | JCL para mesclagem de definições da RAM                            | Nenhum            |
| CRA#UQRY       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL] | JCL para extração de definições da RAM                             | Nenhum            |
| CRAXJCL        | FEK.SFEKSAMP<br>[FEK.#CUST.ASM] | Código de origem de amostra para substituição de IRXJCL            | Nenhum            |
| CRA#CIRX       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL] | JCL para compilar CRAXJCL                                          | Nenhum            |
| ADNCSDRS       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL] | JCL para definir servidor CRD RESTful para região do CICS primária | Nenhum            |
| ADNCSDTX       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL] | JCL para definir IDs de transação alternativos para região do CICS | Nenhum            |
| ADNTXNC        | FEK.SFEKSAMP<br>[FEK.#CUST.JCL] | JCL para criação de IDs de transação alternativa                   | Nenhum            |

Tabela 24. Customizações da versão 9.0 (continuação)

| Membro/Arquivo | Local padrão                        | Propósito                                                                      | Notas de migração               |
|----------------|-------------------------------------|--------------------------------------------------------------------------------|---------------------------------|
| ADNMSGHC       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para compilar ADNMSGHS                                                     | Nenhum                          |
| ADNMSGHS       | FEK.SFEKSAMP<br>[FEK.#CUST.COBOL]   | Código de origem de amostra para o Manipulador de Mensagem de Pipeline         | Nenhum                          |
| ADNVCRD        | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para criar o repositório do CRD                                            | Nenhum                          |
| ADNCSDWS       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para definir o servidor CRD de serviço da Web para região do CICS primária | Nenhum                          |
| ADNCSDAR       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para definir o servidor CRD para regiões do CICS não primárias             | Nenhum                          |
| ADNJSPAU       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para atualizar os padrões do CRD                                           | Nenhum                          |
| ADNVMFST       | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | JCL para criar e definir o repositório do Manifesto                            | Nenhum                          |
| FEKTEP2        | FEK.SFEKSAMP<br>[FEK.#CUST.SQL]     | Entrada do comando SQL usada pelo ELAXF*                                       | Novo, a customização é opcional |
| FEKTIAD        | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]     | Entrada do comando SQL usada pelo ELAXF*                                       | Novo, a customização é opcional |
| AZUZUNIT       | FEK.SFEKSAMP<br>[FEK.#CUST.PROCLIB] | Procedimento JCL para zUnit                                                    | Nenhum                          |

Tabela 24. Customizações da versão 9.0 (continuação)

| Membro/Arquivo        | Local padrão                         | Propósito                                                                      | Notas de migração                                                                                            |
|-----------------------|--------------------------------------|--------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------|
| FEKRNPLI              | FEK.SFEKSAMP<br>[FEK.#CUST.CNTL]     | REXX para chamar o compilador PL/I a partir da estrutura de pré-processador    | Nenhum                                                                                                       |
| FEKLOGS               | FEK.SFEKSAMP<br>[FEK.#CUST.JCL]      | JCL para coletar arquivos de log                                               | Incluídas verificações adicionais. Qualquer customização nos arquivos mais antigos deve ser feita novamente. |
| rsed.envvars          | /usr/lpp/rdz/samples/<br>[/etc/rdz/] | Variáveis de ambiente do RSE                                                   | As cópias mais antigas devem ser substituídas por esta e as customizações devem ser feitas novamente.        |
| ISPF.conf             | /usr/lpp/rdz/samples/<br>[/etc/rdz/] | Arquivo de configuração de Gateway do Cliente TSO/ISPF                         | Nenhum                                                                                                       |
| CRASRV.properties     | /usr/lpp/rdz/samples/<br>[/etc/rdz/] | Arquivo de configuração CARMA                                                  | Incluído suporte para valores padrão                                                                         |
| crastart.conf         | /usr/lpp/rdz/samples/<br>[/etc/rdz/] | Arquivo de configuração do CARMA para uso de CRASTART                          | Nenhum                                                                                                       |
| crastart.endevor.conf | /usr/lpp/rdz/samples/<br>[/etc/rdz/] | Arquivo de configuração do CARMA para uso do CRASTART para CA Endevor® SCM RAM | Nenhum                                                                                                       |
| include.conf          | /usr/lpp/rdz/samples/<br>[/etc/rdz/] | Inclusões forçadas para assistente de conteúdo C/C++                           | Nenhum                                                                                                       |

Tabela 24. Customizações da versão 9.0 (continuação)

| Membro/Arquivo          | Local padrão                         | Propósito                                                    | Notas de migração |
|-------------------------|--------------------------------------|--------------------------------------------------------------|-------------------|
| ssl.properties          | /usr/lpp/rdz/samples/<br>[/etc/rdz/] | Arquivo de configuração de comunicação criptografada do RSE  | Nenhum            |
| rsecomm.properties      | /usr/lpp/rdz/samples/<br>[/etc/rdz/] | Arquivo de configuração de rastreo RSE                       | Nenhum            |
| pushtoclient.properties | /usr/lpp/rdz/samples/<br>[/etc/rdz/] | Enviar informações para o arquivo de configuração do cliente | Nenhum            |

## IBM Rational Developer for z Systems Host Utilities, FMID HAKG900

Não há nenhuma nota de migração porque não há nenhuma função equivalente na versão 8.5.

### Arquivos Configuráveis

A Tabela 25 mostra uma visão geral de arquivos que são customizados na versão 9.0. As bibliotecas de amostra do Developer for z Systems Host Utilities, AKG.SAKGSAMP e /usr/lpp/rdzutil/samples/, contêm mais membros customizáveis do que os listados aqui, como o script de pós-processamento de revisão de código de amostra.

**Nota:** A tarefa de amostra AKGSETUP copia todos os membros listados em conjuntos de dados diferentes, padrão AKG.#CUST.\*.

Tabela 25. Customizações do Host Utilities Versão 9.0

| Membro ou Arquivo | Local padrão                        | Propósito                                                                   | Notas de migração          |
|-------------------|-------------------------------------|-----------------------------------------------------------------------------|----------------------------|
| AKGSETUP          | AKG.SAKGSAMP<br>[AKG.#CUST.JCL]     | JCL para criar conjuntos de dados e preenchê-los com arquivos customizáveis | Nenhum                     |
| AKGCC             | AKG.SAKGSAMP<br>[AKG.#CUST.PROCLIB] | JCL para cobertura de código                                                | Nenhum                     |
| AKGCR             | AKG.SAKGSAMP<br>[AKG.#CUST.PROCLIB] | JCL para revisão de código                                                  | Novo DDs BIMPORT e BEXPORT |
| AKGCRADD          | AKG.SAKGSAMP<br>[AKG.#CUST.JCL]     | JCL para incluir código de terceiro para revisão de código                  | Nenhum                     |



---

## Capítulo 11. Referência de Configuração do Host

Esta seção resume as informações no *IBM Rational Developer for z Systems Host Configuration Reference (SC27-8578)*. Para obter mais detalhes, consulte essa publicação.

---

### Entendendo o Developer for z Systems

O sistema host do Developer for z Systems consiste em vários componentes que interagem para fornecer ao cliente acesso aos serviços e dados do sistema host. Entender o design desses componentes pode ajudá-lo a tomar as decisões corretas de configuração.

---

### Considerações de segurança

O Developer for z Systems interage com outros componentes de host, que tem implicações de segurança.

---

### Considerações de TCP/IP

O Developer for z Systems usa TCP/IP para fornecer acesso ao mainframe para usuários de uma estação de trabalho sem mainframe. Ele também usa TCP/IP para comunicação entre vários componentes e outros produtos.

---

### Considerações WLM

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

---

### Considerações Push-to-client

O Developer for z Systems estende o direcionar ao cliente do z/OS Explorer ou o controle de cliente baseado em host, com suporte para definições de projeto.

---

### considerações CICSTS

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

---

### Configurando o AT-TLS

Esta seção é fornecida para ajudá-lo com alguns problemas comuns que podem ser encontrados durante a configuração do Application Transparent Layer Security(AT-TLS), ou durante a verificação ou modificação de uma configuração existente.





---

## Bibliografia

---

### Publicações Referenciadas

As publicações a seguir são referenciadas neste documento:

*Tabela 26. Publicações Referenciadas*

| Título da publicação                                                                    | Número da ordem | Referência              | Web site de referência                                                                                                            |
|-----------------------------------------------------------------------------------------|-----------------|-------------------------|-----------------------------------------------------------------------------------------------------------------------------------|
| Program Directory for IBM Rational Developer for z Systems                              | GI11-8298       | Developer for z Systems | <a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a> |
| Program Directory for IBM Rational Developer for z Systems Host Utilities               | GI13-2864       | Developer for z Systems | <a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a> |
|                                                                                         |                 |                         |                                                                                                                                   |
| Guia de Configuração de Host do IBM Rational Developer for z Systems                    | SC27-8577       | Developer for z Systems | <a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a> |
| IBM Rational Developer for z SystemsReferência de Configuração do Host                  | SC27-8578       | Developer for z Systems | <a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a> |
| IBM Rational Developer for z Systems Common Access Repository Manager Developer's Guide | SC23-7660       | Developer for z Systems | <a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a> |
| SCLM Developer Toolkit: Guia do Administrador                                           | SC23-9801       | Developer for z Systems | <a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a> |
| IBM Explorer for z/OS Host Configuration Guide                                          | SC27-8437       | z/OS Explorer           |                                                                                                                                   |
| IBM Explorer for z/OS Host Configuration Reference                                      | SC27-8438       | z/OS Explorer           |                                                                                                                                   |
| Communications Server IP CICS: Guia de Soquetes                                         | SC31-8807       | z/OS 1.13               | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a> |
| Communications Server IP Configuration Guide                                            | SC31-8775       | z/OS 1.13               | <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 1.13               | <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 1.13               | <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 1.13               | <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 1.13               | <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 1.13               | <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 1.13               | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a> |

*Tabela 26. Publicações Referenciadas (continuação)*

| <b>Título da publicação</b>                         | <b>Número da ordem</b> | <b>Referência</b> | <b>Web site de referência</b>                                                                                                     |
|-----------------------------------------------------|------------------------|-------------------|-----------------------------------------------------------------------------------------------------------------------------------|
| Security Server RACF Command Language Reference     | SA22-7687              | z/OS 1.13         | <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 1.13         | <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 1.13         | <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 1.13         | <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 1.13         | <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 1.13         | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a> |

Os Web sites a seguir são referidos neste documento:

*Tabela 27. Web Sites Referidos*

| <b>Descrição</b>                                        | <b>Web site de referência</b>                                                                                                                                                                   |
|---------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Centro de Conhecimento IBM do Developer for z Systems   | <a href="http://www-01.ibm.com/support/knowledgecenter/SSQ2R2/rdz_welcome.html">http://www-01.ibm.com/support/knowledgecenter/SSQ2R2/rdz_welcome.html</a>                                       |
| Biblioteca do Developer for z Systems                   | <a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a>                                                               |
| Developer for z Systems Página inicial do               | <a href="http://www-03.ibm.com/software/products/en/developerforsystemz/">http://www-03.ibm.com/software/products/en/developerforsystemz/</a>                                                   |
| Serviço Recomendado do Developer for z Systems          | <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> |
| Solicitação de Aprimoramento do Developer for z Systems | <a href="https://www.ibm.com/developerworks/support/rational/rfe/">https://www.ibm.com/developerworks/support/rational/rfe/</a>                                                                 |
| Download do Apache Ant                                  | <a href="http://ant.apache.org/">http://ant.apache.org/</a>                                                                                                                                     |

## Publicações Informativas

As publicações a seguir podem ser úteis no entendimento de problemas de configuração para os componentes do sistema host de requisito:

*Tabela 28. Publicações Informativas*

| <b>Título da publicação</b>                            | <b>Número da ordem</b> | <b>Referência</b> | <b>Website 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 28. Publicações Informativas (continuação)*

| <b>Título da publicação</b>                                                          | <b>Número da ordem</b> | <b>Referência</b> | <b>Website de referência</b>                                            |
|--------------------------------------------------------------------------------------|------------------------|-------------------|-------------------------------------------------------------------------|
| Implementação do TCPIP<br>Volume 1: Funções Base,<br>Conectividade e Roteamento      | SG24-7532              | Redbook           | <a href="http://www.redbooks.ibm.com/">http://www.redbooks.ibm.com/</a> |
| TCPIP Implementation<br>Volume 3: High Availability,<br>Scalability, and Performance | SG24-7534              | Redbook           | <a href="http://www.redbooks.ibm.com/">http://www.redbooks.ibm.com/</a> |
| TCP/IP Implementation<br>Volume 4: Security and<br>Policy-Based Networking           | SG24-7535              | Redbook           | <a href="http://www.redbooks.ibm.com/">http://www.redbooks.ibm.com/</a> |
| Tivoli Directory Server for<br>z/OS                                                  | SG24-7849              | Redbook           | <a href="http://www.redbooks.ibm.com/">http://www.redbooks.ibm.com/</a> |



---

## Avisos

Estas informações foram desenvolvidas para produtos e serviços oferecidos nos EUA. Esse material pode estar disponível por meio da IBM em outros idiomas. No entanto, pode ser necessário possuir uma cópia do produto ou da versão do produto nesse idioma a fim de acessá-lo.

É possível que a IBM não ofereça os produtos, serviços ou recursos discutidos nesta publicação em outros países. Consulte um representante IBM local para obter informações sobre produtos e serviços disponíveis atualmente em sua área. Qualquer referência a produtos, programas ou serviços IBM não significa que apenas produtos, programas ou serviços IBM possam ser utilizados. Qualquer produto, programa ou serviço funcionalmente equivalente, que não infrinja nenhum direito de propriedade intelectual da IBM poderá ser utilizado em substituição a este produto, programa ou serviço. Entretanto, a avaliação e verificação da operação de qualquer produto, programa ou serviço não-IBM são de inteira responsabilidade do Cliente.

A IBM pode ter patentes ou solicitações de patente pendentes relativas a assuntos tratados nesta publicação. O fornecimento desta publicação não lhe garante direito algum sobre tais patentes. Pedidos de licença devem ser enviados, por escrito, para:

Gerência de Relações Comerciais e Industriais da *IBM Brasil*  
*IBM Corporation*  
*Av. Pasteur, 138-146*  
*Rio de Janeiro, RJ*  
*BR*

Para consultas sobre licença relacionadas a informações de conjunto de caracteres de byte duplo (DBCS), entre em contato com o Departamento de Propriedade Intelectual da IBM em seu país ou envie consultas, por escrito, para:

*Intellectual Property Licensing*  
*Legal and Intellectual Property Law*  
*IBM Japan Ltd.*  
*19-21, Nihonbashi-Hakozakicho, Chuo-ku*  
*Tokyo 103-8510, Japan*

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 A ELAS NÃO SE LIMITANDO, AS GARANTIAS IMPLÍCITAS DE NÃO INFRAÇÃO, COMERCIALIZAÇÃO OU ADEQUAÇÃO A UM DETERMINADO PROPÓSITO. Alguns países não permitem a exclusão de garantias expressas ou implícitas em certas transações; portanto, essa 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, sem aviso prévio.

Quaisquer referências nestas informações a websites não IBM são fornecidas apenas por conveniência e não servem de maneira alguma como um endosso a esses websites. Os materiais contidos nesses websites não fazem parte dos materiais desse produto IBM e a utilização desses websites é de inteira responsabilidade do Cliente.

A IBM pode usar ou distribuir quaisquer das informações fornecidas da maneira que ela julgar apropriado sem incorrer em qualquer obrigação para com você.

Portadores de Licenças deste programa que desejarem ter informações sobre ele com a finalidade de: (i) troca de informações entre programas criados de forma independente de outros programas (inclusive este) e (ii) o uso mútuo de informações trocadas, deverão entrar em contato com:

Gerência de Relações Comerciais e Industriais da *IBM Brasil*  
*IBM Corporation*  
*Av. Pasteur, 138-146*  
*Rio de Janeiro, RJ*  
*BR*

Estas informações podem estar disponíveis, sujeitas a termos e condições apropriados, 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 Internacional de Licença do Programa IBM ou de qualquer outro contrato equivalente.

Os exemplos de clientes e dados de desempenho citados são apresentados com propósitos meramente ilustrativos. Os resultados de desempenho reais podem variar, dependendo de configurações e condições operacionais específicas.

As informações referentes 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 esses 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 aos fornecedores desses produtos.

As instruções referentes à orientação ou intenção futura da IBM estão sujeitas a mudanças ou cancelamento sem aviso prévio, e representam somente metas e objetivos.

Estas informações contêm exemplos de dados e relatórios utilizados em operações de negócios diárias. Para ilustrá-los da forma mais completa possível, os exemplos incluem nomes de indivíduos, empresas, marcas e produtos. Todos esses nomes são fictícios e qualquer semelhança com pessoas ou empresas reais é mera coincidência.

#### LICENÇA DE COPYRIGHT:

Estas informações contêm programas de aplicativos de amostra na linguagem fonte, ilustrando as técnicas de programação em diversas plataformas operacionais. É permitido copiar, modificar e distribuir esses programas de amostra de qualquer forma sem pagamento à IBM, para propósitos de desenvolvimento, uso, marketing ou distribuição dos programas aplicativos em conformidade com a interface de

programação de aplicativos da plataforma operacional para a qual os programas de amostra foram escritos. Esses exemplos não foram testados completamente em todas as condições. Portanto, a IBM não pode garantir ou implicar a confiabilidade, manutenção ou função destes programas. Os programas de amostra são fornecidos "NO ESTADO EM QUE SE ENCONTRAM", sem garantia de nenhum tipo. A IBM não é responsável por nenhum dano decorrente do uso dos programas de amostra.

---

## **Informações sobre a interface de programação**

---

### **Marcas registradas**

IBM, o logotipo IBM e [ibm.com](http://ibm.com) são marcas ou marcas registradas da International Business Machines Corp., registradas em vários países no mundo todo. Outros nomes de produtos e serviços podem ser marcas registradas da IBM ou de outras empresas. Uma lista atual de marcas comerciais da IBM está disponível na web em "Informações de marca comercial e copyright" em [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

---

## **Termos e condições para a documentação do produto**

---

Permissões para o uso dessas publicações são concedidas sujeitas aos termos e condições a seguir.

### **Aplicabilidade**

Estes termos e condições são incluídos em quaisquer termos de uso do website da IBM.

### **Uso Pessoal**

Você pode reproduzir estas publicações para seu uso pessoal não comercial, contanto que todos os avisos do proprietário sejam preservados. O cliente não pode distribuir, exibir ou criar trabalho derivado destas publicações, ou de qualquer parte delas, sem o consentimento expresso da IBM.

### **Uso Comercial**

Você pode reproduzir, distribuir e exibir estas publicações somente dentro de sua empresa, contanto que todos os avisos do proprietário sejam preservados. O cliente não pode realizar trabalhos derivados destas publicações, ou reproduzir, distribuir ou exibir estas publicações ou de qualquer parte delas fora da empresa, sem o consentimento expresso da IBM.

### **Direitos**

Exceto conforme expressamente concedido nesta permissão, nenhuma outra permissão, licença ou direito serão concedidos, expressos ou implícitos, às publicações ou quaisquer informações, dados, software ou outra propriedade intelectual contida aqui.

A IBM reserva-se o direito de retirar as permissões concedidas aqui, sempre que, a seu critério, o uso das publicações for prejudicial ao seu interesse ou, conforme determinado pela IBM, as instruções anteriores que não estiverem sendo seguidas adequadamente.

O Cliente não pode fazer download, exportar ou exportar novamente essas informações, exceto em conformidade total com todas as leis e regulamentos aplicáveis, incluindo todas as leis e regulamentos dos Estados Unidos.

A IBM NÃO OFERECE GARANTIAS SOBRE O CONTEÚDO DESTAS PUBLICAÇÕES. AS PUBLICAÇÕES SÃO FORNECIDAS "NO ESTADO EM QUE SE ENCONTRAM" E SEM GARANTIA DE NENHUM TIPO, SEJA EXPRESSA OU IMPLÍCITA, INCLUINDO, MAS A ELAS NÃO SE LIMITANDO, AS GARANTIAS IMPLÍCITAS DE NÃO INFRAÇÃO, COMERCIALIZAÇÃO OU ADEQUAÇÃO A UM DETERMINADO PROPÓSITO.

---

## Licença de Copyright

Estas informações contêm programas de aplicativos de amostra na linguagem fonte, ilustrando as técnicas de programação em diversas plataformas operacionais. O Cliente pode copiar, modificar e distribuir estes programas de amostra 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 amostra são criados. Esses exemplos não foram testados completamente em todas as condições. Portanto, a IBM não pode garantir ou implicar a confiabilidade, manutenção ou função destes programas. Os programas de amostra são fornecidos "NO ESTADO EM QUE SE ENCONTRAM", sem garantia de nenhum tipo. A IBM não é responsável por nenhum 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 vários países no mundo todo. Outros nomes de produtos e serviços podem ser marcas registradas da IBM ou de outras empresas. Uma lista atual de marcas registradas da IBM está disponível na Web em [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

Adobe e PostScript são marcas registradas da Adobe Systems Incorporated.

Cell Broadband Engine - Sony Computer Entertainment Inc.

Rational é uma marca registrada da International Business Machines Corporation e Rational Software Corporation, nos Estados Unidos e/ou em outros países.

Intel, Intel Centrino, Intel SpeedStep, Intel Xeon, Celeron, Itanium e Pentium são marcas registradas da Intel Corporation nos Estados Unidos e/ou em outros países.

IT Infrastructure Library é uma marca registrada da Central Computer and Telecommunications Agency

ITIL é uma marca registrada da The Minister for the Cabinet Office

Linear Tape-Open, LTO e Ultrium são marcas registradas da HP, IBM Corp. e Quantum

Linux é uma marca registrada da Linus Torvalds



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 registradas 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 do The Open Group nos Estados Unidos e em outros países.



---

## Comentários do Leitor

IBM Rational Developer for z Systems

Versão 9.5.1:

Guia de Configuração

Publicação N° SC43-2905-00

Neste formulário, faça-nos saber sua opinião sobre este manual. Utilize-o se encontrar algum erro, ou se quiser externar qualquer opinião a respeito (tal como organização, assunto, aparência...) ou fazer sugestões para melhorá-lo.

Para pedir publicações extras, fazer perguntas ou tecer comentários sobre as funções de produtos ou sistemas IBM, fale com o seu representante IBM.

Quando você envia seus comentários, concede direitos, não exclusivos, à IBM para usá-los ou distribuí-los da maneira que achar conveniente, sem que isso implique em qualquer compromisso ou obrigação para com você.

Não se esqueça de preencher seu nome e seu endereço abaixo, se deseja resposta.

Comentários:

---

Nome

---

Endereço

---

Companhia ou Empresa

---

Telefone

IBM Brasil - Centro de Traduções  
Building 501  
Caixa postal 12195  
Research Triangle Park, NC  
EUA





Impresso no Brasil

SC43-2905-00

