

IBM Rational Developer for System z
버전 9.0.1

구성 안내서



IBM Rational Developer for System z
버전 9.0.1

구성 안내서



참고

이 정보를 사용하기 전에 반드시 235 페이지의 『IBM Rational Developer for System z의 문서 주의사항』에 있는 일반 정보를 읽으십시오.

제 11판(2013년 12월)

이 개정판은 새 개정판에 별도로 명시하지 않는 한, IBM Rational Developer for System z 버전 9.0.1(프로그램 번호 5724-T07) 및 모든 후속 릴리스와 수정에 적용됩니다.

전화 또는 팩스로 책을 주문할 수 있습니다. IBM Software Manufacturing Solutions에서는 동부 표준시(EST) 오전 8:30과 오후 7:00 사이에 책 주문을 받습니다. 전화 번호는 (800) 879-2755입니다. 팩스 번호는 (800) 445-9269입니다. 팩스는 Attn: Publications, 3rd floor로 보내셔야 합니다.

한국 IBM 담당자 또는 해당 지역의 IBM 지방 사무소로 책을 주문할 수도 있습니다. 다음 주소에서는 책을 구비하고 있지 않습니다.

IBM은 고객의 의견을 소중하게 생각합니다. 다음 주소로 의견을 보내주십시오.

135-700

서울특별시 강남구 도곡동 467-12,

군인공제회관 빌딩

한국 아이.비.엠 주식회사

고객만족센터

다음 팩스 번호로 의견을 보내주십시오. 1-800-227-5088(미국 및 캐나다)

IBM에 정보를 보내는 경우, IBM은 귀하의 권리를 침해하지 않는 범위 내에서 적절하다고 생각하는 방식으로 귀하가 제공한 정보를 사용하거나 배포할 수 있습니다.

Note to U.S. Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

© Copyright IBM Corporation 2000, 2013.

목차

그림.	vii
표.	ix
이 책 정보.	xi
이 책의 사용자.	xii
변경사항 요약.	xii
문서 컨텐츠 설명.	xv
계획.	xv
기본 사용자 정의.	xv
(선택사항) CARMA(Common Access Repository Manager).	xvi
(선택사항) SCLM 개발자 툴킷.	xvi
(선택사항) 애플리케이션 배치 관리자(더 이상 사 용되지 않음).	xvi
(선택사항) 호스트 기반 코드 분석.	xvii
(선택사항) 기타 사용자 정의 태스크.	xvii
설치 검증.	xviii
보안 정의.	xviii
마이그레이션 안내서.	xviii
운영자 명령.	xviii
호스트 구성 참조서.	xviii

IBM Rational Developer for System z

호스트 구성 안내서. 1

제 1 장 계획.	3
마이그레이션 고려사항.	3
계획 고려사항.	3
제품 개요.	3
스킬 요구사항.	4
시간 요구조건.	4
설치 전 고려사항.	5
설치 사용자 ID.	5
필수 제품.	5
필수 자원.	6
사전 구성 고려사항.	10
워크로드 관리.	10
자원 사용량 및 시스템 한계.	10
필수 제품에 대해 요구되는 구성.	10
사용자 ID 고려사항.	10
서버 고려사항.	12
구성 방법.	12

사전 배치 고려사항.	13
클라이언트 체크리스트.	14

제 2 장 기본 사용자 정의. 17

요구사항 및 체크리스트.	17
사용자 정의 설치.	18
PARMLIB 변경사항.	19
BPXPRMxx에서 z/OS UNIX 한계 설정.	19
COMMNDxx에 시작 태스크 추가.	21
IEASVCxx의 SVC 정의.	22
LPALSTxx의 LPA 정의.	22
PROGxx에 APF 권한 부여.	23
PROGxx의 LINKLIST 정의.	24
필수 LINKLIST 및 LPA 정의.	25
기타 제품에 대한 LINKLIST 정의.	27
PROCLIB 변경사항.	27
JMON, JES 작업 모니터 시작 태스크.	28
DBGMR, 디버그 관리자 시작 태스크.	28
RSED, RSE 디먼 시작 태스크.	29
PARM 변수에 대한 JCL 제한사항.	30
TMPDIR 처리.	32
ELAXF* 원격 빌드 프로시저.	33
보안 정의.	34
FEJJCNFG, JES 작업 모니터 구성 파일.	35
rsed.envvars, RSE 구성 파일.	41
RSE 서버에 사용 가능한 PORTRANGE 정의.	53
_RSE_JAVAOPTS를 사용하여 여분의 Java 시작 매개변수 정의.	54
_RSE_ISPF_OPTS를 사용한 여분의 Java 시작 매개변수 정의.	63
ISPF.conf, ISPF의 TSO/ISPF Client Gateway 구 성 파일.	64
선택적 컴포넌트.	65
설치 검증.	66

제 3 장 (선택사항) CARMA(Common Access

Repository Manager).	69
요구사항 및 체크리스트.	69
서버 시작 메소드 및 활성화 RAM 선택.	70
CARMA 서버 시작.	70
CRASTART.	70
일괄처리 제출.	71

(더 이상 사용되지 않음) TSO/ISPF Client		CRASTRS, 사용자 정의 문자열 데이터 세트	94
Gateway	71	CARMA VSAM 마이그레이션 노트	95
프로덕션 RAM	71	CARMA RAM(Repository Access Manager)	96
CA Endeavor SCM RAM	71	CA Endeavor SCM RAM	96
CA Endeavor SCM 패키지 RAM	71	CA Endeavor SCM 패키지 RAM	97
샘플 RAM	71	PDS RAM	97
PDS RAM	72	스켈레톤 RAM	97
스켈레톤 RAM	72	SCLM RAM	97
SCLM RAM	72	CRACFG, CRASHOW 및 CRATMAP: CA	
사전 구성된 RAM 및 서버 시작 조합	72	Endeavor SCM RAM 구성 파일	98
CA Endeavor SCM RAM을 사용한 CRASTART	72	CRACFG: SCM과 CA Endeavor SCM RAM	
CARMA VSAM 데이터 세트 작성	73	의 상호작용	98
CRASRV.properties 사용자 정의	73	CRASHOW, CA Endeavor SCM RAM 기본	
crastart.endeavor.conf 사용자 정의	73	필터	99
(선택사항) 추가적 CA Endeavor SCM RAM 사		CRATMAP, CA Endeavor SCM RAM 파일	
용자 정의	75	확장자 맵핑	99
샘플 RAM을 사용한 CRASTART	75	CRANDVRA: CA Endeavor SCM RAM 할당	
CARMA VSAM 데이터 세트 작성	75	exec.	99
CARMA	76	CA Endeavor SCM RAM 일괄처리 조치	100
샘플 RAM	76	CRABCFG, CA Endeavor SCM RAM 일괄	
CRASRV.properties 사용자 정의	76	처리 조치 구성	101
crastart.conf 사용자 정의	76	CRABATCA, CA Endeavor SCM RAM 일	
I (선택사항) 추가 사용자 설치 RAM의 사용자 정의	77	괄처리 조치 JCL	102
CA Endeavor SCM RAM을 사용한 일괄처리 제출	78	CRABJOBC, CA Endeavor SCM RAM 일	
CARMA VSAM 데이터 세트 작성	78	괄처리 조치 JOB 카드	103
CRASRV.properties 사용자 정의	78	I CRAALLOC: 사용자 정의 RAM 할당 exec	104
CRASUBCA 사용자 정의	78	CARMA 리턴 코드	104
(선택사항) 추가적 CA Endeavor SCM RAM 사		(선택사항) 다중 RAM 지원	105
용자 정의	81	예제	105
샘플 RAM을 사용한 일괄처리 제출	81	(선택사항) 사용자 정의 할당 exec	106
VSAM 데이터 세트 작성	81	I (선택사항) CARMA 사용자 엑시트	107
CARMA	81	(선택사항) IRXJCL 대 CRAXJCL	108
샘플 RAM	82	CRAXJCL 작성	109
CRASRV.properties 사용자 정의	82		
CRASUBMT 사용자 정의	82	제 4 장 (선택사항) SCLM 개발자 툴킷	111
I (선택사항) 추가 사용자 설치 RAM의 사용자 정		요구사항 및 체크리스트	111
I 의	83	전제조건	112
CARMA 구성 세부사항	84	SCLMDT의 ISPF.conf 업데이트	113
CRASRV.properties, CARMA에 대한 RSE 인		SCLMDT에 대한 rsed.envvars 업데이트	113
터페이스	84	(선택사항) 긴/짧은 이름 변환	114
crastart*.conf, CRASTART 서버 시작	88	LSTRANS.FILE 작성, 긴/짧은 이름 변환	
CRASTART 로그 파일 수집	91	VSAM	114
CRASUB*, 일괄처리 제출 서버 시작	92	긴/짧은 이름 변환을 위한 rsed.envvars 업데이트	116
CARMA VSAM 데이터 세트	94	(선택사항) Ant 설치 및 사용자 정의	116
CRADef, 구성 데이터 세트	94	SCLMDT에 대한 SCLM 업데이트	118
CRAMSG, 메시지 데이터 세트	94	WORKAREA 및 /tmp에서 이전 파일 제거	118

제 5 장 (선택사항) 애플리케이션 배치 관리자(더 이상 사용되지 않음)	119	(선택사항) WORKAREA 및 /tmp 정리	152
요구사항 및 체크리스트	119	제 8 장 설치 검증	153
CRD 저장소	120	시작 태스크 확인	153
CICS 관리 유틸리티	121	JMON, JES 작업 모니터	153
RESTful 대 웹 서비스	121	RSED, RSE 디먼.	153
RESTful 인터페이스를 사용하는 CRD 서버	122	DBGMGR, 디버그 관리자.	158
CICS 기본 연결 리전	122	IVP 운영자 명령	158
CICS 비기본 연결 리전.	122	PassTicket 재사용가능성	159
(선택사항) CRD 서버 트랜잭션 ID 사용자 정의	123	RSE 디먼 연결.	159
웹 서비스 인터페이스를 사용하는 CRD 서버.	123	ISPF Client Gateway	159
파이프라인 메시지 핸들러	124	서비스 확인	161
CICS 기본 연결 리전	125	IVP 초기화	161
CICS 비기본 연결 리전.	125	포트 사용 가능성	162
(선택사항) Manifest 저장소	126	TCP/IP 설정	163
제 6 장 (선택사항) 호스트 기반 코드 분석	127	RSE 디먼 연결.	164
요구사항 및 체크리스트	127	JES 작업 모니터 연결	165
코드 검토	127	ISPF의 TSO/ISPF Client Gateway 연결.	165
코드 검토 처리 수정	128	(선택사항) CARMA 연결	167
코드 적용	128	(선택사항) SCLMDT 연결.	168
코드 적용 결과물	129	(선택사항) 통합 디버거 연결	170
제 7 장 (선택사항) 기타 사용자 정의 태스크.	131	제 9 장 보안 정의	171
(선택사항) pushtoclient.properties, 호스트 기반 클라이언트 제어	131	요구사항 및 체크리스트	171
(선택사항) ssl.properties, RSE SSL 암호화	135	보안 설정 및 클래스 활성화	173
(선택사항) rsecomm.properties, RSE 추적	138	Developer for System z 사용자에게 대한 OMVS 세그먼트 정의	174
(선택사항) include.conf, C/C++ 콘텐츠 지원을 위해 강제로 포함됨	139	Developer for System z 시작 태스크 정의	174
(선택사항) z/OS UNIX 서브프로젝트	140	RSE를 보안 z/OS UNIX 서버로 정의.	176
REXEC 또는 SSH 설정	141	RSE에 대한 MVS 프로그램 제어 라이브러리 정의	176
(선택사항) 프리프로세서 지원 포함	141	RSE에 대한 PassTicket 지원 정의	177
(선택사항) Enterprise COBOL 및 PL/I용 xUnit 지원	142	RSE에 대한 애플리케이션 보호 정의	179
(선택사항) 엔터프라이즈 서비스 도구 지원.	144	JES 명령 보안 정의	179
(선택사항) CICS 양방향 언어 지원	144	데이터 세트 프로파일 정의.	181
(선택사항) 생성된 코드의 진단 IRZ 메시지	145	RSE에 대한 z/OS UNIX 프로그램 제어 파일 정의	187
(선택사항) 통합 디버거	146	보안 설정 확인.	187
통합 디버거 구성 매개변수.	148	제 10 장 마이그레이션 안내서	189
통합 디버거 parmlib 업데이트	148	마이그레이션 고려사항	189
통합 디버거 TCP/IP 업데이트.	149	이전에 구성된 파일 백업	189
통합 디버거 보안 업데이트.	149	버전 9.0 마이그레이션 참고	190
통합 디버거 CICS 업데이트	149	IBM Rational Developer for System z, FMID HHOP900.	190
(선택사항) 문제점 판별 도구 지원	150	IBM Rational Developer for System z 호스트 유틸리티, FMID HAKG900	191
(선택사항) DB2 및 IMS 디버그 지원	150	버전 8.5에서 버전 9.0으로 마이그레이션	191
(선택사항) 파일 관리자 지원	151		

IBM Rational Developer for System z, FMID HHOP900.	191	한 라인 초과	226
구성 가능한 파일	194	구문 단편	226
IBM Rational Developer for System z <u>호스트</u> 유틸리티, FMID HAKG900	199		
구성 파일	199	제 12 장 호스트 구성 참조서	227
버전 8.5 마이그레이션 참고	199	Developer for System z 이해	227
버전 8.0.1에서 버전 8.5로 마이그레이션	201	보안 고려사항	227
IBM Rational Developer for System z, FMID HHOP850.	201	TCP/IP 고려사항	227
구성 가능한 파일	204	WLM 고려사항	227
제 11 장 운영자 명령	209	튜닝 고려사항	227
Start(S)	209	성능 고려사항	228
I 통합 디버거	209	클라이언트로 푸시 고려사항	228
JES 작업 모니터	210	CICSTS 고려사항.	228
RSE 디먼	211	사용자 종료 고려사항	228
Modify(F)	211	TSO 환경 사용자 정의	228
I 통합 디버거	211	다중 인스턴스 실행	228
JES 작업 모니터	213	구성 문제점 해결	229
RSE 디먼	215	SSL 및 X.509 인증 설정	229
Stop(P)	224	TCP/IP 설정	229
구문 다이어그램 읽는 방법.	224	참고 문헌	231
기호	225	참조된 서적.	231
피연산자	225	정보 서적	234
구문 예제	225		
영숫자가 아닌 문자 및 공백	225	IBM Rational Developer for System z의 문서	
둘 이상의 피연산자 선택하기	226	주의사항	235
		저작권 라이선스	238
		상표 정보	238

그림

1. JMON: JES 작업 모니터 시작 태스크	28	23. CRASUB*: 일괄처리 제출을 사용하는 CARMA 시작	93
2. DBGMGR: 디버그 관리자 시작 태스크	29	24. CRACFG: SCM과 CA Endeavor SCM RAM 의 상호작용	98
3. RSED: RSE 디먼 시작 태스크	30	25. CRASHOW - CA Endeavor SCM RAM 기 본 필터	99
4. RSED: 대체 RSE 디먼 시작	31	26. CRATMAP: CA Endeavor SCM RAM 기본 필터	99
5. rsed.stdin.sh: 대체 RSE 디먼 시작	31	27. CRABCFG: CA Endeavor SCM RAM 일괄 처리 조치 구성	102
6. RSED: 대체 TMPDIR 처리	32	28. CRABATCA: CA Endeavor SCM RAM 일 괄처리 조치 JCL	103
7. rsed.stdenv: 대체 TMPDIR 처리	32	29. CRABJOBC: CA Endeavor SCM RAM 일 괄처리 조치 JOB 카드	103
8. FEJJCNFG, JES 작업 모니터 구성 파일	36	30. SCLMDT의 ISPF.conf 업데이트	113
9. rsed.envvars: RSE 구성 파일	42	31. SCLMDT에 대한 rsed.envvars 업데이트	114
10. rsed.envvars: RSE 구성 파일(계속)	44	32. FLM02LST: 긴/짧은 이름 변환 설정 JCL	115
11. rsed.envvars: RSE 구성 파일(계속)	45	33. pushtoclient.properties: 호스트 기반 클라이언 트 제어 구성 파일	132
12. ISPF.conf: ISPF 구성 파일	64	34. ssl.properties - SSL 구성 파일	136
13. CRASRV.properties: CA Endeavor SCM RAM을 사용하는 CRASTART	73	35. rsecomm.properties - 로깅 구성 파일	138
14. crastart.endeavor.conf: CA Endeavor SCM RAM 사용하는 CRASTART	74	36. include.conf - C/C++ 콘텐츠 지원을 위한 강 제 포함	140
15. CRASRV.properties: 샘플 RAM을 사용하는 CRASTART	76	37. START DBGMGR 운영자 명령	209
16. crastart.conf: 샘플 RAM을 사용하는 CRASTART	77	38. START JMON 운영자 명령	210
17. CRASRV.properties: CA Endeavor SCM RAM을 사용하는 일괄처리 제출	78	39. START RSED 운영자 명령	211
18. CRASUBCA: CA Endeavor SCM RAM을 사 용한 일괄처리 제출	80	40. MODIFY DBGMGR 운영자 명령	212
19. CRASRV.properties: 샘플 RAM을 사용하는 일괄처리 제출	82	41. MODIFY JMON 운영자 명령	213
20. CRASUBMT: 샘플 RAM을 사용하는 일괄처 리 제출	83	42. MODIFY RSED 운영자 명령	215
21. CRASRV.properties - CARMA 구성 파일	84	43. STOP 운영자 명령	224
22. crastart*.conf: CRASTART를 사용하는 CARMA 서버 시작	90		

표

1. 필수 자원	6	16. 클라이언트로 푸시 그룹 지원	134
2. 선택적 자원	6	17. SSL 인증서 스토리지 메커니즘	135
3. 필수 태스크에 필요한 관리자	7	18. 유효한 키 저장소 유형	137
4. 선택적 태스크에 필요한 관리자	8	19. 서비스를 위한 IVP	161
5. 클라이언트 체크리스트: 필수 부분	14	20. 보안 설정 변수	171
6. 클라이언트 체크리스트: 선택적 부분	14	21. JES2 작업 모니터 운영자 명령	180
7. 기능과 로드 모듈 일치	24	22. JES3 작업 모니터 운영자 명령	180
8. 샘플 ELAXF* 프로시저	33	23. 버전 9.0 사용자 정의	194
9. ELAXF* 상위 레벨 규정자 체크리스트	34	24. 호스트 유틸리티 버전 9.0 사용자 정의	199
10. ELAXF*	34	25. 버전 8.5 사용자 정의	205
11. LIMIT_COMMANDS 명령 권한 매트릭스	38	26. 스레드 풀 오류 상태	217
12. CARMA 리턴 코드	104	27. 참조된 서적	231
13. SCLM 관리자 체크리스트	118	28. 참조된 웹 사이트	233
14. 기본 CRD 서버 트랜잭션 ID	123	29. 정보 서적	234
15. 기본 CRD 서버 트랜잭션 ID	124		

이 책 정보

이 책에서는 IBM® Rational® Developer for System z® 기능의 구성에 대해 설명합니다. 여기에는 z/OS® 호스트 시스템에서 IBM Rational Developer for System z 버전 9.0.1을 구성하기 위한 지시사항이 포함됩니다.

지금부터 이 매뉴얼에서는 다음 이름을 사용합니다.

- *IBM Rational Developer for System z*는 *Developer for System z*라고 합니다.
- *IBM Rational Developer for System z* 통합 디버거는 통합 디버거라고 합니다.
- *Common Access Repository Manager*의 약어는 *CARMA*입니다.
- *Software Configuration and Library Manager Developer Toolkit*은 *SCLM* 개발자 툴킷이라고 하며 약어는 *SCLMDT*입니다.
- *IBM z/OS Automated Unit Testing Framework*는 *zUnit*이라고 합니다.
- *z/OS UNIX System Services*는 *z/OS UNIX*라고 합니다.
- *Customer Information Control System Transaction Server*는 *CICSTS*라고 하며 약어는 *CICS*®입니다.

이 문서는 Developer for System z 호스트 시스템 구성에 대해 설명하는 문서 세트의 일부입니다. 이들 문서는 각각 특정 사용자를 대상으로 합니다. Developer for System z 구성을 완성하는 데 모든 문서를 읽을 필요가 없습니다.

- *System z용 Rational Developer 호스트 구성 안내서(SA30-4578)*에서는 모든 계획 태스크, 구성 태스크 및 옵션(선택적 옵션 포함)에 대해 자세히 설명하며 대체 시나리오를 제공합니다.
- *System z용 Rational Developer 호스트 구성 참조서(SA30-4501)*에서는 Developer for System z 디자인에 대해 설명하고 Developer for System z, z/OS 컴포넌트 및 Developer for System z와 관련된 다른 제품(예: WLM 및 CICS)의 다양한 구성 태스크에 대한 배경 정보를 제공합니다.
- *System z용 Rational Developer 호스트 구성 빠른 시작 안내서(GA30-4183)*에서는 Developer for System z의 최소 설정에 대해 설명합니다.
- *System z용 Rational Developer 호스트 구성 유틸리티 안내서(SA30-4582)*는 Developer for System z에 대한 기본적인 공통적인 선택적 사용자 정의 단계를 안내하는 ISPF 패널 애플리케이션인 *호스트 구성 유틸리티*를 설명합니다.

이 문서의 정보는 IBM Rational Developer for System z 버전 9.0의 모든 패키지에 적용됩니다.

이 책의 사용자

이 문서는 IBM Rational Developer for System z 버전 9.0.1을 설치하고 구성하는 시스템 프로그래머를 위한 것입니다.

이 문서는 기본이 아닌 일부 시나리오를 포함하여 제품의 전체 설정을 수행하는 데 필요한 단계에 대해 상세히 나열합니다. 구성을 계획하고 실행하는 데 도움이 될 수 있는 배경 정보는 *IBM Rational Developer for System z 호스트 구성 참조서(SA30-4501)*에 있습니다. 이 문서를 사용하려면 z/OS UNIX 시스템 서비스 및 MVS™ 호스트 시스템에 대해 잘 알아야 합니다.

변경사항 요약

이 절에서는 *IBM Rational Developer for System z 버전 9.0.1 호스트 구성 안내서, SA30-4578-10*(2013년 12월 업데이트) 변경사항을 요약합니다.

텍스트 및 그림에 대한 기술적 변경사항이나 추가사항은 변경사항 왼쪽에 세로선으로 표시됩니다.

새 정보:

- 버전 9.0.1 마이그레이션 정보. 190 페이지의 『버전 9.0 마이그레이션 참고』의 내용을 참조하십시오.
- 새 선택적 PARMLIB 업데이트. 19 페이지의 『PARMLIB 변경사항』의 내용을 참조하십시오.
- 새 선택적 시작 태스크. 27 페이지의 『PROCLIB 변경사항』의 내용을 참조하십시오.
- 새 운영자 명령. 209 페이지의 제 11 장 『운영자 명령』의 내용을 참조하십시오.
- rsed.envvars의 신규 지시문 및 변경된 지시문. 41 페이지의 『rsed.envvars, RSE 구성 파일』의 내용을 참조하십시오.
- CRASRV.properties의 새 지시문. 84 페이지의 『CRASRV.properties, CARMA에 대한 RSE 인터페이스』의 내용을 참조하십시오.
- CARMA CA Endeavor® SCM RAM의 새 구성 파일. 98 페이지의 『CRACFG, CRASHOW 및 CRATMAP: CA Endeavor® SCM RAM 구성 파일』의 내용을 참조하십시오.
- CARMA 시작 중에 호출된 사용자 엑시트를 지원합니다. 107 페이지의 『(선택사항) CARMA 사용자 엑시트』의 내용을 참조하십시오.

이 문서에는 *IBM Rational Developer for System z 버전 9.0 호스트 구성 안내서, SA30-4578-09*에서 이전에 제공된 정보가 포함되어 있습니다.

새 정보:

- 버전 9.0 마이그레이션 정보. 191 페이지의 『버전 8.5에서 버전 9.0으로 마이그레이션』의 내용을 참조하십시오.
- rsed.envvars의 신규 지시문 및 변경된 지시문. 41 페이지의 『rsed.envvars, RSE 구성 파일』의 내용을 참조하십시오.
- FEJCNFG의 신규 지시문 및 제거된 지시문. 35 페이지의 『FEJCNFG, JES 작업 모니터 구성 파일』의 내용을 참조하십시오.
- 새 ELAXF* PROCLIB 멤버. 33 페이지의 『ELAXF* 원격 빌드 프로시저』의 내용을 참조하십시오.
- 새 JMON 및 RSED 운영자 명령. 211 페이지의 『Modify(F)』의 내용을 참조하십시오.
- 호스트 기반 코드 분석에 대한 정보가 추가되었습니다. 127 페이지의 제 6 장 『(선택사항) 호스트 기반 코드 분석』의 내용을 참조하십시오.

제거된 정보:

- LOCKD 시작 태스크는 더이상 사용되지 않으므로 잠금 디먼에 대한 모든 정보도 제거됩니다.
- 샘플 DB2[®] 스토어드 프로시저가 새 ELAXF* 빌드 프로시저로 대체되었으므로 DB2 스토어드 프로시저에 대한 모든 정보는 제거됩니다.
- 지원하지 않는 릴리스에 대한 마이그레이션 정보를 제거합니다.

이 문서에는 *IBM Rational Developer for System z 버전 8.5.1 호스트 구성 안내서*, SA30-4578-08에서 이전에 제공되었던 정보가 포함되어 있습니다.

새 정보:

- 버전 8.5.1 마이그레이션 정보. 199 페이지의 『버전 8.5 마이그레이션 참고』의 내용을 참조하십시오.
- rsed.envvars의 신규 지시문 및 변경된 지시문. 41 페이지의 『rsed.envvars, RSE 구성 파일』의 내용을 참조하십시오.
- CA Endevor[®] SCM RAM 일괄처리 조치를 위한 향상된 지원. 100 페이지의 『CA Endevor[®] SCM RAM 일괄처리 조치』의 내용을 참조하십시오.
- 설치 사용자 ID 요구사항. 5 페이지의 『설치 사용자 ID』의 내용을 참조하십시오.

이 문서에는 *IBM Rational Developer for System z 버전 8.5 호스트 구성 안내서*, SC23-7658-07에서 이전에 제공된 정보가 포함되어 있습니다.

새 정보:

- FEJCNFG의 새 선택적 지시문. 35 페이지의 『FEJCNFG, JES 작업 모니터 구성 파일』의 내용을 참조하십시오.

- rsed.envvars의 새 선택적 지시문. 41 페이지의 『rsed.envvars, RSE 구성 파일』의 내용을 참조하십시오.
- 새 선택적 구성 파일. 139 페이지의 『(선택사항) include.conf, C/C++ 콘텐츠 지원을 위해 강제로 포함됨』의 내용을 참조하십시오.
- 새 선택적 컴포넌트. 141 페이지의 『(선택사항) 프리프로세서 지원 포함』의 내용을 참조하십시오.
- 새 선택적 컴포넌트. 142 페이지의 『(선택사항) Enterprise COBOL 및 PL/I용 xUnit 지원』의 내용을 참조하십시오.
- 새 선택적 컴포넌트. 150 페이지의 『(선택사항) DB2 및 IMS 디버그 지원』의 내용을 참조하십시오.
- 신규 및 개선된 운영자 명령. 209 페이지의 제 11 장 『운영자 명령』의 내용을 참조하십시오.
- 버전 8.5 마이그레이션 정보. 201 페이지의 『버전 8.0.1에서 버전 8.5로 마이그레이션』의 내용을 참조하십시오.

제거된 정보:

- 파일 관리자 통합에 대한 지원이 변경되어 151 페이지의 『(선택사항) 파일 관리자 지원』 절에 있는 대부분의 정보가 제거되었습니다.

이 문서에는 *Rational Developer for System z* 버전 8.0.3 호스트 구성 안내서, SA30-4578-06에서 이전에 제공되었던 정보가 포함되어 있습니다.

새 정보:

- rsed.envvars의 새 지시문. 41 페이지의 『rsed.envvars, RSE 구성 파일』의 내용을 참조하십시오.
- CA Endevor® SCM 배경 조치에 대해 추가된 지원. 100 페이지의 『CA Endevor® SCM RAM 일괄처리 조치』의 내용을 참조하십시오.
- CA Endevor® SCM 패키지에 대해 추가된 지원. 96 페이지의 『CARMA RAM(Repository Access Manager)』의 내용을 참조하십시오.
- pushtoclient.properties의 새 지시문. 131 페이지의 『(선택사항) pushtoclient.properties, 호스트 기반 클라이언트 제어』의 내용을 참조하십시오.
- 파일 관리자 통합은 더 이상 사용되지 않습니다. 151 페이지의 『(선택사항) 파일 관리자 지원』의 내용을 참조하십시오.
- 신규 및 개선된 운영자 명령. 209 페이지의 제 11 장 『운영자 명령』의 내용을 참조하십시오.
- 새 서적, *IBM Rational Developer for System z* 메시지 및 코드(SA30-4577).

이 문서에는 *Rational Developer for System z* 버전 8.0.1 호스트 구성 안내서, SA30-4578-05에서 이전에 제공되었던 정보가 포함되어 있습니다.

새 정보:

- FEJJCNFG의 새 지시문. 35 페이지의 『FEJJCNFG, JES 작업 모니터 구성 파일』의 내용을 참조하십시오.
- rsed.envvars의 새 지시문. 41 페이지의 『rsed.envvars, RSE 구성 파일』의 내용을 참조하십시오.
- CARMA 장이 재구성되었고 일부 신규 정보가 추가되었습니다. 69 페이지의 제 3 장 『(선택사항) CARMA(Common Access Repository Manager)』의 내용을 참조하십시오.
- 새 구성 파일 pushtoclient.properties. 131 페이지의 『(선택사항) pushtoclient.properties, 호스트 기반 클라이언트 제어』의 내용을 참조하십시오.

제거된 정보:

- *Rational Developer for System z* 버전 7.6.1 호스트 구성 안내서(SA30-4578-04)에서 이전에 제공되었던 정보는 이제 *Rational Developer for System z 호스트* 구성 안내서(SA30-4578) 및 *Rational Developer for System z 호스트* 구성 참조서(SA30-4501) 두 문서로 분리됩니다.
- APPC 설정에 관한 정보는 *Using APPC to provide TSO command services* (SC14-7291) 백서로 이동되었습니다.
- ISPF 클라이언트 게이트웨이를 사용하는 CARMA에 관한 정보는 *Using ISPF Client Gateway to provide CARMA services*(SC14-7292) 백서로 이동되었습니다.
- "(선택적) 기타 사용자 정의 태스크"의 "(선택적) 호스트 기반 특성 그룹" 절(propertiescfg.properties에 대해 설명함)
- "(선택적) 기타 사용자 정의 태스크"의 "(선택적) 호스트 기반 프로젝트" 절(projectcfg.properties에 대해 설명함)
- "(선택적) 기타 사용자 정의 태스크"의 "(선택적) 편집 불가능한 문자" 절(uchars.settings에 대해 설명함)
- "마이그레이션 가이드"의 "버전 7.6.1 마이그레이션 참고" 절

문서 콘텐츠 설명

이 절에서는 이 문서에서 제공하는 정보를 요약합니다.

계획

Developer for System z의 설치 및 배치를 계획하려면 이 장의 정보를 이용하십시오.

기본 사용자 정의

다음 사용자 정의 단계는 기본 Developer for System z 설정을 위한 것입니다.

- 18 페이지의 『사용자 정의 설치』

- 19 페이지의 『PARMLIB 변경사항』
- 27 페이지의 『PROCLIB 변경사항』
- 34 페이지의 『보안 정의』
- 35 페이지의 『FEJCNFG, JES 작업 모니터 구성 파일』
- 41 페이지의 『rsed.envvars, RSE 구성 파일』
- 64 페이지의 『ISPF.conf, ISPF의 TSO/ISPF Client Gateway 구성 파일』

(선택사항) CARMA(Common Access Repository Manager)

CARMA(Common Access Repository Manager)는 RAM(Repository Access Manager)용 서버 플랫폼입니다. RAM은 z/OS 시스템이 기반인 SCM(Software Configuration Manager)용 API(Application Programming Interface(API))입니다. RAM에서 SCM 기능을 랩핑하면 클라이언트는 단일 API를 사용하여 지원되는 SCM에 액세스할 수 있습니다.

Developer for System z에서는 자체 RAM 작성을 위해 사전 빌드된 여러 RAM 및 소스 코드 예제를 제공합니다.

IBM Rational Developer for System z Interface for CA Endevor® Software Configuration Manager는 Developer for System z 클라이언트에 CA Endevor® SCM에 대한 직접 액세스를 제공합니다.

(선택사항) SCLM 개발자 툴킷

SCLM 개발자 툴킷에서는 SCM의 기능을 클라이언트로 확장하는 데 필요한 도구를 제공합니다. SCLM 그 자체는 ISPF에 포함되는 호스트 기반 소스 코드 관리자입니다.

SCLM 개발자 툴킷에는 SCLM과 인터페이스하고 유산 코드 개발을 위한 전체 SCL 프로세스 액세스 및 메일프레임의 SCLM과 동기화된 워크스테이션에서 Java™ 및 J2EE 개발 전체에 대한 지원을 제공하는 Eclipse 기반 플러그인이 있습니다. 동기화 활동에는 메인프레임에서 J2EE 코드의 빌드, 어셈블링 및 배포가 포함됩니다.

(선택사항) 애플리케이션 배치 관리자(더 이상 사용되지 않음)

Developer for System z는 애플리케이션 배치 관리자의 특정 기능을 다양한 컴포넌트의 공통 배치 방법으로 사용합니다. 선택적 사용자 정의를 통해 더 많은 애플리케이션 배치 관리자 기능을 사용할 수 있으며 Developer for System z에 다음과 같은 서비스를 추가할 수 있습니다.

- IBM CICS Explorer®에서는 CICS 자원을 보고 관리하기 위한 Eclipse 기반 인프라를 제공하고, CICS 도구 사이의 좀 더 큰 통합을 가능하게 합니다.
- CRD(CICS Resource Definition) 클라이언트 및 서버는 다음 기능을 제공합니다.
 - CRD(CICS Resource Definition) 편집기

- 제한되고 제어되고 안전한 방식에서 애플리케이션 개발자가 정의하는 CICS 자원
- 파일 정의에 있는 실제 데이터 세트 이름 속성에 대한 CICS 관리자 제어를 제공함으로써 권한이 없거나 올바르지 않은 VSAM 데이터 세트로의 CICS 개발 액세스 방지
- 기타 CICS 개발 보조장치
- 다양한 CICS 웹 서비스 개발 보조장치

(선택사항) 호스트 기반 코드 분석

Developer for System z 클라이언트와 유사하게, Developer for System z 호스트는 별도의 제품으로 제공하는 코드 분석 도구인 Rational Developer for System z 호스트 유틸리티의 실행을 지원합니다. 호스트에서 코드 분석하면 매일 일괄처리 작업에 통합할 수 있다는 이점이 있습니다.

다음 코드 분석 도구는 호스트에서 사용 가능합니다.

- 코드 검토: 서로 다른 심각도의 규칙을 사용하여 코드 검토는 소스 코드를 스캔하고 규칙 위반을 보고합니다.
- 코드 적용: 실행 프로그램을 분석하고 실행되는 행을 실행 가능한 행의 총 수와 비교하여 보고서를 생성합니다.

(선택사항) 기타 사용자 정의 태스크

이 절에서는 다양한 선택적 사용자 정의 태스크를 결합합니다. 필요한 서비스를 구성하려면 해당 절의 지시사항을 수행하십시오.

Developer for System z 구성 파일에 대한 사용자 정의:

- pushtoclient.properties, 호스트 기반 클라이언트 제어
- ssl.properties, RSE SSL 암호화
- rsecomm.properties, RSE 추적
- include.conf, C/C++ 콘텐츠 지원을 위해 강제로 포함됨

기타 제품에 대한 Developer for System z 관련 사용자 정의:

- DB2 스토어드 프로시저
- z/OS UNIX 서브프로젝트
- 프리프로세서 지원 포함
- Enterprise COBOL 및 PL/I에 대한 xUnit 지원
- 엔터프라이즈 서비스 도구 지원
- CICS 양방향 언어 지원
- 생성된 코드에 대한 진단 IRZ 메시지
- 통합 디버거

- 문제점 판별 도구 지원
- DB2 및 IMS™ 디버그 지원
- 파일 관리자 지원
- WORKAREA 및 /tmp 정리

설치 검증

제품 사용자 정의를 완료한 후, 이 장에 기술된 설치 검증 프로그램(IVP)을 이용하여 핵심 제품 컴포넌트 설정 여부를 검증할 수 있습니다.

보안 정의

이 절에서는 샘플 RACF® 명령에서 필수 보안 정의와 선택적 보안 정의에 대해 설명합니다.

마이그레이션 안내서

이 절에서는 제품의 이전 릴리스와 비교하여 설치 및 구성 변경사항을 강조표시합니다. 이 릴리스로 마이그레이션하기 위한 몇 가지 가이드라인도 제공합니다.

운영자 명령

이 절에서는 Developer for System z에서 사용 가능한 운영자(또는 콘솔) 명령에 대한 개요를 제공합니다.

호스트 구성 참조서

이 절에서는 System z용 IBM Rational Developer 호스트 구성 참조서(SA30-4501)에 있는 정보를 요약합니다.

IBM Rational Developer for System z 호스트 구성 안내서

제 1 장 계획

이 *IBM Rational Developer for System z* 전제조건(SA30-3650) 장에 있는 정보를 사용하여 Developer for System z 설치 및 배치를 계획하십시오. 다음 주제에 대해 설명합니다.

- 『마이그레이션 고려사항』
- 『계획 고려사항』
- 5 페이지의 『설치 전 고려사항』
- 10 페이지의 『사전 구성 고려사항』
- 13 페이지의 『사전 배치 고려사항』
- 14 페이지의 『클라이언트 체크리스트』

마이그레이션 고려사항

189 페이지의 제 10 장 『마이그레이션 안내서』에서는 제품의 이전 릴리스와 비교하여 설치 및 구성 변경사항을 설명합니다. 이 정보를 사용하여 현재 Developer for System z 릴리스로의 마이그레이션을 계획하십시오.

참고:

- IBM Rational Developer for System z, IBM WebSphere Developer for System z, zSeries용 IBM WebSphere Developer 또는 IBM WebSphere Studio Enterprise Developer의 이전 사용자인 경우, IBM Rational Developer for System z 버전 9.0 을 설치하기 전에 관련된 사용자 정의 파일을 저장하십시오. 사용자 정의가 필요한 파일 개요는 189 페이지의 제 10 장 『마이그레이션 안내서』의 내용을 참조하십시오.
- Developer for System z의 여러 인스턴스를 실행하려는 경우, *호스트 구성 참조서* (SA30-4501)에 있는 "여러 인스턴스 실행"의 내용을 참조하십시오.

계획 고려사항

제품 개요

Developer for System z는 사용자의 개인용 컴퓨터에 설치된 클라이언트와 하나 이상의 *호스트* 시스템에 설치된 서버로 구성됩니다. 이 문서에는 z/OS 호스트 시스템에 대한 정보가 포함됩니다. 그러나, System z의 AIX® 및 Linux와 같은 다른 운영 체제도 지원됩니다.

클라이언트는 호스트에 대한 단일 양식 그래픽 인터페이스를 이용하고, 여럿 가운데서도 호스트에서 클라이언트로 작업을 오프로드하여 호스트에 자원을 저장하는 Eclipse 기반 개발 환경을 개발자에게 제공합니다.

호스트 부분은 특별히 시작되는 태스크 및 영구적으로 활성화되는 여러 태스크로 구성됩니다. 이러한 태스크를 사용하면 클라이언트가 MVS 데이터 세트, TSO 명령, z/OS UNIX 파일과 명령, 작업 제출 및 작업 출력과 같은 z/OS 호스트 시스템의 다양한 컴포넌트에 대해 작업할 수 있습니다.

Developer for System z가 이를 수행하도록 구성되고 이러한 공동 필수 요소 제품이 사용 가능한 경우 Developer for System z는 CICS, IBM 파일 관리자 및 SCM(Software Configuration Manager)과 같은 호스트 시스템에서 서브시스템 및 기타 애플리케이션 소프트웨어와 상호작용할 수 있습니다.

Developer for System z 디자인에 대한 기초 이해를 위해 호스트 구성 참조서(SA30-4501)에 있는 "Developer for system z 이해"의 내용을 참조하십시오.

Developer for System z에서 제공하는 기능에 대한 자세한 정보는 Developer for System z 웹 사이트(<http://www-03.ibm.com/software/products/us/en/developerforsystemz/>)를 참조하거나 로컬 IBM 담당자에게 문의하십시오.

스킬 요구사항

SMP/E 스킬은 Developer for System z 호스트 설치에 필요합니다.

Developer for System z 구성에서는 일반 시스템 프로그래밍 권한 및 전문 지식 이상이 필요하므로 기타 지원이 필요할 수도 있습니다. 7 페이지의 표 3 및 8 페이지의 표 4에서는 필수 및 선택적 사용자 정의 태스크에 필요한 관리자를 나열합니다.

시간 요구조건

Developer for System z 호스트 시스템 컴포넌트를 설치 및 구성하는 데 필요한 시간 양은 다음과 같은 여러 요인에 따라 달라집니다.

- 현재 z/OS UNIX 및 TCP/IP 구성
- 필수 소프트웨어 및 유지보수의 가용성
- OMVS 세그먼트가 Developer for System z 사용자에게 대해 정의되었는지 여부
- 클라이언트를 설치한 사용자가 설치를 테스트하고 발생할 수 있는 문제점을 보고할 수 있는지 여부

경험상 Developer for System z 호스트 시스템의 설치 및 구성 프로세스를 완료하려면 1일 - 4일이 필요합니다. 이 시간 요구사항은 경험이 풍부한 시스템 프로그래머가 아무 문제 없이 설치를 수행한 경우입니다. 문제점이 발생하거나 필요한 스킬을 사용할 수 없으면 설정에 더 많은 시간이 소요됩니다.

설치 전 고려사항

제품의 SMP/E 설치에 대한 상세 지시사항은 *Program Directory for IBM Rational Developer for System z*(GI11-8298)를 참조하십시오.

Developer for System z의 다중 인스턴스를 실행하려면, *호스트 구성 참조서* (SA30-4501)에서 "여러 인스턴스 실행"의 내용을 참조하십시오.

SETUID 권한 비트(시스템 기본값임)를 사용하여 Developer for System z가 설치된 파일 시스템(HFS 또는 zFS)을 마운트해야 합니다. 파일 시스템을 NOSETUID 매개변수와 마운트하면 Developer for System z가 사용자의 보안 환경을 작성하지 않고, 클라이언트의 연결 요청을 거부합니다. Java 및 z/OS UNIX 바이너리를 호스트하는 파일 시스템에 대해서는 모두 해당합니다.

설치 사용자 ID

Developer for System z를 설치하거나 유지보수를 설치하는 데 사용되는 사용자 ID에는 다음 속성이 반드시 있어야 합니다.

- TSO 액세스 권한(일반 리전 크기)
- 사용자 ID 및 기본 그룹 모두의 경우 보안 시스템에 대해 정의된 OMVS 세그먼트(예: RACF)
 - HOME 필드는 사용자에게 할당된 홈 디렉토리를 말합니다(읽기, 쓰기 및 실행 액세스 권한 있음).
 - OMVS 세그먼트의 PROGRAM 필드는 /bin/tcsh와 같은, /bin/sh 또는 기타 유효한 z/OS UNIX 셸이어야 합니다.
 - 사용자 ID의 기본 그룹에서는 GID가 필요합니다.
- FACILITY 클래스에서 BPX.SUPERUSER 프로파일에 대한 UID=0 또는 읽기 권한
- BPX.FILEATTR.APF 또는 BPX.FILEATTR.PROGCTL 프로파일이 FACILITY 클래스에서 정의되면, 이 프로파일에 대한 읽기 액세스 권한
- /tmp 디렉토리(또는 TMPDIR 환경 변수에서 참조되는 디렉토리)에 대한 읽기, 쓰기 및 실행 액세스 권한

필수 제품

IBM Rational Developer for System z 전제조건(SA30-3650)에는 Developer for System z가 동작하기 전에 설치 및 운영 중이어야 하는 필수 소프트웨어 목록이 있습니다. 또한 Developer for System z의 특정 기능을 지원하기 위한 상호 필수 소프트웨어 목록도 있습니다. 이러한 필수 소프트웨어를 설치하고 런타임에 실행해야만 연관되는 기능이 설계한 대로 작동할 수 있습니다.

이 필수 제품을 미리 사용할 수 있도록 계획하십시오. 사용자 사이트의 정책에 따라 약간의 시간이 소요될 수 있습니다. 기본 설치를 위한 주요 필수 소프트웨어는 다음과 같습니다.

- z/OS 1.8 이상
- ISPF APAR OA43014(TSO/ISPF Client Gateway)
- Java 6.0 이상(31 또는 64비트)

필수 자원

Developer for System z에는 표 1에 표시된 시스템 자원 할당이 필요합니다. 표 2에 나열되는 자원은 선택적 서비스에 필요합니다. 사용자 사이트의 정책에 따라 소프트웨어를 얻는 데 약간의 시간이 소요될 수 있으므로 이러한 자원은 사용 가능하도록 계획하십시오.

표 1. 필수 자원

자원	기본값	정보
APF 권한 데이터 세트	FEK.SFEKAUTH	23 페이지의 『PROGxx에 APF 권한 부여』
시작 태스크	JMON 및 RSED	27 페이지의 『PROCLIB 변경사항』
호스트 한정 사용을 위한 포트 (JMON)	6715	35 페이지의 『FEJCNFG, JES 작업 모니터 구성 파일』
클라이언트-호스트 통신용 포트 (RSED)	4035	41 페이지의 『rsed.envvars, RSE 구성 파일』
클라이언트-호스트 통신용 포트 범위(RSED)	사용 가능한 모든 포트가 사용됨	53 페이지의 『RSE 서버에 사용 가능한 PORTRANGE 정의』
z/OS UNIX 서버 보안 정의	RSED 시작 태스크의 BPX.SERVER에 대한 UPDATE 권한	176 페이지의 『RSE를 보안 z/OS UNIX 서버로 정의』
PassTicket 보안 정의	기본값 없음	177 페이지의 『RSE에 대한 PassTicket 지원 정의』
MVS 빌드 프로시저	ELAXF*	27 페이지의 『PROCLIB 변경사항』

표 2. 선택적 자원

자원	기본값	정보
IPL(CLPA)	해당사항 없음	146 페이지의 『(선택사항) 통합 디버거』
시작 태스크	DBGMR	146 페이지의 『(선택사항) 통합 디버거』
LINKLIST 데이터 세트	FEK.SFEKAUTH 및 FEK.SFEKLOAD	<ul style="list-style-type: none"> • 111 페이지의 제 4 장 『(선택사항) SCLM 개발자 툴킷』 • 146 페이지의 『(선택사항) 통합 디버거』

표 2. 선택적 자원 (계속)

자원	기본값	정보
LPA 데이터 세트	FEK.SFEKLPA	69 페이지의 제 3 장 『(선택사항) CARMA(Common Access Repository Manager)』
호스트 한정 사용을 위한 포트 범위	사용 가능한 모든 포트가 사용됨	<ul style="list-style-type: none"> 69 페이지의 제 3 장 『(선택사항) CARMA(Common Access Repository Manager)』 127 페이지의 제 6 장 『(선택사항) 호스트 기반 코드 분석』
호스트 한정 사용을 위한 포트 범위	5336	146 페이지의 『(선택사항) 통합 디버거』
클라이언트-호스트 통신용 포트	<ul style="list-style-type: none"> 웹 서비스의 경우 5129, RESTful 서비스의 경우 5130 통합 디버거의 경우, 5335 	<ul style="list-style-type: none"> 119 페이지의 제 5 장 『(선택사항) 애플리케이션 배치 관리자 (더 이상 사용되지 않음)』 146 페이지의 『(선택사항) 통합 디버거』
CICS CSD 업데이트	다중 값	<ul style="list-style-type: none"> 119 페이지의 제 5 장 『(선택사항) 애플리케이션 배치 관리자 (더 이상 사용되지 않음)』 146 페이지의 『(선택사항) 통합 디버거』
CICS JCL 업데이트	<ul style="list-style-type: none"> FEK.SFEKLOAD FEK.SFEKAUTH 	<ul style="list-style-type: none"> 119 페이지의 제 5 장 『(선택사항) 애플리케이션 배치 관리자 (더 이상 사용되지 않음)』 144 페이지의 『(선택사항) CICS 양방향 언어 지원』 146 페이지의 『(선택사항) 통합 디버거』

Developer for System z 구성에서는 일반 시스템 프로그래밍 권한 및 전문 지식 이상이 필요하므로 기타 지원이 필요할 수 있습니다. 표 3 및 8 페이지의 표 4에서는 필수 및 선택적 사용자 정의 태스크에 필요한 관리자를 나열합니다.

표 3. 필수 태스크에 필요한 관리자

관리자	태스크	정보
시스템	모든 사용자 정의 태스크에 일반 시스템 프로그래머 조치가 필요함	적용되지 않음

표 3. 필수 태스크에 필요한 관리자 (계속)

관리자	태스크	정보
보안	<ul style="list-style-type: none"> • Developer for System z 사용자에게 대한 OMVS 세그먼트 정의 • 데이터 세트 프로파일 정의 • 시작 태스크 정의 • 운영자 명령 보안 정의 • z/OS UNIX 서버 프로파일 정의 • 애플리케이션 보안 정의 • PassTicket 지원 정의 • 프로그램으로 제어되는 데이터 세트 정의 • 프로그램으로 제어되는 z/OS UNIX 파일 정의 	호스트 구성 참조서 (SA30-4501)의 "보안 고려사항"
TCP/IP	새 TCP/IP 포트 정의	호스트 구성 참조서 (SA30-4501)의 "TCP/IP 고려사항"
WLM	서버 및 하위 프로세스에 시작 태스크 목표 지정	호스트 구성 참조서 (SA30-4501)의 "WLM 고려사항"

표 4. 선택적 태스크에 필요한 관리자

관리자	태스크	정보
시스템	모든 사용자 정의 태스크에 일반 시스템 프로그래머 조치가 필요함	적용되지 않음

표 4. 선택적 태스크에 필요한 관리자 (계속)

관리자	태스크	정보
보안	<ul style="list-style-type: none"> • 데이터 세트 프로파일 정의 • 프로그램으로 제어되는 데이터 세트 정의 • xxx* 작업을 제출하기 위한 권한 정의 • CICS 트랜잭션 보안 정의 • SSL을 위한 인증 추가 • X.509 클라이언트 인증 지원 정의 • 클라이언트로 푸시를 위한 그룹 및 프로파일 정의 • 클라이언트 기능 변경에 필요한 프로파일 정의 • 시작 태스크 정의 • z/OS UNIX 서버 프로파일 정의 • 권한 부여된 디버깅 프로파일 정의 	<ul style="list-style-type: none"> • 호스트 구성 참조서 (SA30-4501)의 "보안 고려사항" • 호스트 구성 참조서 (SA30-4501)의 "CICSTS 고려사항" • 호스트 구성 참조서 (SA30-4501)의 "SSL 및 X.509 인증 설정"
TCP/IP	새 TCP/IP 포트 정의	호스트 구성 참조서 (SA30-4501)의 "TCP/IP 포트"
SCLM	<ul style="list-style-type: none"> • Java EE 지원을 위한 SCLM 언어 변환기 정의 • Java EE 지원을 위한 SCLM 유형 정의 	111 페이지의 제 4 장 『(선택사항) SCLM 개발자 툴킷』
CICS TS	<ul style="list-style-type: none"> • CICS 리전 JCL 업데이트 • CICS 리전 CSD 업데이트 • CICS 그룹 정의 • CICS 트랜잭션 이름 정의 • CICS로 프로그램 정의 • CICS로 디버거 정의 	<ul style="list-style-type: none"> • 119 페이지의 제 5 장 『(선택사항) 애플리케이션 배치 관리자 (더 이상 사용되지 않음)』 • 144 페이지의 『(선택사항) CICS 양방향 언어 지원』 • 146 페이지의 『(선택사항) 통합 디버거』
WLM	Developer for System z 태스크에 목적 지정	호스트 구성 참조서 (SA30-4501)의 "WLM 고려사항"
LDAP	클라이언트로 푸시를 위한 그룹 정의	호스트 구성 참조서(SA30-4501)의 "클라이언트로 푸시 고려사항"

사전 구성 고려사항

Developer for System z 자체, 시스템과 상호작용하는 방법, 필수 소프트웨어 및 공동 필수 요소 제품 사용에 대한 자세한 정보는 호스트 구성 참조서 (SA30-4501)의 내용을 참조하십시오. 이 정보는 현재 사용자의 요구사항 및 향후 성장을 지원하는 설정 작성에 도움이 될 수 있습니다.

워크로드 관리

전통적인 z/OS 애플리케이션과 달리 Developer for System z는 워크로드 관리자(WLM)가 쉽게 식별할 수 있는 단일 애플리케이션이 아닙니다. Developer for System z는 클라이언트에게 호스트 시스템 서비스 및 데이터에 대한 액세스를 제공하기 위해 상호 작용하는 여러 컴포넌트로 구성됩니다. WLM 구성을 계획하려면 호스트 구성 참조서 (SA30-4501)에서 "WLM 고려사항"의 내용을 참조하십시오.

참고: Developer for System z는 서로 통신하는 여러 태스크 및 클라이언트로 구성됩니다. 이러한 태스크는 다양한 타이머를 사용하여 참여자와의 통신 유실을 발견합니다. Developer for System z에 대한 CPU 로드가 너무 많거나 워크로드 관리(WLM) 설정이 잘못된 시스템에서는 (제한시간 창에서 CPU 시간 부족으로 인해) 제한시간 문제가 발생할 수 있습니다.

자원 사용량 및 시스템 한계

Developer for System z에서는 주소 공간, z/OS UNIX 프로세스 및 스레드와 같은 여러 시스템 자원을 사용합니다. 이러한 자원의 사용 가능성은 여러 시스템 정의로 제한됩니다. 시스템 구성을 계획할 수 있도록 핵심 자원의 사용량을 측정하려면, 호스트 구성 참조서 (SA30-4501)에서 "튜닝 고려사항"의 내용을 참조하십시오. Developer for System z는 스토리지 자원 제한사항을 과감하게 변경하며 31 비트 또는 64 비트 모드 각각에서 실행될 수 있습니다.

필수 제품에 대해 요구되는 구성

MVS 시스템 프로그래머, 보안 관리자, TCP/IP 관리자에게 문의하여 필수 제품 및 소프트웨어가 설치, 테스트 및 작동하는지 여부를 확인합니다. 여기서는 간과할 수 있는 몇 가지 필수 사용자 정의 태스크에 대해 설명합니다.

- 모든 Developer for System z 사용자는 Java 디렉토리에 대해 읽기와 실행 액세스를 가지고 있어야 합니다.
- z/OS UNIX 서브프로젝트의 원격(호스트 기본) 조치에서는 z/OS UNIX 버전의 REXEC 또는 SSH가 호스트 시스템에서 활성화되어 있어야 합니다.

사용자 ID 고려사항

Developer for System z 사용자의 사용자 ID에는 다음 속성이 있어야 합니다.

- TSO 액세스 권한(일반 리전 크기)

참고: 많은 메모리가 필요한 기능(예: Java)이 실행되므로 설치 검증 프로그램(IVPO)을 실행하는 사용자 ID의 경우 대용량의 리전이 필요합니다. 리전 크기를 131072 킬로바이트(128 메가바이트) 이상으로 설정해야 합니다.

- 사용자 ID 및 기본 그룹 모두의 경우 보안 시스템에 대해 정의된 OMVS 세그먼트(예: RACF)
 - HOME 필드는 사용자에게 할당된 홈 디렉토리를 말합니다(읽기, 쓰기 및 실행 액세스 권한 있음).
 - OMVS 세그먼트의 PROGRAM 필드는 /bin/tcsh와 같은, /bin/sh 또는 기타 유효한 z/OS UNIX 셸이어야 합니다.
 - 시스템 기본값이 사용되도록 ASSIZEMAX 필드는 설정하지 말아야 합니다.
 - 사용자 ID에는 UID 0이 필요하지 않습니다.

예제(명령 **LISTUSER userid NORACF OMVS**):

USER=userid

OMVS INFORMATION

UID= 0000003200
HOME= /u/userid
PROGRAM= /bin/sh
CPUTIMEMAX= NONE
ASSIZEMAX= NONE
FILEPROCMA= NONE
PROCUSERMAX= NONE
THREADSMAX= NONE
MMAPAREAMAX= NONE

- 사용자 ID의 기본 그룹에서는 GID가 필요합니다.

예제(명령 **LISTGRP group NORACF OMVS**):

GROUP group

OMVS INFORMATION

GID= 0000003243

- Developer for System z 설치 및 구성 디렉토리와 파일(기본값은 /usr/lpp/rdz/*, /etc/rdz/* 및 /var/rdz/*)에 대한 읽기 및 실행 액세스 권한
- Developer for System z WORKAREA 디렉토리(기본값 /var/rdz/WORKAREA) 및 사용자 로그 디렉토리(기본값 /var/rdz/logs)에 대한 읽기, 쓰기 및 실행 액세스 권한
- Developer for System z 설치 데이터 세트(기본값 FEK.SFEK*)에 대한 읽기 액세스 권한
- /tmp 디렉토리 또는 TMPDIR 환경 변수에서 참조되는 디렉토리에 대한 읽기, 쓰기 및 실행 액세스 권한

서버 고려사항

Developer for System z는 영구적으로 활성화된 여러 서버로 구성되며, 이는 시작 태스크이거나 사용자 작업이 될 수 있습니다. 이 서버는 요청된 서비스 자체를 제공하거나 기타 서버(z/OS UNIX 스레드 또는 사용자 작업으로서)를 시작하여 서비스를 제공합니다. 특별한 시작 순서는 없습니다. 유일한 요구사항은 첫 번째 사용자가 연결을 시도하기 전에 서버를 구동시켜 실행하는 것입니다. Developer for System z 서버와 서비스가 사용하는 보안 메커니즘은 데이터 세트 및 보안 상태로 상주하는 파일 시스템에 따라 다릅니다. 이는 신뢰할 수 있는 시스템 관리자만 프로그램 라이브러리와 구성 파일을 업데이트할 수 있어야 함을 의미합니다.

- 디버그 관리자(DBGMGR)는 디버그 관련 서비스를 제공합니다.
- JES 작업 모니터(JMON)는 모든 JES 관련 서비스를 제공합니다.
- RSE(Remote Systems Explorer)는 클라이언트를 호스트 시스템에 연결하고 특정 서비스를 위해 기타 서버를 시작하는 등 핵심 서비스를 제공합니다. RSE는 다음 두 개의 논리 엔티티로 구성됩니다.
 - RSE 디먼(RSED) - 연결 설정을 관리하며, 단일 서버 모드에서 실행할 책임이 있습니다.
 - RSE 서버 - 각 클라이언트 요청을 처리합니다.

호스트 구성 참조서 (SA30-4501)의 "TCP/IP 포트"에서 설명한 대로, 클라이언트가 연결하려면 특정 호스트 시스템 서비스 및 이들의 포트를 사용할 수 있어야 하며, 호스트 시스템을 보호하는 방화벽에 정의되어 있어야 합니다. Developer for System z에서 사용하는 기타 모든 포트는 호스트와의 트래픽만 있습니다. 다음은 기본 Developer for System z 설정에서 외부 통신에 필요한 포트를 나열한 것입니다.

- 클라이언트-호스트 통신 설정을 위한 RSE 디먼(TCP 사용), 기본 포트는 4035.
- 클라이언트-호스트 통신을 위한 RSE 서버(TCP 사용). 기본적으로 사용 가능한 포트가 사용되지만, 사용 가능한 포트는 지정된 범위로 제한될 수 있습니다.

구성 방법

Developer for System z에서는 제품의 호스트 시스템측을 구성하는 대체 방법을 제공합니다. 대체 방법은 다음과 같습니다.

- 호스트 구성 유틸리티 사용. 이 ISPF 패널 애플리케이션은 필수 사용자 정의 단계 및 선택된 선택적 사용자 정의 단계를 안내합니다. 자세한 정보는 호스트 구성 유틸리티 안내서(SA30-4582)를 참조하십시오.
- 호스트 구성 빠른 시작 안내서 사용. 이 문서는 필수 사용자 정의 단계를 안내합니다. 이 안내서의 범위는 기본 설치로 제한됩니다.
- 호스트 구성 안내서 사용. 이 문서는 필수 사용자 정의 단계 및 모든 선택적 사용자 정의 단계를 안내합니다. 이 안내서에서는 기본이 아닌 일부 시나리오를 비롯하여 구성 가능한 모든 옵션을 다룹니다.

사전 배치 고려사항

Developer for System z에서는 다른 시스템으로 설치 복사를 지원하므로, 각 시스템에 SMP/E를 설치하지 않아도 됩니다.

다음 데이터 세트, 디렉토리 및 파일은 기타 시스템으로의 배치에 필수사항입니다. 다른 위치로 파일을 복사한 경우, 이 파일은 다음 목록에서 대응부를 대체해야 합니다.

참고: 다음 목록은 필수 소프트웨어 및 공동 필수 소프트웨어의 배치 요구사항에 대해 다루지 않습니다.

Developer for System z

- FEK.SFEKAUTH(*)
- FEK.SFEKLOAD(*)
- FEK.SFEKPROC(*)
- FEK.#CUST.PARMLIB(*)
- FEK.#CUST.PROCLIB(*)
- /usr/lpp/rdz/*
- /etc/rdz/*
- /var/rdz/(디렉토리 구조만 해당)
- optional parts:
 - FEK.SFEKLPA(*)
 - FEK.SFEKLMOD(*)
 - FEK.#CUST.CNTL(*)
 - 정의, 데이터 세트, 파일 및 FEK.#CUST.JCL에서 사용자 정의 작업으로 발생한 디렉토리

Developer for System z 호스트 유틸리티

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

참고:

- FEK 및 /usr/lpp/rdz는 Developer for System z 설치 중에 사용되는 상위 레벨 규정자와 경로입니다. FEK.#CUST, /etc/rdz 및 /var/rdz는 제품 사용자 정의 도중 사용되는 기본 위치입니다(자세한 정보는 18 페이지의 『사용자 정의 설치』 참조).
- AKG 및 /usr/lpp/rdzutil은 Developer for System z 호스트 유틸리티 설치 중에 사용되는 상위 레벨 규정자와 경로입니다.
- 제품의 z/OS UNIX 파트를 용이하게 배치하려면 개인 파일 시스템(HFS 또는 zFS)에 Developer for System z를 설치해야 합니다.

- 개인 파일 시스템을 사용할 수 없는 경우, z/OS UNIX tar 명령과 같은 아카이브 도구를 사용하여 한 시스템에서 다른 시스템으로 z/OS UNIX 디렉토리를 전송하십시오. 이 메소드는 Developer for System z 파일 및 디렉토리에 대한 속성(예: 프로그램 제어)을 유지하기 위한 것입니다.

Developer for System z 설치 디렉토리를 아카이브 및 복원하는 다음 샘플 명령에 대한 자세한 정보는 *UNIX System Services Command Reference(SA22-7802)*를 참조하십시오.

- 아카이브: `cd /SYS1/usr/lpp/rdz; tar -cSf /u/userid/rdz.tar`
- 복원: `cd /SYS2/usr/lpp/rdz; tar -xSf /u/userid/rdz.tar`

클라이언트 체크리스트

클라이언트가 올바르게 작업하려면 Developer for System z 클라이언트 사용자가 TCP/IP 포트 번호와 같은 특정 호스트 시스템 사용자 정의 결과를 알고 있어야 합니다. 이 체크리스트를 사용하여 필요한 정보를 수집하십시오.

표 5의 체크리스트는 필수 사용자 정의 단계의 필요한 결과를 표시합니다. 표 6에서는 선택적 사용자 정의 단계의 필수 결과를 표시합니다.

표 5. 클라이언트 체크리스트: 필수 부분

사용자 정의	값
RSE 디먼 TCP/IP 포트 번호입니다. 기본값은 4035입니다.	
29 페이지의 『RSED, RSE 디먼 시작 태스크』의 내용을 참조하십시오.	

표 6. 클라이언트 체크리스트: 선택적 부분

사용자 정의	값
ELAXF* 프로시저가 시스템 프로시저 라이브러리에 없는 경우 이들의 위치입니다. 기본값은 FEK.#CUST.PROCLIB입니다.	
33 페이지의 『ELAXF* 원격 빌드 프로시저』에 있는 JCLLIB 노트를 참조하십시오.	
ELAXF* 프로시저 이름이 변경된 경우 프로시저 또는 단계 이름입니다.	
33 페이지의 『ELAXF* 원격 빌드 프로시저』에서 변경에 대한 노트를 참조하십시오.	
시스템 프로시저 라이브러리에 없는 경우 AKGCR 프로시저의 위치입니다. 기본값은 AKG.#CUST.PROCLIB입니다.	
127 페이지의 『코드 검토』에 있는 JCLLIB 노트를 참조하십시오.	
시스템 프로시저 라이브러리에 없는 경우 AKGCC 프로시저의 위치입니다. 기본값은 AKG.#CUST.PROCLIB입니다.	
128 페이지의 『코드 적용』에 있는 JCLLIB 노트를 참조하십시오.	

표 6. 클라이언트 체크리스트: 선택적 부분 (계속)

사용자 정의	값
FEKRNPLI 포함 프리프로세서 exec 문의 위치입니다. 기본값은 FEK.#CUST.CNTL입니다. 141 페이지의 『(선택사항) 프리프로세서 지원 포함』의 내용을 참조하십시오.	
LINKLIST에 없는 경우, 디버거 로드 모듈의 위치. 기본값은 FEK.SFEKAUTH입니다. 146 페이지의 『(선택사항) 통합 디버거』의 내용을 참조하십시오.	
rsed.envvars의 LINKLIST 또는 STEPLIB에 없는 경우, 단위 테스트 로드 모듈의 위치입니다. 기본값은 FEK.SFEKLOAD입니다. 142 페이지의 『(선택사항) Enterprise COBOL 및 PL/I용 xUnit 지원』의 내용을 참조하십시오.	
시스템 프로시저 라이브러리에 없는 경우 AZUZUNIT 프로시저의 위치입니다. 기본값은 FEK.#CUST.PROCLIB입니다. 142 페이지의 『(선택사항) Enterprise COBOL 및 PL/I용 xUnit 지원』에 있는 JCLLIB 노트를 참조하십시오.	
단위 테스트 출력 형식 지정에 사용되는 샘플 *.xsd 및 *.xsl XML 파일의 위치입니다. 기본값은 /usr/lpp/rdz/samples/zunit/xsd 및 /usr/lpp/rdz/samples/zunit/xsl입니다. 142 페이지의 『(선택사항) Enterprise COBOL 및 PL/I용 xUnit 지원』의 내용을 참조하십시오.	
(공동 필수 요소) 호스트 연결 에뮬레이터의 TN3270 포트 번호입니다. 기본값은 23입니다. 호스트 구성 참조서 (SA30-4501)에 있는 "TCP/IP 포트"의 내용을 참조하십시오.	
(공동 필수 요소) REXEC 또는 SSH 포트 번호로서, 기본값은 512 또는 22입니다. 140 페이지의 『(선택사항) z/OS UNIX 서브프로젝트』의 내용을 참조하십시오.	
(공동 필수 요소) 디버거 도구 서버 포트 번호(기본값 없음)입니다. 150 페이지의 『(선택사항) DB2 및 IMS 디버거 지원』의 내용을 참조하십시오.	
애플리케이션 배치 관리자 포트 번호로서, 웹 서비스의 경우 기본값은 5129이고, REST 서비스의 경우 기본값은 5130입니다. 호스트 구성 참조서(SA30-4501)의 "TCP/IP 포트"의 내용을 참조하십시오.	
CARMA RAM 샘플의 SFEKSAMP 샘플 라이브러리 위치입니다. 기본값은 FEK.SFEKSAMP입니다. CARMA Developer's Guide(SC23-7660)를 참조하십시오.	
CARMA SCLM RAM 데이터 세트 할당을 위한 CRA#ASLM JCL의 위치입니다. 기본값은 FEK.#CUST.JCL입니다. 97 페이지의 『SCLM RAM』에 있는 CRA#ASLM 노트를 참조하십시오.	

제 2 장 기본 사용자 정의

다음 사용자 정의 단계는 기본 Developer for System z 설정을 위한 단계입니다. 사용자 정의 요구사항에 대한 선택적 컴포넌트 장을 참조하십시오.

요구사항 및 체크리스트

이 사용자 정의 작업을 완료하려면 보안 관리자 및 TCP/IP 관리자의 도움이 필요하며, 다음 자원 또는 특수 사용자 정의 작업이 필요합니다.

- APF 권한 데이터 세트
- 다양한 PARMLIB 업데이트
- 다양한 보안 소프트웨어 업데이트
- 내부 및 클라이언트-호스트 통신을 위한 여러 TCP/IP 포트
- 선택적 SVC를 활성화하기 위한 IPL

사용자 사이트에서 설치를 확인하고 Developer for System z 사용을 시작하려면 다음 작업을 수행하십시오. 특별히 언급되지 않는 한 모든 작업이 필수입니다.

1. 사용자 정의할 수 있는 샘플 사본을 작성한 후 Developer for System z의 작업 환경을 작성하십시오. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오.
2. z/OS UNIX 시스템 한계를 업데이트하고, 시작된 작업을 시작하고, APF 권한 및 LINKLIST 데이터 세트를 정의한 후 선택적으로 SVC 및 LPA 데이터 세트를 정의하십시오. 자세한 정보는 19 페이지의 『PARMLIB 변경사항』의 내용을 참조하십시오.
3. 시작 작업 프로시저를 작성하고 프로시저를 컴파일 및 링크하십시오. 자세한 정보는 27 페이지의 『PROCLIB 변경사항』의 내용을 참조하십시오.
4. 보안 정의를 업데이트하십시오. 자세한 정보는 34 페이지의 『보안 정의』의 내용을 참조하십시오. 스프레드 보안을 설정하려면 PassTicket 사용법을 이해해야 합니다. 호스트 구성 참조서 (SA30-4501)에 있는 "PassTicket 사용"의 내용을 참조하십시오.
5. Developer for System z에 대한 Developer 구성 파일을 사용자 정의하십시오. 세부사항은 다음을 참조하십시오.
 - 35 페이지의 『FEJCNFG, JES 작업 모니터 구성 파일』
 - 41 페이지의 『rsed.envvars, RSE 구성 파일』
 - 64 페이지의 『ISPF.conf, ISPF의 TSO/ISPF Client Gateway 구성 파일』

사용자 정의 설치

Developer for System z에는 여러 개의 샘플 구성 파일 및 샘플 JCL이 포함되어 있습니다. 유지보수 적용 시 사용자 정의를 겹쳐쓰지 않도록 하기 위해, 이 멤버 모두와 z/OS UNIX 파일을 다른 위치로 복사하고 사본을 사용자 정의하십시오.

Developer for System z의 일부 기능에서는 z/OS UNIX에 특정 디렉토리가 존재해야 하며, 이 디렉토리는 제품 사용자 정의 도중 작성되어야 합니다. 설치를 쉽게 하기 위해, 사본 및 필수 디렉토리를 작성하는 샘플 작업 FEKSETUP이 제공됩니다.

참고: *Rational Developer for System z 호스트 구성 유틸리티 안내서*(SA30-4582)에서는 호스트 구성 유틸리티를 사용할 경우 호스트 시스템 구성에 대해 설명합니다. FEKSETUP 작업과 유틸리티는 해당 태스크가 이미 수행되었는지 확인하지 않고 동일한 태스크의 일부를 수행합니다. 그러므로 이미 구해진 변경사항을 실행 취소할 수 있습니다. 그러므로 단일 설치에서 두 메소드 모두를 사용해서는 안됩니다.

구성 파일과 구성 JCL에 대해 사용자 정의 할 수 있는 사본을 작성하고 필요한 z/OS UNIX 디렉토리를 작성하려면, FEK.SFEKSAMP 데이터 세트에서 샘플 FEKSETUP 멤버를 사용자 정의하고 제출하십시오. 필요한 사용자 정의 단계는 멤버 안에 설명되어 있습니다.

이 작업은 다음 태스크를 수행합니다.

- FEK.#CUST.PARMLIB를 작성하여 샘플 구성 파일로 채웁니다.
- FEK.#CUST.PROCLIB를 작성하여 샘플 SYS1.PROCLIB 멤버로 채웁니다.
- FEK.#CUST.JCL을 작성하여 샘플 구성 JCL로 채웁니다.
- FEK.#CUST.CNTL을 작성하여 샘플 서버 시작 스크립트로 채웁니다.
- FEK.#CUST.ASM을 작성하여 샘플 어셈블러 소스 코드로 채웁니다.
- FEK.#CUST.COBOLE를 작성하여 샘플 COBOL 소스 코드로 채웁니다.
- FEK.#CUST.SQL을 작성하여 샘플 SQL 명령 파일로 채웁니다.
- /etc/rdz/*를 작성하여 샘플 구성 파일로 채웁니다.
- 다양한 Developer for System z 기능의 작업 디렉토리로 /var/rdz/*를 작성하여 샘플 파일로 채웁니다.

참고:

- 특별히 표시하지 않으면 본 간행물의 구성 단계에서는 FEKSETUP 작업에서 작성된 멤버와 파일 위치를 사용합니다. 업데이트해서는 안되는 원래 샘플은 FEK.SFEKSAMP 및 /usr/lpp/rdz/samples/에 있습니다.
- 어느 샘플 멤버를 어느 데이터 세트로 복사해야 하는지에 대한 자세한 정보, 작성되는 디렉토리, 권한 비트마스크 및 여러 샘플 파일이 복사되는 위치에 대한 자세한 정보는, FEK.SFEKSAMP(FEKSETUP)의 주석을 참조하십시오.

- 기존 설정의 마이그레이션에 도움을 주기 위해 FEK.SFEKSAMP(FEKSETUP)의 주석에서도 Developer for System z 버전 사이의 변경에 대해 설명합니다.
- 동일한 파일 시스템(HFS 또는 zFS)에 Developer for System z z/OS UNIX 파일 모두를 보관하지만 구성 파일은 /etc/rdz에 배치하려는 경우, 기호 링크를 사용하여 이 문제점을 해결할 수 있습니다. 다음 샘플 z/OS UNIX 명령은 기존 파일 시스템(/usr/lpp/rdz/cust)에서 새로운 디렉토리를 작성하고, 이에 대한 기호 링크(/etc/rdz)를 정의합니다.

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

PARMLIB 변경사항

이 절에서는 다음 PARMLIB 변경사항에 대해 설명합니다.

- 『BPXPRMxx에서 z/OS UNIX 한계 설정』
- 21 페이지의 『COMMNDxx에 시작 태스크 추가』
- 22 페이지의 『IEASVCxx의 SVC 정의』
- 22 페이지의 『LPALSTxx의 LPA 정의』
- 23 페이지의 『PROGxx에 APF 권한 부여』
- 24 페이지의 『PROGxx의 LINKLIST 정의』
- 25 페이지의 『필수 LINKLIST 및 LPA 정의』
- 27 페이지의 『기타 제품에 대한 LINKLIST 정의』

다음 섹션에서 나열한 PARMLIB 정의에 대한 자세한 정보는 *MVS 초기화 및 튜닝 참조서(SA22-7592)*를 참조하십시오. 샘플 콘솔 명령에 대한 자세한 정보는 *MVS 시스템 명령(SA22-7627)*을 참조하십시오.

BPXPRMxx에서 z/OS UNIX 한계 설정

호스트 시스템에 클라이언트를 연결한 것과 같이 핵심 서비스를 제공하는 원격 시스템 탐색기(RSE)는 z/OS UNIX 기반 프로세스입니다. 그러므로, 동시에 활성화되는 Developer for System z 사용자와 그들의 평균 워크로드 수에 따라 BPXPRMxx에서 z/OS UNIX 시스템 한계에 대해 올바른 값을 설정하는 것이 중요합니다. IPL 동안 사용하여 하는 BPXPRMxx parmlib 멤버를 지정하려면 IEASYSxx parmlib 멤버에서 OMVS=xx를 정의하십시오.

다른 BPXPRMxx 정의 한계와 Developer for System z에 미치는 이들의 영향에 대한 자세한 정보는 *호스트 구성 참조서 (SA30-4501)*에서 "튜닝 고려사항"의 내용을 참조하십시오.

MAXASSIZE는 최대 주소 공간(프로세스) 리전 크기를 지정합니다. SYS1.PARMLIB (BPXPRMxx)의 MAXASSIZE를 2G로 설정하십시오. 이것이 허용되는 최대값입니다. 이는

시스템 전체 한계이므로 모든 z/OS UNIX 주소 공간에 대해 활성화됩니다. 이것이 사용자가 원하는 내용이 아니면, 174 페이지의 『Developer for System z 시작 태스크 정의』에 설명한 대로 보안 소프트웨어에서 Developer for System z에 대해서만 한계를 설정할 수 있습니다.

MAXTHREADS는 단일 프로세스의 최대 활성 스레드 수를 지정합니다. SYS1.PARMLIB(BPXPRMxx)의 MAXTHREADS를 1500 이상으로 설정하십시오. 이는 시스템 전체 한계이므로 모든 z/OS UNIX 주소 공간에 대해 활성화됩니다. 이것이 사용자가 원하는 내용이 아니면, 174 페이지의 『Developer for System z 시작 태스크 정의』에 설명한 대로 보안 소프트웨어에서 Developer for System z에 대해서만 한계를 설정할 수 있습니다.

MAXTHREADTASKS는 단일 프로세스의 최대 활성 MVS 태스크 수를 지정합니다. SYS1.PARMLIB(BPXPRMxx)의 MAXTHREADTASKS를 1500 이상으로 설정하십시오. 이는 시스템 전체 한계이므로 모든 z/OS UNIX 주소 공간에 대해 활성화됩니다. 이것이 사용자가 원하는 내용이 아니면, 174 페이지의 『Developer for System z 시작 태스크 정의』에 설명한 대로 보안 소프트웨어에서 Developer for System z에 대해서만 한계를 설정할 수 있습니다.

MAXPROCUSER는 단일 z/OS UNIX 사용자 ID가 동시에 활성화할 수 있는 최대 프로세스 수를 지정합니다. SYS1.PARMLIB(BPXPRMxx)의 MAXPROCUSER를 50 이상으로 설정하십시오. Developer for System z를 사용하는 각각 클라이언트에 대하여 활성화되어야 하므로 이 설정은 시스템 범용 한계로 사용됩니다.

이러한 값은 다음과 같은 콘솔 명령으로 동적으로 검사하고 설정할 수 있습니다(다음 IPL까지).

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

참고:

- 주소 공간 크기가 설정되거나 제한되는 기타 위치에 대한 자세한 정보는 호스트 구성 참조서 (SA30-4501)에서 "주소 공간 크기"의 내용을 참조하십시오.
- 여기에서 제안하는 MAXPROCUSER 값은 고유한 z/OS UNIX 사용자 ID(UID)를 가지고 있는 사용자에게 기초를 둡니다. 사용자가 똑같은 UID를 공유하면 이 값을 늘리십시오.

- MAXPROCSYS 및 MAXUIDS 값과 같이 기타 BPXPRMxx 값이 동시에 활성화되는 예상 Developer for System z 사용자 수를 처리하기에 충분한지 확인하십시오. 자세한 내용은 호스트 구성 참조서 (SA30-4501)의 "튜닝 고려사항"의 내용을 참조하십시오.
- Developer for System z의 SMP/E 설치 도중, 코드를 별도의 파일 시스템(HFS의 zFS)에 두고 BPXPRMxx를 업데이트하여 시스템 IPL 도중 이 파일 시스템을 마운트하는 것이 좋습니다. 이 업데이트를 계속 수행해야 하는 경우 샘플 마운트 명령 반복이 포함됩니다.

```

MOUNT FILESYSTEM('#dsn')
  MOUNTPPOINT('-PathPrefix-usr/lpp/rdz')
  MODE(RDWR) /* can be MODE(READ) */
  TYPE(ZFS) PARM('AGGRGROW') /* zFS, with extents */
/* TYPE(HFS) */ /* HFS, auto. extent */

```

- Developer for System z 호스트 유틸리티의 SMP/E 설치 도중, 코드를 별도의 파일 시스템(HFS의 zFS)에 두고 BPXPRMxx를 업데이트하여 시스템 IPL 도중 이 파일 시스템을 마운트하는 것이 좋습니다. 이 업데이트를 계속 수행해야 하는 경우 샘플 마운트 명령 반복이 포함됩니다.

```

MOUNT FILESYSTEM('#dsn')
  MOUNTPPOINT('-PathPrefix-usr/lpp/rdzutil')
  MODE(RDWR) /* can be MODE(READ) */
  TYPE(ZFS) PARM('AGGRGROW') /* zFS, with extents */
/* TYPE(HFS) */ /* HFS, auto. extent */

```

COMMNDxx에 시작 태스크 추가

Developer for System z RSED 및 JMON 서버에 대한 시작 명령을 SYS1.PARMLIB (COMMANDxx)에 추가하여 다음 시스템 IPL에서 자동으로 시작하도록 하십시오. IPL 동안 사용해야 하는 COMMNDxx parmlib 멤버를 지정하려면 IEASYSxx parmlib 멤버에서 CMD=xx를 정의하십시오.

선택적 통합 디버거의 경우, Developer for System z DBGMR 서버가 사용자의 시스템에서 활성 상태여야 합니다.

서버를 정의 및 구성한 후에는 다음 콘솔 명령을 사용하여 동적으로(다음 IPL 때까지) 시작될 수 있습니다.

- S RSED
- S JMON
- S DBGMR

참고: 서버의 경우, 특별한 시작 순서는 없습니다. 유일한 요구사항은 첫 번째 사용자가 연결을 시도하기 전에 서버를 구동시켜 실행하는 것입니다.

IEASVCxx의 SVC 정의

선택적 통합 디버거는 읽기 전용 메모리에 로드된 CICS 트랜잭션을 디버깅할 수 있습니다. 이 경우 Developer for System z 수퍼바이저 호출(SVC)이 시스템에 정의되어 있어야 합니다.

설치 정의 SVC는 SYS1.PARMLIB(IEASVCxx)에 정의되며 IPL의 활성화를 필요로 합니다. 관련된 로드 모듈이 IPL 시에 LPA에 로드되어야 합니다. IPL 동안 사용해야 하는 IEASVCxx parmlib 멤버를 지정하려면 IEASYSxx parmlib 멤버에서 SVC=xx를 정의하십시오.

Developer for System z SVC를 정의하려면 IEASVCxx에 다음을 지정하십시오.

```
SVC Parm 251, REPLACE, TYPE(4), EPNAME(AQESVC01) /* RDz debug */
```

기본 SVC 번호를 사용하지 않는 경우 SVC Parm 정의의 251 값을 선택한 번호로 변경하고 DBGMGR 시작 태스크 JCL에서 SVC 시작 매개변수를 업데이트하십시오.

LPALSTxx의 LPA 정의

선택적 CARMA(Common Access Repository Manager) 서비스는 CARMA 서버에 대해 다른 서버 시작 메소드를 지원합니다. CRASTART 시작 메소드에서는 FEK.SFEKLPA 로드 라이브러리의 CRASTART 모듈이 링크 팩 영역(LPA)에 있어야 합니다.

선택적 통합 디버거는 읽기 전용 메모리에 로드된 CICS 트랜잭션을 디버깅할 수 있습니다. 이 경우 IPL 동안 FEK.SFEKLPA 로드 라이브러리의 로드 모듈 AQESVC01이 링크 팩 영역(LPA)에 있어야 합니다.

LPA 데이터 세트는 SYS1.PARMLIB(LPALSTxx)에 정의되어 있습니다. IPL 동안 사용해야 하는 LPALSTxx parmlib 멤버를 지정하려면 IEASYSxx parmlib 멤버에서 LPA=xx를 정의하십시오.

LPA 정의는 다음 콘솔 명령을 사용하여 (다음 IPL 때까지) 동적으로 설정될 수 있습니다.

- SETPROG LPA,ADD,DSN=FEK.SFEKLPA

참고:

- LPALSTxx에 나열된 데이터 세트는 마스터 카탈로그 또는 LPALSTxx 멤버에서 식별된 사용자 카탈로그에서 카탈로그화해야 합니다.
- LPALSTxx에 새 데이터 세트를 추가하려면 CLPA(LPA 작성)에서 IPL을 활성화해야 합니다.

- LPA로 로드되는 모든 라이브러리는 APF 권한을 받았고 프로그램 제어를 받는 것으로 자동 간주됩니다. 이 라이브러리를 대신하여 적합한 보안 제어를 가지고 있는지 확인하십시오.
- LPA 배치를 위해 디자인한 라이브러리를 LPA에 배치하지 않도록 선택하고 대신 LINKLIST 또는 STEPLIB를 사용하는 경우 APF 권한 부여 및 프로그램 제어 상태를 정의했는지 확인하십시오.

PROGxx에 APF 권한 부여

JES 작업 모니터가 JES 스킵 파일에 액세스하려면 FEK.SFEKAUTH 로드 라이브러리 및 LE(Language Environment[®]) 런타임 라이브러리(CEE.SCEERUN*)에 있는 FEJJMON 모듈이 APF 권한을 받아야 합니다.

선택적 디버그 관리자가 작동하려면 FEK.SFEKAUTH 로드 라이브러리의 AQEZPCM 모듈이 APF 권한을 받아야 합니다.

선택적 SCLM 개발자 툴킷 서비스가 작동하려면, REXX 런타임 라이브러리 (REXX.*.SEAGLPA)가 APF 권한을 받아야 합니다.

ISPF가 TSO/ISPF TSO/ISPF Client Gateway를 작성하려면, SYS1.LINKLIB에 있는 ISPZTS0 모듈이 APF 권한을 받아야 합니다. TSO/ISPF Client Gateway는 Developer for System z의 TSO 명령 서비스와 SCLM 개발자 툴킷에서 사용됩니다.

기본적으로 APF 권한 부여는 SYS1.PARMLIB(PROGxx)에 정의되어 있습니다. IPL 동안 사용해야 하는 PROGxx parmlib 멤버를 지정하려면 IEASYSxx parmlib 멤버에서 PROG=xx를 정의하십시오.

APF 권한 부여는 다음 콘솔 명령을 사용하여 동적으로 설정할 수 있습니다(다음 IPL 까지). 여기서 volser는 데이터 세트가 상주하는 볼륨입니다(SMS가 관리되지 않는 경우).

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

참고:

- REXX 제품 패키지에 대한 대체 라이브러리를 사용하는 경우, 기본 REXX 런타임 라이브러리 이름은 이전 샘플에서 사용된 대로 REXX.*.SEAGLPA 대신 REXX.*.SEAGALT입니다.
- REXX.*.SEAGLPA와 같은 LPA 라이브러리는 LPA에 위치할 때 자동적으로 APF 권한을 받으므로 명시적 정의가 필요하지 않습니다.

- 공동 필수 제품(예: IBM 파일 관리자) 일부에서도 APF 권한 부여가 필요합니다. 자세한 정보는 관련 제품 사용자 정의 안내서를 참조하십시오.

PROGxx의 LINKLIST 정의

Developer for System z의 LINKLIST 정의는 3가지 카테고리로 그룹화될 수 있습니다.

- Developer for System z 기능에 필요한 Developer for System z 로드 라이브러리. 이 정의에 대해서는 이 절에서 기술합니다.
- Developer for System z 기능에 필요한 필수 로드 라이브러리. 이 정의는 25 페이지의 『필수 LINKLIST 및 LPA 정의』에서 설명합니다.
- 기타 제품에 필요한 Developer for System z 로드 라이브러리. 이 정의는 27 페이지의 『기타 제품에 대한 LINKLIST 정의』에서 설명합니다.

표 7. 기능과 로드 모듈 일치

로드 라이브러리	로드 모듈	사용법	STEPLIB
FEK.SFEKAUTH	AQE* 및 CEE*	146 페이지의 『(선택사항) 통합 디버거』	ELAXFGO 프로시저 또는 CICS
	FEJJ*	27 페이지의 『PROCLIB 변경사항』 (JES 작업 모니터 시작 태스크)	시작 태스크 프로시저
FEK.SFEKLMOD	IRZ* 및 IIRZ*	145 페이지의 『(선택사항) 생성된 코드의 진단 IRZ 메시지』	CICS, IMS 또는 MVS 일괄처리
FEK.SFEKLOAD	AND*	119 페이지의 제 5 장 『(선택사항) 애플리케이션 배치 관리자(더 이상 사용되지 않음)』	CICS
	AZU* 및 IAZU*	142 페이지의 『(선택사항) Enterprise COBOL 및 PL/I용 xUnit 지원』	rsed.envvars 또는 MVS 일괄처리
	BWB*	111 페이지의 제 4 장 『(선택사항) SCLM 개발자 툴킷』	rsed.envvars
	CRA*	69 페이지의 제 3 장 『(선택사항) CARMA(Common Access Repository Manager)』	CRASUB* 또는 crastart*.conf
	ELAX*	33 페이지의 『ELAXF* 원격 빌드 프로시저』 (오류 피드백 및 포함 프리프로세서)	ELAXF* 프로시저
	FEJB*	144 페이지의 『(선택사항) CICS 양방향 언어 지원』	CICS
FEK.SFEKLPA	CRA*	69 페이지의 제 3 장 『(선택사항) CARMA(Common Access Repository Manager)』	CRASRV.properties

표시된 Developer for System z 서비스가 올바르게 작동하려면, 서비스와 관련되는 24 페이지의 표 7에서 기술된 모든 모듈이 STEPLIB 또는 LINKLIST(또는 LPA)를 통해 이용할 수 있어야 합니다. SFEKLMOD 라이브러리는 Developer for System z 자체에서 사용되지 않고 Developer for System z가 생성한 코드에서 사용됨을 참고하십시오. STEPLIB를 사용하여 STEPLIB(또는 CICS의 경우 DFHRPL)를 반드시 정의해야 하는 위치를 알도록 선택하는 경우 24 페이지의 표 7의 STEPLIB 열을 참조하십시오. 그러나 다음을 참조해야 합니다.

- z/OS UNIX에서 STEPLIB를 사용하면 성능이 저하됩니다.
- 하나의 STEPLIB 라이브러리에 APF 권한이 부여되면, 모두에게 권한이 부여된 것입니다. 라이브러리는 STEPLIB의 권한이 부여되지 않은 라이브러리와 혼합되면 APF 권한을 유실합니다.
- JCL의 STEPLIB DD에 추가되는 라이브러리는 JCL로 시작되는 z/OS UNIX 프로세스로 전파되지 않습니다.

사이트가 IBM 권장사항을 따랐다면 LINKLIST 데이터 세트는 SYS1.PARMLIB(PROGxx)에 정의되어 있습니다. IPL 동안 사용해야 하는 PROGxx parmlib 멤버를 지정하려면 IEASYSxx parmlib 멤버에서 PROG=xx를 정의하십시오.

필수 정의는 다음과 유사합니다. 여기서 listname가 활성화될 LINKLIST의 이름이고 volser는 마스터 카탈로그에 카탈로그되지 않은 경우 데이터 세트가 상주하는 볼륨입니다.

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

LINKLIST 정의는 다음 콘솔 명령 그룹을 사용하여 동적으로 작성할 수 있습니다(다음 IPL까지). 여기서, volser는 데이터 세트가 상주하는 볼륨입니다(마스터 카탈로그에서 SMS가 카탈로그화되지 않는 경우).

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

필수 LINKLIST 및 LPA 정의

RSE(Remote Systems Explorer)는 MVS 로드 라이브러리에 액세스해야 하는 z/OS UNIX 프로세스입니다. JES 작업 모니터 및 통합 디버거 서버도 시스템, LE(Language Environment) 및 C 라이브러리에 액세스할 수 있어야 합니다. 다음 필수 소프트웨어 라이브러리는 STEPLIB 또는 LINKLIST/LPALIB를 통해 사용할 수 있어야 합니다.

- 시스템 로드 라이브러리
 - SYS1.LINKLIB

- Language Environment 런타임
 - CEE.SCEERUN
 - CEE.SCEERUN2
- C++의 DLL 클래스 라이브러리
 - CBC.SCLBDLL
- ISPF의 TSO/ISPF Client Gateway
 - ISP.SISPLoad
 - ISP.SISPLPA

선택적 서비스 사용을 지원하려면 다음 추가 라이브러리가 STEPLIB 또는 LINKLIST/LPALIB를 통해 사용할 수 있어야 합니다. 이 목록에는 Developer for System z가 상호작용하는 제품(예: z/OS용 IBM 파일 관리자) 고유의 데이터 세트가 포함되지 않습니다.

- REXX 런타임 라이브러리(SCLM 개발자 툴킷 및 RSE_DSICALL=TSO의 경우)
 - REXX.*.SEAGLPA
- 시스템 로드 라이브러리(SSL 암호화용)
 - SYS1.SIEALNKE
- 시스템 로드 라이브러리(z/OS 1.13 이상의 통합 디버거용)
 - SYS1.SIEAMIGE
- 시스템 로드 라이브러리(Enterprise COBOL 및 PL/I 단위 테스트용)
 - SYS1.CSSLIB
 - SYS1.SIXMLOD1

참고:

- REXX 제품 패키지에 대한 대체 라이브러리를 사용하는 경우, 기본 REXX 런타임 라이브러리 이름은 이전 샘플에서 사용된 대로 REXX.*.SEAGLPA 대신 REXX.*.SEAGALT입니다.
- LPA로 로드되는 모든 라이브러리는 APF 권한을 받았고 프로그램 제어를 받는 것으로 자동 간주됩니다. 이 라이브러리를 대신하여 적합한 보안 제어를 가지고 있는지 확인하십시오.
- LPA 배치(예: REXX.*.SEAGLPA)용으로 디자인된 라이브러리는 LINKLIST 또는 STEPLIB를 통해 액세스하는 경우 추가 프로그램 제어 또는 APF 권한 부여가 필요할 수 있습니다.
- 일부 공통 필수 제품(예: IBM 파일 관리자)도 STEPLIB 또는 LINKLIST/LPALIB 정의가 필요합니다. 자세한 정보는 관련 제품 사용자 정의 안내서를 참조하십시오.

기본적으로 LINKLIST 데이터 세트는 SYS1.PARMLIB(PROGxx)에 정의되어 있습니다. LPA 데이터 세트는 SYS1.PARMLIB(LPALSTxx)에 정의되어 있습니다.

STEPLIB를 사용하려면 RSE 구성 파일 rsed.envvars의 STEPLIB 지시문에 LINKLIST/LPALIB를 통해 사용할 수 없는 라이브러리를 정의해야 합니다. 그러나 다음 사항을 알아 두십시오.

- z/OS UNIX에서 STEPLIB를 사용하면 성능이 저하됩니다.
- 하나의 STEPLIB 라이브러리가 APF 권한을 받으면, 나머지 STEPLIB 라이브러리 모두가 승인을 받은 것입니다. 라이브러리는 STEPLIB의 권한이 부여되지 않은 라이브러리와 혼합되면 APF 권한을 유실합니다.
- JCL의 STEPLIB DD에 추가되는 라이브러리는 JCL로 시작되는 z/OS UNIX 프로세스에 전파되지 않습니다.

기타 제품에 대한 LINKLIST 정의

Developer for System z 클라이언트에는 엔터프라이즈 서비스 도구라는 코드 생성 컴포넌트가 있습니다. 생성된 코드에서 진단 오류 메시지를 실행하려면, FEK.SFEKLMOD 로드 라이브러리에 있는 IRZM* 및 IIRZ* 모듈 모두를 STEPLIB 또는 LINKLIST를 통해 사용할 수 있어야 합니다.

기본적으로 LINKLIST 데이터 세트는 SYS1.PARMLIB(PROGxx)에 정의되어 있습니다.

STEPLIB를 사용하도록 선택하면, 코드(IMS 또는 일괄처리 작업)을 실행하는 태스크의 STEPLIB 지시문에서 LINKLIST를 통해 사용할 수 없는 라이브러리를 정의해야 합니다. 그러나, 하나의 STEPLIB 라이브러리가 APF 권한을 받으면 다른 모든 STEPLIB 라이브러리도 승인을 받은 것입니다. 라이브러리는 STEPLIB의 권한이 부여되지 않은 라이브러리와 혼합되면 APF 권한을 유실합니다.

PROCLIB 변경사항

이 절에서는 다음 PROCLIB 변경사항에 대해 설명합니다.

- 28 페이지의 『JMON, JES 작업 모니터 시작 태스크』
- 28 페이지의 『DBGMR, 디버그 관리자 시작 태스크』
- 29 페이지의 『RSED, RSE 디먼 시작 태스크』
- 33 페이지의 『ELAXF* 원격 빌드 프로시저』

추가 정보는 다음 하위 절에서 사용할 수 있습니다.

- 30 페이지의 『PARM 변수에 대한 JCL 제한사항』
- 32 페이지의 『TMPDIR 처리』

다음 절에 표시되는 시작 태스크 및 원격 빌드 프로시저는 사용자의 JES 서브시스템에 정의되는 시스템 프로시저 라이브러리에 있어야 합니다. 다음 절의 지시사항에서는 IBM 기본 프로시저 라이브러리(SYS1.PROCLIB)가 사용됩니다.

JMON, JES 작업 모니터 시작 태스크

멤버 내에 설명된 대로 FEK.#CUST.PROCLIB(JMON) 샘플 시작 태스크 멤버를 사용자 정의한 후 SYS1.PROCLIB로 복사하십시오. 다음 코드 샘플에 표시된 대로 이 정보를 제공하십시오.

- (권한이 부여된) 로드 라이브러리의 상위 레벨 규정자(기본값: FEK)
- JES 작업 모니터 구성 파일(기본값: FEK.#CUST.PARMLIB(FEJJCNFG))

```

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

```

그림 1. JMON: JES 작업 모니터 시작 태스크

참고:

- 시작 매개변수에 대한 자세한 정보는 209 페이지의 제 11 장 『운영자 명령』의 내용을 참조하십시오.
- 샘플 JCL의 초기 이름은 FEK.SFEKSAMP(FEJJJCL)이고, 18 페이지의 『사용자 정의 설치』에서는 FEK.#CUST.PROCLIB(JMON)로 이름이 바뀝니다.
- 209 페이지의 제 11 장 『운영자 명령』에서 기술된 대로 추적은 콘솔 명령에 의해서도 제어될 수 있습니다.
- 이 태스크를 위해 추천된 WLM(Workload Manager) 목적의 경우 호스트 구성 참조서(SA30-4501)의 "WLM 고려사항"을 참조하십시오.

DBGMR, 디버그 관리자 시작 태스크

멤버 내에 설명된 대로 FEK.#CUST.PROCLIB(DBGMR) 샘플 시작 태스크 멤버를 사용자 정의한 후 SYS1.PROCLIB로 복사하십시오. 다음 코드 샘플에 표시된 대로 이 정보를 제공하십시오.

- 시간대 오프셋: 기본값(EST5DST)
- 외부(클라이언트-호스트) 통신에 사용하는 포트, 기본값 5335
- 내부(호스트 한정) 통신에 사용하는 포트, 기본값 5336
- 읽기 전용 CICS 트랜잭션을 디버깅하는 데 사용하는 SVC 번호, 기본값 251
- 로드 라이브러리의 상위 레벨 규정자(기본값 FEK)

```

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

```

그림 2. DBGMGR: 디버그 관리자 시작 태스크

참고:

- 이 매개변수는 선택적 시작 태스크입니다. 이는 Developer for System z의 통합 디버거 기능에 의해 사용됩니다. 자세한 정보는 146 페이지의 『(선택사항) 통합 디버거』의 내용을 참조하십시오.
- 샘플 JCL의 초기 이름은 FEK.SFEKSAMP(AQESTC)이고, 18 페이지의 『사용자 정의 설치』에서는 FEK.#CUST.PROCLIB(DBGMGR)로 이름이 바뀝니다.
- 이 태스크를 위해 추천된 WLM(Workload Manager) 목적의 경우 호스트 구성 참조서(SA30-4501)의 "WLM 고려사항"을 참조하십시오.

RSED, RSE 디먼 시작 태스크

멤버 내에 설명된 대로 FEK.#CUST.PROCLIB(RSED) 샘플 시작 태스크 멤버를 사용자 정의한 후 SYS1.PROCLIB로 복사하십시오. 다음 코드 샘플에 표시된 대로 이 정보를 제공하십시오.

- Developer for System z가 설치된 홈 디렉토리. 기본값은 /usr/lpp/rdz입니다.
- 구성 파일의 위치(기본값 /etc/rdz)

```

/*
/* RSE DAEMON
/*
//RSED      PROC IVP=,                * IVP 테스트를 수행할 'IVP'
//          PORT=,
//          CNFG='/etc/rdz',
//          HOME='/usr/lpp/rdz'
/*
//RSED      EXEC PGM=BPXBATSL,REGION=0M,TIME=NOLIMIT,
// PARM='PGM &HOME./bin/rsed.sh &IVP -C&CNFG -P&PORT'
//STDOUT    DD SYSOUT=*
//STDERR    DD SYSOUT=*
//          PEND
/*

```

그림 3. RSED: RSE 디먼 시작 태스크

참고:

- 시작 매개변수에 대한 자세한 정보는 209 페이지의 제 11 장 『운영자 명령』의 내용을 참조하십시오.
- 샘플 JCL의 초기 이름은 FEK.SFEKSAMP(FEKRSED)이고, 18 페이지의 『사용자 정의 설치』에서는 FEK.#CUST.PROCLIB(RSED)로 이름이 바뀝니다.
- 작업 이름의 길이를 7자 이내로 제한하십시오. 8자 이름이 사용되면, **modify** 및 **stop** 운영자 명령은 "IEE342I MODIFY REJECTED-TASK BUSY" 메시지와 함께 실패합니다. 이 동작은 하위 프로세스에 대한 z/OS UNIX 디자인으로 인한 것입니다.
- 이 태스크 및 하위 프로세스에 대해 권장하는 워크로드 관리자의 목적은 호스트 구성 참조서 (SA30-4501)에 있는 "WLM 고려사항"의 내용을 참조하십시오. 하위 프로세스의 이름은 상위 태스크의 이름 RSED와 동일하며, 랜덤의 1자 숫자가 첨부됩니다(예: RSED8).

PARM 변수에 대한 JCL 제한사항

PARM 변수의 최대 길이는 100자이며, 사용자 정의 디렉토리 이름을 사용하는 경우 문제가 발생할 수 있습니다. 이 문제점을 피해가려면 다음 옵션 중 하나를 선택하십시오.

- 기본값을 사용합니다.

rsed.sh 시작 스크립트는 인수 없이 시작될 수 있습니다. 이 경우 기본 인수 값이 사용됩니다.

- 기호 링크를 사용합니다.

기호 링크는 긴 디렉토리 이름의 축약형으로 사용될 수 있습니다. 다음 샘플 z/OS UNIX 명령은 다른 디렉토리(/long/directory/name/usr/lpp/rdz)에 대한 기호 링크(/usr/lpp/rdz)를 정의합니다.

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

- STDIN을 사용합니다.

PARM 필드가 비어 있을 때, **BPXBATCH**는 z/OS UNIX 셸을 시작하고, STDIN에서 제공하는 셸 스크립트를 실행합니다. STDIN은 ORDONLY으로서 할당된 z/OS UNIX 파일이어야 합니다. STDIN을 사용하면 TMPDIR과 같은 PROC 변수의 사용이 사용 안함으로 설정됩니다. 셸은 /etc/profile 및 \$HOME/.profile 셸 로그인 스크립트를 실행합니다.

이 메소드를 사용하려면 먼저 다음 샘플과 유사한 내용과 일치하도록 시작 JCL을 업데이트하십시오.

```

/*
/* RSE 디먼 - STDIN 사용
/*
//RSED      PROC CNFG='/etc/rdz'
/*
//RSE       EXEC PGM=BPXBATCH,REGION=0M,TIME=NOLIMIT
//STDOUT    DD SYSOUT=*
//STDERR    DD SYSOUT=*
//STDIN     DD PATHOPTS=(ORDONLY),PATH='&CNFG./rsed.stdin.sh'
//          PEND
/*

```

그림 4. RSED: 대체 RSE 디먼 시작

그리고 나서, RSE 디먼을 시작할 셸 스크립트(이 예에서 /etc/rdz/rsed.stdin.sh)를 작성하십시오. TSO **OEDIT** 명령으로 파일을 편집할 수 있습니다. 이 스크립트의 콘텐츠는 다음과 같은 샘플과 유사합니다.

```

CNFG=/etc/rdz
PORT=
IVP=
/long/directory/name/usr/lpp/rdz/bin/rsed.sh $IVP -C$CNFG -P$PORT -T$TMPDIR

```

그림 5. rsed.stdin.sh: 대체 RSE 디먼 시작

참고: 이 메소드를 사용하는 경우, RSE 디먼 그 자체는 RSED 주소 공간에서 활성화되지 않지만 RSEDx 주소 공간에서는 활성화됩니다. 이는 z/OS UNIX가 별도의 주소 공간에서 (셸 시작과 같은) 하위 프로세스를 운영하기 때문입니다. 지시문은 너무 늦게 해석되므로 **_BPX_SHAREAS=YES** 지시문으로 STDENV DD를 추가해도 이 동작이 변경되지 않습니다. 이 부작용으로 인해 Developer for System z 운영자 명령 사용이 상당히 복잡해 집니다.

TMPDIR 처리

z/OS UNIX는 /tmp 또는 TMPDIR 변수에서 참조되는 다른 디렉토리에 대한 쓰기 액세스 권한이 필요하며, 시작 태스크 시작 도중 특정 명령을 처리할 수 있어야 합니다. Developer for System z는 시작 태스크 시작 도중 다음 논리를 사용하여 TMPDIR을 설정합니다.

시작 태스크 시작 도중 Developer for System z는 TMPDIR이 이미 설정되어 있는지 (DD STDENV) 확인합니다. 설정되어 있으면 시작 태스크는 해당 값을 사용합니다. TMPDIR이 설정되어 있지 않으면 시작 태스크는 /tmp를 사용할 수 있는지 테스트합니다. 사용할 수 없는 경우, 시작 태스크는 시작 태스크 사용자 ID에 지정된 홈 디렉토리를 사용할 수 있는지 테스트합니다. 이 디렉토리도 사용할 수 없는 경우, 시작에 실패합니다.

/tmp의 기본 백업인 홈 디렉토리를 사용할 수 없는 경우, 다음 샘플에서와 같이 DD STDENV를 사용하여 TMPDIR을 사전 정의해야 합니다.

```
//*  
/* RSE DAEMON  
/*  
//RSED      PROC IVP=,                * IVP 테스트를 수행할 'IVP'  
//          PORT=,  
//          CNFG='/etc/rdz',  
//          HOME='/usr/lpp/rdz'  
/*  
//RSED      EXEC PGM=BPXBATSL,REGION=0M,TIME=NOLIMIT,  
// PARM='PGM &HOME./bin/rsed.sh &IVP -C&CNFG -P&PORT'  
//STDOUT    DD SYSOUT=*  
//STDERR    DD SYSOUT=*  
//STDENV    DD PATHOPTS=(ORDONLY),PATH='&CNFG./rsed.stdenv'  
//          PEND  
/*
```

그림 6. RSED: 대체 TMPDIR 처리

그런 다음 TMPDIR 정의를 보관할 파일(이 예제에서는 /etc/rdz/rsed.stdenv)을 작성하십시오. TSO **OEDIT** 명령으로 파일을 편집할 수 있습니다. 이 파일의 콘텐츠는 다음 샘플과 유사합니다.

```
TMPDIR=/tmp
```

그림 7. rsed.stdenv: 대체 TMPDIR 처리

rsed.envvars에 시작 태스크가 rsed.envvars를 해석한 후 사용할 TMPDIR 변수가 있더라도, 시작 장애의 원인이 되므로 rsed.envvars를 DD STDENV에 링크해서는 안됩니다.

ELAXF* 원격 빌드 프로시저

Developer for System z에서는 CICS BMS 맵, IMS MFS 화면, COBOL, PL/I, 어셈블러 및 C/C++ 프로그램의 JCL 생성, 원격 프로젝트 빌드 및 원격 구문 검사 기능에 사용할 수 있는 샘플 JCL 프로시저를 제공합니다. 이러한 프로시저를 사용하면 설치 시 자체 표준을 적용할 수 있으며 개발자는 동일한 컴파일러 옵션 및 컴파일러 레벨로 동일한 프로시저를 사용하게 됩니다.

샘플 프로시저 및 해당 기능이 표 8에 나열되어 있습니다.

표 8. 샘플 ELAXF* 프로시저

멤버	목적
ELAXFADT	상위 레벨 어셈블러 프로그램 어셈블링 및 디버깅을 위한 샘플 프로시저
ELAXFASM	상위 레벨 어셈블러 프로그램 어셈블링을 위한 샘플 프로시저
ELAXFBMS	CICS BMS 오브젝트 및 이와 연관된 복사, dsect 또는 포함 구성원을 작성하는 샘플 프로시저
ELAXFCOC	COBOL 컴파일, 통합 CICS 변환 및 통합 DB2 변환을 수행하기 위한 샘플 프로시저
ELAXFCOP	COBOL 프로그램에 임베드된 EXEC SQL문의 DB2 프리프로세스를 위한 샘플 프로시저
ELAXFCOT	COBOL 프로그램에 임베드된 EXEC CICS 문의 CICS 변환을 수행하는 샘플 프로시저
ELAXFCPC	C 컴파일을 위한 샘플 프로시저
ELAXFCPP	C++ 컴파일을 위한 샘플 프로시저
ELAXFCP1	SCM 프리프로세서 명령문(-INC 및 ++INCLUDE)으로 COBOL 컴파일을 수행하기 위한 샘플 프로시저
ELAXFDCL	TSO 모드로 프로그램을 실행하기 위한 샘플 프로시저
ELAXFGO	GO 단계를 수행하는 샘플 프로시저
ELAXFLNK	C/C++, COBOL, PLI 및 상위 레벨 어셈블러 프로그램 링크를 위한 샘플 프로시저
ELAXFMFS	IMS MFS 화면을 작성하는 샘플 프로시저
ELAXFPLP	PLI 프로그램에 임베드된 EXEC SQL문의 DB2 프리프로세스를 위한 샘플 프로시저
ELAXFPLT	PLI 프로그램에 임베드된 EXEC CICS 문의 CICS 변환을 수행하기 위한 샘플 프로시저
ELAXFPL1	PL/I 컴파일링, 통합 CICS 변환과 통합 DB2 변환을 위한 샘플 프로시저
ELAXFPP1	SCM 프리프로세서 명령문(-INC 및 ++INCLUDE)으로 PL/I 컴파일을 수행하는 샘플 프로시저
ELAXFSP	스토어드 프로시저를 DB2에 등록하는 샘플 프로시저
ELAXFSQL	SQL을 호출하기 위한 샘플 프로시저
ELAXFTSO	TSO 모드에서 생성된 DB2 코드를 실행하고 디버깅하기 위한 샘플 프로시저
ELAXFUOP	CICS 또는 IMS 서브시스템에서 실행되는 프로그램을 빌드할 때 UOPT 단계를 생성하는 샘플 프로시저

프로시저의 이름과 프로시저에 있는 단계의 이름은 Developer for System z 클라이언트에 포함되는 기본 특성과 일치합니다. 프로시저의 이름 또는 프로시저에 있는 단계의

이름이 변경되면, 모든 클라이언트에서 상응하는 특성 파일을 업데이트해야 합니다. 프로시저와 단계 이름을 변경해서는 안됩니다.

샘플 빌드 프로시저 멤버 FEK.#CUST.PROCLIB(ELAXF*)를 멤버 내에 설명된 대로 사용자 정의하여 SYS1.PROCLIB로 복사하십시오. 표 9에 설명된 대로 다양한 제품 라이브러리에 올바른 상위 레벨 규정자를 제공하십시오.

표 9. ELAXF* 상위 레벨 규정자 체크리스트

제품	기본 HLQ	값
Developer for System z	FEK	
CICS	CICSTS42.CICS	
DB2	DSNA10	
IMS	IMS	
COBOL	IGY.V4R2M0	
PL/I	PLI.V4R2M0	
C/C++	CBC	
LE	CEE	
시스템 LINKLIB	SYS1	
시스템 MACLIB	SYS1	

일부 ELAXF* 프로시저 참조 데이터는 고정 하위 레벨 규정자가 없는 이름을 설정합니다. 그 예로는 DB2 관리자가 컴파일하는 DB2 유틸리티를 보유하는 DB2 실행 라이브러리가 있습니다. 표 10을 사용하여 기본 데이터 세트 이름을 사용자 사이트에서 사용하는 이름으로 맵핑합니다.

표 10. ELAXF*. 완전한 데이터 세트 체크리스트

제품	기본 DSN	값
Developer for System z - SQL 샘플	FEK.#CUST.SQL	
DB2 실행 라이브러리	DSNA10.RUNLIB.LOAD	

ELAXF* 프로시저를 시스템 프로시저 라이브러리로 복사할 수 없으면 Developer for System z 사용자에게 JCLLIB 카드를 클라이언트의 작업 특성에(JOB 카드 바로 뒤에) 추가하도록 요청하십시오.

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

보안 정의

Developer for System z에 대한 보안 정의를 작성하려면, 샘플 FEKRACF 멤버를 사용자 정의하고 제출하십시오. 이러한 작업을 제출하는 사용자는 보안 관리자 권한(예: RACF SPECIAL)이 있어야 합니다.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 FEKRACF는 FEK.#CUST.JCL에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오.

참고:

- z/OS용 CA ACF2™을 사용하는 사이트의 경우 CA 지원 사이트(<https://support.ca.com>)의 제품 페이지를 참조하거나 관련 Developer for System z 지식 문서, TEC492389를 확인하십시오. 이 지식 문서에는 Developer for System z를 적절히 구성하는 데 필요한 보안 명령에 대한 세부사항이 들어 있습니다.
- z/OS용 CA Top Secret®을 사용하는 사이트의 경우 CA 지원 사이트(<https://support.ca.com>)의 제품 페이지를 참조하거나 관련 Developer for System z 지식 문서, TEC492091을 확인하십시오. 이 지식 기반 데이터베이스 문서에는 Developer for System z를 적절히 구성하는 데 필요한 보안 명령에 대한 세부사항이 들어 있습니다.

171 페이지의 제 9 장 『보안 정의』에서는 Developer for System z의 보안 관련 정의에 대한 다음 목록에 대해 자세히 설명합니다.

- 보안 설정 및 클래스 활성화
- Developer for System z 사용자에게 대한 OMVS 세그먼트 정의
- 데이터 세트 프로파일 정의
- JMON 및 RSED 시작 태스크 정의
- JES 명령 보안 정의
- RSE를 보안 z/OS UNIX 서버로 정의
- RSE 대한 MVS 프로그램 제어 라이브러리 정의
- RSE에 대한 애플리케이션 보안 정의
- RSE에 대한 PassTicket 지원 정의
- RSE 대한 z/OS UNIX 프로그램 제어 파일 정의

참고: 샘플 FEKRACF 작업에는 RACF 명령뿐만 아니라 다른 명령도 포함됩니다. 보안 정의의 마지막 단계는 z/OS UNIX 파일 프로그램을 제어하는 것입니다. 사이트의 정책에 따라 이 작업은 보안 관리자의 태스크가 아닌 시스템 프로그래머의 태스크일 수도 있습니다.

경고: PassTicket이 올바르게 설정되지 않으면 클라이언트 연결 요청이 실패합니다.

FEJJCNFG, JES 작업 모니터 구성 파일

JES 작업 모니터(JMON)는 모든 JES 관련 서비스를 제공합니다. JES 작업 모니터의 동작은 FEJJCNFG의 정의로 제어할 수 있습니다.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 FEJJCNFG는 FEK.#CUST.PARMLIB에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오.

샘플 JES 작업 모니터 구성 멤버 FEJJCNFG를 다음 샘플에 표시된 대로 사용자 정의하십시오. US 코드 페이지를 사용할 때 주석 행은 숫자 부호(#)로 시작합니다. 데이터 라인에는 지시문과 할당값만 있을 수 있습니다. 주석은 같은 라인에서 허용되지 않습니다.

참고: 변경된 내용을 적용하려면, JMON 시작 태스크를 다시 시작해야 합니다.

```
SERV_PORT=6715
TZ=EST5EDT
#APPLID=FEKAPPL
#AUTHMETHOD=SAF
#CODEPAGE=UTF-8
#CONCHAR=$
#CONSOLE_NAME=JMON
#GEN_CONSOLE_NAME=OFF
#HOST_CODEPAGE=IBM-1047
#LIMIT_COMMANDS=NOLIMIT
#LIMIT_CONSOLE=LIMITED
#LIMIT_VIEW=USERID
#LISTEN_QUEUE_LENGTH=5
#LOOPBACK_ONLY=ON
#MAX_DATASETS=32
#MAX_THREADS=200
#TIMEOUT=3600
#TIMEOUT_INTERVAL=1200
#TRACE_STORAGE=OFF
#SEARCHALL=OFF
#SUBMIT_TIMEOUT=30
#SUBMITMETHOD=TSO
#TSO_TEMPLATE=FEK.#CUST.CNTL(FEJTSO)
```

그림 8. FEJJCNFG, JES 작업 모니터 구성 파일

SERV_PORT

JES 작업 모니터의 포트 번호. 기본 포트는 6715입니다. 포트는 필요하다면 변경할 수 있습니다.

참고:

- 이 값은 rsed.envvars 구성 파일의 JES 작업 모니터에 대해 설정된 포트 번호와 일치해야 합니다. 이 값이 다른 경우 RSE가 JES 작업 모니터로 클라이언트를 연결할 수 없습니다. RSE 변수의 정의 방법을 배우려면 41 페이지의 『rsed.envvars, RSE 구성 파일』의 내용을 참조하십시오.
- 포트를 선택하기 전에 **NETSTAT** 및 **NETSTAT PORTL** TSO 명령을 사용하여 시스템에서 포트를 사용 가능한지 확인하십시오.

TZ 시간대 선택기. 기본값은 EST5EDT입니다. 기본 시간대는 UTC +5 시간 (EST(Eastern Standard Time), EDT(Eastern Daylight Savings Time))입니다. 해당 시간대를 표시하도록 이 값을 변경하십시오. 추가 정보는 *UNIX System Services Command Reference*(SA22-7802)에 있습니다.

다음과 같은 정의는 선택사항입니다. 생략되면, 아래 지정된 대로 기본값이 사용됩니다.

APPLID

보안 소프트웨어에 JES 작업 모니터를 식별하는 데 사용되는 애플리케이션 ID를 지정합니다. 기본값은 FEKAPPL입니다. 필요한 애플리케이션 ID를 주석 해제한 후 변경하십시오.

참고: 이 값은 `rsed.envvars` 구성 파일의 RSE에 대한 애플리케이션 ID 세트와 일치해야 합니다. 이 값이 다른 경우 RSE가 JES 작업 모니터로 클라이언트를 연결할 수 없습니다. RSE 변수의 정의 방법을 배우려면 41 페이지의 『`rsed.envvars`, RSE 구성 파일』의 내용을 참조하십시오.

AUTHMETHOD

기본값은 SAF이며, 시스템 권한 부여 기능(SAF) 보안 인터페이스가 사용됨을 의미합니다. IBM 지원 센터에서 지시한 경우가 아니면 변경하지 마십시오.

CODEPAGE

워크스테이션 코드 페이지. 기본값은 UTF-8입니다. 워크스테이션 코드 페이지는 UTF-8로 설정되고, 일반적으로 변경할 수 없습니다. 통화 기호와 같이 다국어 문자에 어려움이 있으면, 지시문의 주석을 해제하고 워크스테이션의 코드 페이지와 일치하도록 UTF-8을 변경해야 합니다.

CONCHAR

JES 콘솔 명령 문자를 지정합니다. CONCHAR의 기본값은 JES2의 경우 `CONCHAR=$`이고, JES3의 경우 `CONCHAR=*`입니다. 요청된 명령 문자를 주석 해제한 후 변경하십시오.

CONSOLE_NAME

작업에 대해 명령(보류, 해제, 취소 및 제거)을 실행할 때 사용되는 EMCS 콘솔의 이름을 지정합니다. 기본값은 JMON입니다. 다음 가이드라인을 사용하여, 필요한 콘솔 이름을 주석 해제하고 변경하십시오.

- `CONSOLE_NAME`은 2 - 8 영숫자 문자로 구성된 콘솔 이름 또는 '`&SYSUID`' (따옴표 없음)여야 합니다.
- 콘솔 이름이 지정된 경우 모든 사용자에게 해당 이름의 단일 콘솔이 사용됩니다. 해당 이름의 콘솔이 이미 사용 중이면, 클라이언트가 실행한 명령은 실패합니다.
- `&SYSUID`가 지정되면, 클라이언트 사용자 ID는 콘솔 이름으로 사용됩니다. 그러므로, 사용자 각각에 대해 다른 콘솔이 사용됩니다. 해당 이름의 콘솔이 이

미 사용 중이면(예를 들어, 사용자가 SDSF ULOG를 사용 중인 경우), 클라이언트가 실행한 명령은 GEN_CONSOLE_NAME 설정에 따라 실패할 수 있습니다.

사용된 콘솔 이름에 관계없이, 명령을 요청하는 클라이언트의 사용자 ID는 콘솔의 LU로 사용되며, SYSLOG 메시지 IEA630I 및 IEA631I에 추적이 남겨 집니다.

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

GEN_CONSOLE_NAME

대체 콘솔 이름의 자동 생성을 사용하거나 사용하지 않도록 설정합니다. 기본값은 OFF입니다. 대체 콘솔 이름을 가능하게 하려면 주석 해제한 후 ON으로 변경하십시오.

CONSOLE_NAME이 &SYSUID와 같고 사용자 ID를 콘솔 이름으로 사용할 수 없는 경우에만 이 지시문이 사용됩니다.

GEN_CONSOLE_NAME=ON인 경우, 사용자 ID에 한 개의 숫자 자리를 첨부하여 대체 콘솔 이름이 생성됩니다. 0 - 9의 숫자가 시도됩니다. 사용 가능한 콘솔이 없는 경우 클라이언트가 실행한 명령이 실패합니다.

GEN_CONSOLE_NAME=OFF인 경우 클라이언트가 실행한 명령은 실패합니다.

참고: 유효한 설정은 ON 및 OFF뿐입니다.

HOST_CODEPAGE

호스트 시스템 코드 페이지. 기본값은 IBM-1047입니다. 주석 해제 후 호스트 시스템 코드 페이지에 맞도록 변경하십시오.

이 코드 페이지는 데이터 해석에는 사용되지 않고 서버 오퍼레이션 및 클라이언트 연결 설정에만 사용됩니다. Developer for System z 클라이언트는 데이터 해석에 사용할 코드 페이지를 제공합니다("MVS 파일" 서브시스템의 특성에서 검색됨).

LIMIT_COMMANDS

사용자가 선택한 JES 명령(JCL 표시, 정지, 해제, 취소 및 제거)을 실행할 수 있는 작업을 정의합니다. 기본값(LIMIT_COMMANDS=USERID)은 사용자가 소유하는 작업에 대한 명령으로 한정됩니다. 사용자가 모든 스폴 파일에 대해 명령을 실행할 수 있도록 하려면, 보안 제품에서 허용하는 경우 이 지시문을 주석 해제하고 LIMITED 또는 NOLIMIT를 지정하십시오.

표 11. LIMIT_COMMANDS 명령 권한 매트릭스

LIMIT_COMMANDS	작업 소유자	
	사용자	기타
USERID(기본값)	허용	허용되지 않음

표 11. LIMIT_COMMANDS 명령 권한 매트릭스 (계속)

	작업 소유자	
LIMITED	허용	보안 프로파일에서 명시적으로 허용되는 경우에만 허용
NOLIMIT	허용	보안 프로파일을 통해 또는 JESSPOOL 클래스가 활성화되지 않은 경우 허용

참고: 유효한 설정은 USERID, LIMITED 및 NOLIMIT입니다.

LIMIT_CONSOLE

지원되는 JES 명령(정지, 해제, 취소 및 제거)을 실행할 때 사용되는 콘솔에 부여되는 권한을 정의합니다. 기본값(LIMIT_CONSOLE=LIMITED)은 OPERCMDS 클래스에서 보안 프로파일이 보호하는 명령에 대한 권한으로 한정됩니다. 보안 프로파일이 보호하지 않는 지원되는 JES 명령을 실행하려면, 이 지시문을 주석 해제하고 NOLIMIT를 지정하십시오.

명령에 대한 보안 프로파일이 있으면, 사용자는 LIMIT_CONSOLE 설정에 상관 없이, 명령을 실행하기에 충분한 권한을 가지고 있어야 합니다. 유효한 설정은 LIMITED 및 NOLIMIT입니다.

LIMIT_VIEW

사용자가 볼 수 있는 결과물을 정의합니다. 보안 제품에서 허용되는 경우, 기본값(LIMIT_VIEW=NOLIMIT)에서는 사용자가 모든 JES 결과물을 볼 수 있습니다. 사용자가 소유하는 결과물로만 보기를 제한하려면, 이 지시문을 주석 해제하고 USERID를 지정하십시오.

참고: 유효한 설정은 USERID 및 NOLIMIT입니다.

LISTEN_QUEUE_LENGTH

TCP/IP 청취 큐 길이. 기본값은 5입니다. IBM 지원 센터에서 지시한 경우가 아니면 변경하지 마십시오.

LOOPBACK_ONLY

JES 작업 모니터가 루프 백 주소에만 바인드되는지 또는 사용 가능한 모든 TCP/IP 스택에 바인드되는지를 정의합니다. 이 z/OS 시스템의 로컬 태스크만 JES 작업 모니터에 연결할 수 있으므로 루프백에 바인딩하는 것이 안전합니다. 기본값은 ON입니다. JES 작업 모니터를 모든 TCP/IP 스택에 바인드하려는 경우 이 지시문을 주석 해제하고 OFF를 지정하십시오.

MAX_DATASETS

JES 작업 모니터가 클라이언트에 반환할 스푼된 결과물 데이터 세트(예: SYSOUT, SYSPRINT, SYS00001 등등)의 최대 숫자. 기본값은 32입니다. 최대값은 2147483647입니다.

MAX_THREADS

한 번에 하나의 JES 작업 모니터를 사용할 수 있는 최대 사용자 수. 기본값은 200입니다. 최대값은 2147483647입니다. 이 숫자를 늘리면 JES 모니터 주소 공간의 크기도 늘려야 합니다.

TIMEOUT

클라이언트와의 상호작용 부족으로 인해 스레드가 종료하기 전 시간 길이(초)입니다. 기본값은 3600(1시간)입니다. 최대값은 2147483647입니다. TIMEOUT=0은 기능을 사용 안함으로 설정합니다.

TIMEOUT_INTERVAL

제한시간 검사 사이의 시간(초)입니다. 기본값은 1200입니다. 최대값은 2147483647입니다.

TRACE_STORAGE

스토리지 추적이 가능합니다. 기본값은 OFF입니다. 유효한 값은 ON 및 OFF입니다. IBM 지원 센터에서 지시할 경우에만 사용하십시오. 각 명령 실행 후 스토리지 보고서를 DD SYSOUT에 기록하려면 이 지시문을 주석 해제하고 ON을 지정하십시오.

SEARCHALL

JES 작업 모니터 필터와 일치하는 z/OS UNIX 결과물(예: CRASTART 메소드를 사용하여 시작된 Developer for System z CARMA 서버가 SYSOUT에 기록한 결과물) 및 APPC를 수집하십시오. 기본값은 OFF입니다. 유효한 값은 ON 및 OFF입니다. 추가로 스폴 파일을 수집하려면, 이 지시문을 주석 해제하고 ON을 지정하십시오.

SUBMIT_TIMEOUT

Developer for System z가 TSO_TEMPLATE 작업이 완료할 때까지 기다리는 시간(초)입니다. 기본값은 30입니다. 최대값은 2147483647입니다. 참고: SUBMITMETHOD=TSO도 지정되는 경우가 아니면 SUBMIT_TIMEOUT은 아무 효과가 없습니다.

SUBMITMETHOD=TSO

TSO를 통해 작업을 제출합니다. 기본값(SUBMITMETHOD=JES)은 JES로 직접 작업을 제출합니다. TSO **SUBMIT** 명령을 통해 작업을 제출하려면 이 지시문을 주석 해제하고 TSO를 지정하십시오. 이 메소드에서는 TSO 종료가 호출될 수 있지만 성능상 약점이 있습니다.

참고:

- 유효한 설정은 TSO 및 JES입니다.
- SUBMITMETHOD=TSO가 지정되면, TSO_TEMPLATE도 함께 정의해야 합니다.

TSO_TEMPLATE

TSO를 통해 작업을 제출하기 위한 랩퍼 JCL입니다. 기본값은

FEK.#CUST.CNTL(FEJTSO)입니다. 이 명령문은 TSO **SUBMIT** 명령의 랩퍼로 사용될 JCL의 완전한 멤버명을 말합니다. 자세한 정보는 SUBMITMETHOD 명령문을 참조하십시오.

참고:

- 샘플 랩퍼 작업은 FEK.#CUST.CNTL(FEJTSO)에서 제공됩니다. 필요한 사용자 정의에 대한 자세한 정보는 이 멤버를 참조하십시오.
- SUBMITMETHOD=TSO도 지정되는 경우가 아니면 TSO_TEMPLATE는 아무 효과가 없습니다.

rsed.envvars, RSE 구성 파일

RSE 서버 프로세스(RSE 디먼, RSE 스프레드 풀 및 RSE 서버)는 rsed.envvars의 정의를 사용합니다. 선택적 Developer for System z 및 씨드파티 서비스는 이 구성 파일을 사용하여 자체 용도에 맞게 환경 변수를 정의할 수도 있습니다.

RSE(Remote Systems Explorer)는 클라이언트를 호스트 시스템에 연결하고 특정 서비스를 위해 기타 서버를 시작하는 등 핵심 서비스를 제공합니다.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 rsed.envvars는 /etc/rdz/에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오. TSO **OEDIT** 명령으로 파일을 편집할 수 있습니다.

다음 샘플 rsed.envvars 파일은 사용하는 시스템 환경에 맞게 사용자 정의해야 합니다. US 코드 페이지를 사용할 때 주석 행은 숫자 부호(#)로 시작합니다. 데이터 라인에는 지시문과 할당값만 있을 수 있습니다. 같은 라인에서 주석은 허용되지 않습니다. 등호(=) 주위에 있는 라인 연속과 공백은 지원하지 않습니다.

참고: 변경사항을 적용하려면, RSED 시작 태스크를 다시 시작해야 합니다.

```

#=====
# (1) 필수 정의
JAVA_HOME=/usr/lpp/java/J6.0
RSE_HOME=/usr/lpp/rdz
_RSE_RSED_PORT=4035
_RSE_JMON_PORT=6715
RSE_HLQ=FEK
_RSE_HOST_CODEPAGE=IBM-1047
TZ=EST5EDT
LANG=C
PATH=/bin:/usr/sbin
_CEE_DMPTARG=/tmp
STEPLIB=NONE
#STEPLIB=$STEPLIB:CEE.SCEERUN:CEE.SCEERUN2:CBC.SCLBDLL
_RSE_JAVAOPTS=""
_RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Xms128m -Xmx512m"
_RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Ddaemon.log=/var/rdz/logs"
_RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Duser.log=/var/rdz/logs"
_RSE_JAVAOPTS="$ _RSE_JAVAOPTS -DDSTORE_LOG_DIRECTORY="
_RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Dlog.retention.period=5"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Dmaximum.clients=30"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Dmaximum.threads=520"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Dminimum.threadpool.process=1" #_RSE_JAVAOPTS="$ _RSE_JAVAOPTS
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Ddisplay.users=true"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Dkeep.all.logs=false"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Dkeep.last.log=true"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Denable.standard.log=true"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Denable.port.of.entry=true"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Denable.certificate.mapping=false"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Denable.automount=true"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Denable.audit.log=true"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Daudit.cycle=30"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Daudit.retention.period=0"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Daudit.log.mode=RW.R."
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Daudit.action=<user_exit>"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Daudit.action.id=<userid>"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Dlogon.action=<user_exit>"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Dlogon.action.id=<userid>"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Ddeny.nonzero.port=true"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Dsingle.logon=false"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Dprocess.cleanup.interval=0"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Dreject.logon.threshold=1000000"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Dinclude.c=/etc/rdz/include.conf"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Dinclude.cpp=/etc/rdz/include.conf"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -DCPP_CLEANUP_INTERVAL=60000"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -DRIS_BUFFER=8"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -DAPPLID=FEKAPPL"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -DRSE_DSICALL=TSO"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Dsearch.server.limit.hits=0"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Dsearch.server.limit.datasets=0"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Dsearch.server.limit.lines=0"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -Dsearch.server.limit.timeout=0"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -DDISABLE_TEXT_SEARCH=true"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -DDENY_PASSWORD_SAVE=true"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -DHIDE_ZOS_UNIX=true"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -DDISABLE_REMOTE_INDEX_SEARCH=true"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -DDISABLE_DELETE_IN_SUBPROJECT=true"
# _RSE_JAVAOPTS="$ _RSE_JAVAOPTS -DDSTORE_IDLE_SHUTDOWN_TIMEOUT=3600000"

```



```

#_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DDSTORE_SSL_ALGORITHM=TLSv1.2"
#_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DDSTORE_TCP_NO_DELAY=true"
#_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DDSTORE_TRACING_ON=true"
#_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DDSTORE_MEMLOGGING_ON=true"#=====
# (2) TSO/ISPF Client Gateway의 필수 정의
CGI_ISPHOME=/usr/lpp/ispf
CGI_ISPCONF=/etc/rdz
CGI_ISPWORK=/var/rdz
#STEPLIB=$STEPLIB:ISP.SISPLoad:ISP.SISPLPA:SYS1.LINKLIB
_RSE_ISPF_OPTS=""
#_RSE_ISPF_OPTS="$_RSE_ISPF_OPTS&ISPPROF=&SYSUID..ISPPROF"
#CGI_ISPPREF=&SYSPREF..ISPF.VCMISPF"
#=====
# (3) SCLM 개발자 툴킷의 필수 정의
_SCLMDT_CONF_HOME=/var/rdz/scldmt
#STEPLIB=$STEPLIB:$RSE_HLQ.SFEKAUTH:$RSE_HLQ.SFEKLOAD
#_SCLMDT_TRANTABLE=FEK.#CUST.LSTRANS.FILE
#ANT_HOME=/usr/lpp/Apache/Ant/apache-ant-1.7.1
#=====
# (4) 선택적 정의
#_RSE_PORTRANGE=8108-8118
#_BPXK_SETIBMOPT_TRANSPORT=TCPIP
#TMPDIR=/tmp
#_RSE_FEK_SAF_CLASS=FACILITY
#_RSE_LDAP_SERVER=ldap_server_url
#_RSE_LDAP_PORT=389
#_RSE_LDAP_PTC_GROUP_SUFFIX="o=PTC,c=DeveloperForZ"
#GSK_CRL_SECURITY_LEVEL=HIGH
#GSK_LDAP_SERVER=ldap_server_url
#GSK_LDAP_PORT=ldap_server_port
#GSK_LDAP_USER=ldap_userid
#GSK_LDAP_PASSWORD=ldap_server_password
#STEPLIB=$RSE_HLQ.SFEKLOAD:SYS1.CSSLIB:SYS1.SIXMLOD1
#RSE_UBLD_DD=$CGI_ISPCONF/ISPF.conf
#RSE_UBLD_STEPLIB=$STEPLIB
#=====
# (5) IBM 지원 센터에서 지시한 경우가 아니면 변경하지 마십시오.
_RSE_SAF_CLASS=/usr/include/java_classes/IRRRacf.jar
_CEE_RUNOPTS="ALL31(ON) HEAP(32M,32K,ANYWHERE,KEEP,,) TRAP(ON)"
_BPX_SHAREAS=YES
_BPX_SPAWN_SCRIPT=YES
_EDC_ADD_ERRNO2=1
JAVA_PROPAGATE=NO
RSE_DSN_SFEKLOAD=$RSE_HLQ.SFEKLOAD
RSE_LIB=$RSE_HOME/lib
PATH=.:$JAVA_HOME/bin:$RSE_HOME/bin:$CGI_ISPHOME/bin:$PATH
LIBPATH=$JAVA_HOME/bin:$JAVA_HOME/bin/classic:$RSE_LIB:$RSE_LIB/iclass
LIBPATH=.:usr/lib:$LIBPATH
CLASSPATH=$RSE_LIB:$RSE_LIB/dstore_core.jar:$RSE_LIB/clientserver.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/dstore_extra_server.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/zosserver.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/dstore_miners.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/universalminers.jar:$RSE_LIB/mvsminers.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/carma.jar:$RSE_LIB/luceneminer.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/mvsluceneminer.jar:$RSE_LIB/cdzminer.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/mvscdzminer.jar:$RSE_LIB/jesminers.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/mvsutil.jar:$RSE_LIB/jesutils.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/lucene-core-2.3.2.jar
CLASSPATH=$CLASSPATH:$RSE_LIB/cdtparser.jar:$RSE_LIB/wdzBidi.jar
CLASSPATH=$CLASSPATH:$RSE_SAF_CLASS
CLASSPATH=.:$CLASSPATH
_RSE_PTC=$_RSE_LDAP_PTC_GROUP_SUFFIX
_RSE_ISPF_OPTS="$&SESSION=SPAWN$_RSE_ISPF_OPTS"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Dldap_server_address=$_RSE_LDAP_SERVER"

```



```

_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Xbootclasspath/p:$RSE_LIB/bidiTools.jar"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Dfile.encoding=$_RSE_HOST_CODEPAGE"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Dconsole.encoding=$_RSE_HOST_CODEPAGE"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DDSTORE_INITIAL_SIZE=0"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DDSTORE_MAX_FREE=0"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DDSTORE_SPIRIT_ON=false"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DSPIRIT_EXPIRY_TIME=90"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DSPIRIT_INTERVAL_TIME=6"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Dcom.ibm.cacheLocalHost=true"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Duser.home=$HOME"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Dclient.username=$RSE_USER_ID"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Dlow.heap.usage.ratio=15"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Dmaximum.heap.usage.ratio=40"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DDSTORE_KEEPALIVE_ENABLED=true"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DDSTORE_KEEPALIVE_RESPONSE_TIMEOUT=60000"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DDSTORE_IO_SOCKET_READ_TIMEOUT=180000"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DRSECOMM_LOGFILE_MAX=0"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Djob.monitor.port=$_RSE_JMON_PORT"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Dlock.info.timeout=10000"
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -showversion"
_RSE_SERVER_CLASS=org.eclipse.dstore.core.server.Server
_RSE_DAEMON_CLASS=com.ibm.etools.zos.server.RseDaemon
_RSE_POOL_SERVER_CLASS=com.ibm.etools.zos.server.ThreadPoolProcess
_RSE_SERVER_TIMEOUT=120000
_SCLMDT_BASE_HOME=$RSE_HOME
_SCLMDT_WORK_HOME=$CGI_ISPHOME
CGI_DTWORK=$_SCLMDT_WORK_HOME
_CMDSERV_BASE_HOME=$CGI_ISPHOME
_CMDSERV_CONF_HOME=$CGI_ISPCONF
_CMDSERV_WORK_HOME=$CGI_ISPWORK
#=====
# (6) 추가 환경 변수

```

그림 11. *rsed.envvars*: RSE 구성 파일(계속)

참고: 기호가 *rsed.envvars*에 정의되어 있는 동안은 *rsed.envvars*에서 값과 디렉토리를 지정할 때 기호 링크를 사용할 수 있습니다.

다음 정의가 필요합니다.

JAVA_HOME

Java 홈 디렉토리. 기본값은 /usr/lpp/java/J6.0입니다. 사용하는 Java 설치에 맞게 변경하십시오.

RSE_HOME

RSE 홈 디렉토리. 기본값은 /usr/lpp/rdz입니다. Developer for System z 설치에 맞게 변경하십시오.

_RSE_RSED_PORT

RSE 디먼 포트 번호. 기본값은 4035입니다. 필요한 경우 변경할 수 있습니다.

참고:

- 포트를 선택하기 전에, **NETSTAT** 및 **NETSTAT PORTL** TSO 명령을 사용하여 사용자 시스템에서 포트를 이용 가능한지 확인하십시오.
- 이 포트는 클라이언트-호스트 통신에 사용됩니다.
- RSED 시작 태스크는 여기에서 지정된 포트 번호를 대체할 수 있습니다.

_RSE_JMON_PORT

JES 작업 모니터 포트 번호. 기본값은 6715입니다. 필요한 경우 변경할 수 있습니다.

참고:

- 이 값은 FEJJCNFG 구성 파일의 JES 작업 모니터에 대해 설정된 포트 번호와 일치해야 합니다. 이 값이 다른 경우 RSE가 JES 작업 모니터로 클라이언트를 연결할 수 없습니다. JES 작업 모니터의 변수 정의 방법을 알려면 35 페이지의 『FEJJCNFG, JES 작업 모니터 구성 파일』의 내용을 참조하십시오.
- 포트를 선택하기 전에, **NETSTAT** 및 **NETSTAT PORTL** TSO 명령을 사용하여 사용자 시스템에서 포트를 이용 가능한지 확인하십시오.
- 이 포트에서의 모든 통신은 z/OS 호스트 시스템에만 한정됩니다.

RSE_HLQ

Developer for System z를 설치하는 데 사용되는 상위 레벨 규정자. 기본값은 FEK입니다. Developer for System z 데이터 세트의 위치와 일치하도록 변경하십시오.

_RSE_HOST_CODEPAGE

호스트 시스템 코드 페이지. 기본값은 IBM-1047입니다. 호스트 시스템 코드 페이지와 일치하도록 변경하십시오. 이 코드 페이지는 데이터 해석에는 사용되지 않고 서버 오퍼레이션 및 클라이언트 연결 설정에만 사용됩니다. Developer for System z 클라이언트는 데이터 해석에 사용할 코드 페이지를 제공합니다("MVS 파일" 서브시스템의 특성에서 검색됨).

TZ 시간대 선택기. 기본값은 EST5EDT입니다. 기본 시간대는 UTC +5 시간 (EST(Eastern Standard Time), EDT(Eastern Daylight Savings Time))입니다. 해당 시간대에 맞게 변경하십시오.

추가 정보는 *UNIX System Services Command Reference*(SA22-7802)에 있습니다.

LANG

기본 로케일의 이름을 지정합니다. 기본값은 C입니다. C는 POSIX 로케일을 지정하고 (예를 들어) Ja_JP는 일본어 로케일을 지정합니다. 사용하는 로케일에 맞게 변경하십시오.

PATH

명령 경로. 기본값은 /bin:/usr/sbin:입니다. 필요한 경우 변경할 수 있습니다.

_CEE_DMPTARG

JVM(Java Virtual Machine)에서 사용하는 Language Environment(LE) z/OS UNIX 덤프 위치. 기본값은 /tmp입니다.

STEPLIB

LINKLIST/LPALIB에 없는 MVS 데이터 세트에 액세스합니다. 기본값은 NONE입니다.

다음 STEPLIB 지시문 중 하나 이상을 주석 해제하고 사용자 정의하면 LINKLIST/LPALIB에 필수 소프트웨어 라이브러리가 없어도 됩니다. 다음 목록에서 라이브러리의 사용법에 대한 자세한 정보는 19 페이지의 『PARMLIB 변경사항』의 내용을 참조하십시오.

```
# RSE
STEPLIB=$STEPLIB:CEE.SCEERUN:CEE.SCEERUN2:CBC.SCLBDLL
# ISPF
STEPLIB=$STEPLIB:ISP.SISPLOAD:ISP.SISPLPA:SYS1.LINKLIB# SCLM 개발자 툴킷
STEPLIB=$STEPLIB:$RSE_HLQ.SFEKAUTH:$RSE_HLQ.SFEKLOAD
# zUnit, Enterprise COBOL 및 PL/I에 대한 xUnit 지원
STEPLIB=$STEPLIB:$RSE_HLQ.SFEKLOAD:SYS1.CSSLIB:SYS1.SIXML0D1
```

참고:

- z/OS UNIX에서 STEPLIB를 사용하면 성능이 저하됩니다.
- 하나의 STEPLIB 라이브러리가 APF 권한을 받으면, 나머지 STEPLIB 라이브러리 모두가 승인을 받은 것입니다. 라이브러리는 STEPLIB의 권한이 부여되지 않은 라이브러리와 혼합되면 APF 권한을 유실합니다.
- LPA 배치를 위해 디자인된 라이브러리는 LINKLIST 또는 STEPLIB를 통해 액세스될 경우 추가 프로그램 제어나 APF 권한 부여가 필요할 수도 있습니다.
- 서버 JCL에서 STEPLIB DD 문을 코딩해도 요청된 STEPLIB 연결이 설정되지 않습니다.

_RSE_JAVAOPTS

추가 RSE 특정 Java 옵션. 이 정의에 대한 자세한 정보는 54 페이지의 『_RSE_JAVAOPTS를 사용하여 여분의 Java 시작 매개변수 정의』의 내용을 참조하십시오.

TSO 명령 서비스 또는 SCLM 개발자 툴킷에 ISPF의 TSO/ISPF Client Gateway를 사용하는 경우 다음 정의가 필요합니다.

CGI_ISPHOME

TSO/ISPF Client Gateway 서비스를 제공하는 ISPF 코드의 홈 디렉토리. 기

본값은 /usr/lpp/ispf입니다. 사용하는 ISPF 설치에 맞게 변경하십시오. 이 지시문은 ISPF의 TSO/ISPF Client Gateway가 사용되는 경우에만 필요합니다.

CGI_ISPCONF

ISPF 기본 구성 디렉토리. 기본값은 /etc/rdz입니다. TSO/ISPF Client Gateway 사용자 정의 파일 ISPF.conf의 위치에 맞게 변경하십시오. 이 지시문은 ISPF의 TSO/ISPF Client Gateway가 사용되는 경우에만 필요합니다.

CGI_ISPWORK

ISPF 기본 작업 디렉토리. 기본값은 /var/rdz입니다. TSO/ISPF Client Gateway에서 사용하는 WORKAREA 디렉토리의 위치에 맞게 변경하십시오. 이 지시문은 ISPF의 TSO/ISPF Client Gateway가 사용되는 경우에만 필요합니다.

참고:

- TSO/ISPF Client Gateway는 CGI_ISPWORK에 지정된 경로에 /WORKAREA를 추가합니다. 자체적으로 추가하지 마십시오.
- SFEKSAMP(FEKSETUP) 샘플 작업을 사용하여 사용자 정의할 수 있는 환경을 빌드하지 않는 경우, CGI_ISPWORK에서 지정된 경로에 WORKAREA 디렉토리가 있는지 확인하십시오. 디렉토리 권한 비트는 777이어야 합니다.

STEPLIB

STEPLIB는 필요한 정의 절에서 이전에 설명되었습니다.

_RSE_ISPF_OPTS

추가 TSO/ISPF Client Gateway 특정 Java 옵션. 기본값은 ""입니다. 이 정의에 대한 자세한 정보는 63 페이지의 『_RSE_ISPF_OPTS를 사용한 여분의 Java 시작 매개변수 정의』의 내용을 참조하십시오. 이 지시문은 ISPF의 TSO/ISPF Client Gateway가 사용되는 경우에만 필요합니다.

CGI_ISPPREF

TSO/ISPF Client Gateway에서 작성된 임시 데이터 세트의 상위 레벨 규정자. 기본값은 "&SYSPREF..ISPF.VCMISPF"입니다. 사용자의 데이터 세트 이름 지정 규칙에 맞도록 주석 해제 후 변경하십시오. 이 지시문은 ISPF의 TSO/ISPF Client Gateway가 사용되는 경우에만 필요합니다.

데이터 세트 이름에서 사용할 수 있는 변수는 다음과 같습니다.

- &SYSUID. - 개발자의 사용자 ID를 대체
- &SYSPREF. - 개발자의 TSO 접두부 또는 사용자 ID(TSO 접두부를 판별할 수 없는 경우)를 대체
- &SYSNAME. - IEASYMxx parmlib 멤버에 지정된 대로 시스템 이름을 대체

참고: 이 지시문에는 ISPF APAR OA38740이 필요합니다.

SCLM 개발자 툴킷이 사용되면 다음 정의가 필요합니다.

_SCLMDT_CONF_HOME

SCLM 개발자 툴킷 기본 구성 디렉토리. 기본값은 /var/rdz/sclmdt입니다. SCLM 프로젝트 정보를 저장하기 위해 SCLMDT가 사용하는 CONFIG 디렉토리의 위치와 일치하도록 변경하십시오. 이 지시문은 SCLMDT가 사용될 경우에만 필요합니다.

참고: SCLMDT는 SCLMDT_CNF_HOME에 지정된 경로에 /CONFIG 및 /CONFIG/PROJECT를 추가합니다. 자체적으로 추가하지 마십시오.

STEPLIB

STEPLIB는 필요한 정의 절에서 이전에 설명되었습니다.

_SCLMDT_TRANTABLE

긴/짧은 이름 변환 VSAM의 이름. 기본값은 FEK.#CUST.LSTRANS.FILE입니다. 주석 해제 후 ISP.SISPSAMP(FLM02LST) SCLM 샘플 작업에서 사용되는 이름과 일치하도록 변경하십시오. 이 지시문은 SCLM 개발자 툴킷에서 긴/짧은 이름 변환이 사용되는 경우에만 필요합니다.

ANT_HOME

Ant 설치를 위한 홈 디렉토리. 기본값은 usr/lpp/apache/Ant/apache-ant-1.7.1입니다. Ant 설치에 적합하도록 변경하십시오. 이 지시문은 Java EE 빌드 지원이 SCLM 개발자 툴킷과 함께 사용하는 경우에만 필요합니다.

다음과 같은 정의는 선택사항입니다. 생략되면 기본값이 사용됩니다.

_RSE_PORTRANGE

RSE 서버가 클라이언트와의 통신을 위해 열 수 있는 포트 범위를 지정합니다. 기본적으로 어느 포트든지 사용할 수 있습니다. 이 정의에 대한 자세한 정보는 53 페이지의 『RSE 서버에 사용 가능한 PORTRANGE 정의』의 내용을 참조하십시오. 이것은 선택적 지시문입니다.

_BPXK_SETIBMOPT_TRANSPORT

사용할 TCP/IP 스택의 이름을 지정합니다. 기본값은 TCPIP입니다. 관련된 TCPIP.DATA의 TCPIPJOBNAME 명령문에서 정의한 대로 요청된 TCP/IP 스택 이름을 주석 해제한 후 변경하십시오. 이것은 선택적 지시문입니다.

참고:

- 서버 JCL에서 SYSTCPD DD 문을 코딩해도 요청된 스택 연관 관계가 설정되지 않습니다.
- 이 지시문이 활성화되지 않으면, RSE는 시스템에서 사용 가능한 모든 스택에 바인드됩니다(BIND INADDRANY).

TMPDIR

임시 파일을 저장하는 데 사용되는 경로를 지정합니다. 기본값은 /tmp입니다. 주식 해제한 후 요청된 경로를 사용하도록 변경하십시오. 이것은 선택적 지시문입니다.

_RSE_FEK_SAF_CLASS

FEK.* 프로파일이 정의된 보안 클래스를 지정합니다. 기본값은 FACILITY입니다. 지정된 값을 강제로 사용하려면 주식 해제 후 변경하십시오. 이것은 선택적 지시문입니다.

_RSE_LDAP_SERVER

클라이언트로 푸시 기능에서 사용하는 LDAP 서버 호스트 이름을 지정합니다. 기본값은 현재 z/OS 호스트 이름입니다. 지정된 값을 강제로 사용하려면 주식 해제 후 변경하십시오. 이것은 선택적 지시문입니다.

_RSE_LDAP_PORT

클라이언트로 푸시 기능에서 사용하는 LDAP 서버 포트를 지정합니다. 기본값은 389입니다. 지정된 값을 강제로 사용하려면 주식 해제 후 변경하십시오. 이것은 선택적 지시문입니다.

_RSE_LDAP_PTC_GROUP_SUFFIX

LDAP 서버 내에서 클라이언트로 푸시 그룹을 찾는 데 필요한 "O=<organization>,C=<country>" 접미어를 지정합니다. 기본값은 "O=PTC,C=DeveloperForZ"입니다. 지정된 값을 강제로 사용하려면 주식 해제 후 변경하십시오. 이것은 선택적 지시문입니다.

GSK_CRL_SECURITY_LEVEL

인증서 유효성 검증 도중 폐지된 인증서의 CRL을 확인하기 위해 LDAP 서버에 접속할 때 보안 SSL 애플리케이션이 사용하는 레벨을 지정합니다. 기본값은 MEDIUM입니다. 지정된 값을 강제로 사용하려면 주식 해제 후 변경하십시오. 이것은 선택적 지시문입니다. 올바른 값은 다음과 같습니다.

- LOW: LDAP 서버에 연결할 수 없으면 인증서 유효성 검증이 실패하지 않습니다.
- MEDIUM: 인증서 유효성 검증을 수행하려면 LDAP 서버에 연결할 수 있어야 하지만 CRL을 정의할 필요는 없습니다. 이 값이 기본값입니다.
- HIGH: 인증서 유효성 검증을 수행하려면 LDAP 서버에 연결할 수 있어야 하고 CRL을 정의해야 합니다.

참고: 이 지시문은 z/OS 1.9 이상이 필요합니다.

GSK_LDAP_SERVER

하나 이상의 공백으로 구분된 LDAP 서버 호스트 이름을 지정합니다. CRL을 확보하기 위해 지정된 LSP 서버를 강제로 사용하려면, 주식 해제 후 변경하십시오. 이것은 선택적 지시문입니다.

호스트 이름은 TCP/IP 주소 또는 URL일 수 있습니다. 각 호스트 이름에는 콜론 부호(:)를 사용하여 호스트 이름에서 구분되는 선택적 포트 번호가 포함될 수 있습니다.

GSK_LDAP_PORT

LDAP 서버 포트를 지정합니다. 기본값은 389입니다. 지정된 값을 강제로 사용하려면 주석 해제 후 변경하십시오. 이것은 선택적 지시문입니다.

GSK_LDAP_USER

LDAP 서버에 접속할 때 사용할 식별 이름을 지정합니다. 지정된 값을 강제로 사용하려면 주석 해제 후 변경하십시오. 이것은 선택적 지시문입니다.

GSK_LDAP_PASSWORD

LDAP 서버에 접속할 때 사용할 비밀번호를 지정합니다. 지정된 값을 강제로 사용하려면 주석 해제 후 변경하십시오. 이것은 선택적 지시문입니다.

RSE_UBLD_DD

TSO 또는 ISPF 명령을 호출하는 Developer for System z 클라이언트에서 IBM Rational Team Concert™ 사용자 빌드를 위한 JCL을 생성할 때 사용되는 DD문을 지정합니다. 기본적으로 Developer for System z는 rsed.envvars의 CGI_ISPCONF에서 참조하는 ISPF.conf의 정의를 사용합니다. 주석 해제한 후, 지정된 파일에서 DD 정의를 사용하도록 변경하십시오. 이 경우 *ISPF*의 *TSO/ISPF 클라이언트 게이트웨이 구성 파일인 ISPF.conf*에 지정된 구문 규칙을 따라야 합니다. 이것은 선택적 지시문입니다.

RSE_UBLD_STEPLIB

TSO 또는 ISPF 명령을 호출하는 Developer for System z 클라이언트에서 IBM Rational Team Concert 사용자 빌드를 위한 JCL을 생성할 때 사용되는 STEPLIB문을 지정합니다. 기본적으로 Developer for System z는 rsed.envvars의 STEPLIB 정의를 사용합니다. 주석 해제한 후, 지정된 STEPLIB 정의를 사용하도록 변경하십시오. 이것은 선택적 지시문입니다.

다음 정의가 필요하며, IBM 지원 센터에서 지시하는 경우가 아니면 변경해서는 안됩니다.

_CEE_RUNOPTS

Language Environment(LE) 런타임 옵션. 기본값은 "ALL31(ON) HEAP(32M,32K,ANYWHERE,KEEP,,) TRAP(ON)"입니다. 수정하지 마십시오.

_BPX_SHAREAS

셸과 같은 주소 공간에서 전경 프로세스를 실행합니다. 기본값은 YES입니다. 수정하지 마십시오.

_BPX_SPAWN_SCRIPT

spawn() 함수에서 직접 셸 스크립트를 실행합니다. 기본값은 YES입니다. 수정하지 마십시오.

_EDC_ADD_ERRNO2

z/OS UNIX 오류 메시지에서 이유 코드를 표시합니다. 기본값은 1입니다. 수정하지 마십시오.

JAVA_PROPAGATE

스레드 작성 도중 보안 및 워크로드 컨텍스트를 전파합니다(Java 버전 1.4 이전만 해당). 기본값은 NO입니다. 수정하지 마십시오.

RSE_DSN_SFELLOAD

SFELLOAD 로드 라이브러리의 완전한 데이터 세트 이름. 기본값은 \$RSE_HLQ.SFELLOAD입니다. 수정하지 마십시오.

RSE_LIB

RSE 라이브러리 경로. 기본값은 \$RSE_HOME/lib입니다. 수정하지 마십시오.

PATH

명령 경로. 기본값은 .:\$JAVA_HOME/bin:\$RSE_HOME/bin:\$CGI_ISPHOME/bin:\$PATH입니다. 수정하지 마십시오.

LIBPATH

라이브러리 경로. 기본값은 너무 길어서 반복할 수 없습니다. 수정하지 마십시오.

CLASSPATH

클래스 경로. 기본값은 너무 길어서 반복할 수 없습니다. 수정하지 마십시오.

_RSE_ISPF_OPTS

추가적 TSO 명령 서비스별 Java 옵션. 기본값은 "&SESSION=SPAWN\$_RSE_ISPF_OPTS"입니다. 수정하지 마십시오.

_RSE_JAVAOPTS

추가 RSE 특정 Java 옵션. 기본값은 너무 길어서 반복할 수 없습니다. 수정하지 마십시오.

_RSE_SERVER_CLASS

RSE 서버의 Java 클래스. 기본값은 org.eclipse.dstore.core.server.Server입니다. 수정하지 마십시오.

_RSE_DAEMON_CLASS

RSE 디몬의 Java 클래스. 기본값은 com.ibm.etools.zos.server.RseDaemon입니다. 수정하지 마십시오.

_RSE_POOL_SERVER_CLASS

RSE 스레드 풀의 Java 클래스. 기본값은 com.ibm.etools.zos.server.ThreadPoolProcess입니다. 수정하지 마십시오.

_RSE_SERVER_TIMEOUT

밀리초 단위로 된 RSE 서버의 제한시간 값(클라이언트를 대기함). 기본값은 120000(2분)입니다. 수정하지 마십시오.

SCLMDT_BASE_HOME

SCLM 개발자 툴킷 코드의 홈 디렉토리. 기본값은 \$RSE_HOME입니다. 수정하지 마십시오.

SCLMDT_WORK_HOME

SCLM 개발자 툴킷 기본 작업 디렉토리. 기본값은 \$CGI_ISPHOME입니다. 수정하지 마십시오.

CGI_DTWORK

이전 클라이언트에 대한 SCLM 개발자 툴킷 지원. 기본값은 \$_SCLMDT_WORK_HOME입니다. 수정하지 마십시오.

_CMDSERV_BASE_HOME

ISPF TSO/ISPF Client Gateway 서비스 지원. 기본값은 \$CGI_ISPHOME입니다. 수정하지 마십시오.

_CMDSERV_CONF_HOME

ISPF TSO/ISPF Client Gateway 서비스 지원. 기본값은 \$CGI_ISPCONF입니다. 수정하지 마십시오.

_CMDSERV_WORK_HOME

ISPF TSO/ISPF Client Gateway 서비스 지원. 기본값은 \$CGI_ISPWORK입니다. 수정하지 마십시오.

RSE 서버에 사용 가능한 PORTRANGE 정의

이것은 RSE 서버가 클라이언트와 통신할 수 있는 포트를 지정하는 rsed.envvars 사용자 정의의 일부분입니다. 이 포트 범위는 RSE 디먼 포트와 연결되지 않습니다.

포트 사용법에 대한 이해를 돕기 위해 RSE의 연결 프로세스에 대한 간략한 설명이 아래에 나와 있습니다.

1. 클라이언트가 호스트 시스템 포트 4035, RSE 디먼에 연결됩니다.
2. RSE 디먼은 RSE 서버 스레드를 작성합니다.
3. RSE 서버는 연결할 클라이언트의 호스트 시스템 포트를 엽니다. 이 포트 선택사항은 rsed.envvars에서 _RSE_PORTRANGE 정의를 이용하여 구성될 수 있습니다.
4. RSE 디먼은 포트 번호를 클라이언트로 리턴합니다.
5. 클라이언트가 호스트 시스템 포트에 연결됩니다.

참고:

- 프로세스는 REXEC/SSH를 사용하는 선택적 대체 연결 메소드와 유사합니다.

- 자세한 정보는 *호스트 구성 참조서 (SA30-4501)*에 있는 "Developer for system z 이해"의 내용을 참조하십시오.

클라이언트가 z/OS와 통신하기 위한 포트 범위를 지정하려면 `rsed.envvars`에서 다음 라인을 주석 해제한 후 사용자 정의하십시오.

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

참고: 포트 범위를 선택하기 전에 **NETSTAT** 및 **NETSTAT PORTL** 명령을 사용하여 범위가 시스템에서 사용 가능한지 확인하십시오.

PORTRANGE의 형식은 `_RSE_PORTRANGE=min-max`입니다. max 값은 포괄적이지 않습니다. 예를 들어, `_RSE_PORTRANGE=8108-8118` 표현식은 8108에서 8117까지의 포트 번호를 사용할 수 있음을 의미합니다. RSE 서버에서 사용되는 포트 번호는 다음과 같은 순서로 판별됩니다.

1. 클라이언트의 서브시스템 특성에 0이 아닌 포트 번호가 지정되면, 지정된 포트 번호가 사용됩니다. 포트를 사용할 수 없으면 연결에 실패합니다. 이 설정은 권장하지 않습니다.

참고: 호스트 시스템은 `rsed.envvars`에서 `deny.nonzero.port=true` 지시문을 지정하여 이러한 유형의 연결 요청을 거부할 수 있습니다. 이 지시문에 대한 자세한 정보는 『`_RSE_JAVAOPTS`를 사용하여 여분의 Java 시작 매개변수 정의』의 내용을 참조하십시오.

2. 서브시스템 특성의 포트 번호가 0이고 `_RSE_PORTRANGE`가 `rsed.envvars`에서 지정되면, `_RSE_PORTRANGE`에서 지정된 포트 범위가 사용됩니다. 범위 내에 사용 가능한 포트가 없으면 연결에 실패합니다.

RSE 서버는 클라이언트 연결 기간 동안에만 포트가 필요한 것은 아닙니다. 기타 RSE 서버가 포트에 바인드할 수 없는 서버 바인드와 클라이언트 바인드 사이의 기간에서도 필요합니다. 이는 대부분의 연결이 범위의 첫 번째 포트를 사용하고, 나머지 범위는 여러 개의 동기 로그인인 경우 버퍼가 됨을 의미합니다.

3. 서브시스템 특성의 포트 번호가 0이고 `_RSE_PORTRANGE`가 `rsed.envvars`에 지정되지 않으면, 사용 가능한 포트가 사용됩니다.

`_RSE_JAVAOPTS`를 사용하여 여분의 Java 시작 매개변수 정의

다른 `_RSE_*OPTS` 지시문을 사용하는 경우 `rsed.envvars`에는 RSE 프로세스를 시작할 때 Java에 여분의 매개변수를 제공하는 기능이 있습니다. `rsed.envvars`에 포함된 샘플 옵션은 주석 해제하여 활성화시킬 수 있습니다.

`_RSE_JAVAOPTS`는 표준 및 RSE 특정 Java 옵션을 정의합니다.

```
_RSE_JAVAOPTS=""
```

변수 초기화입니다. 수정하지 마십시오.

`_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Xms128m -Xmx512m"`

초기(Xms) 및 최대(Xmx) 힙 크기를 설정합니다. 기본값은 각각 128M과 512M입니다. 필요한 힙 크기 값을 강제 실행하도록 변경합니다. 이 지시문이 주석 처리되면, Java 기본값이 사용되고 기본값은 4M 및 512M입니다(Java 5.0의 경우 1M 및 64M).

참고: 이 지시문의 최적의 가치를 판별하려면 호스트 구성 참조서 (SA30-4501)에서 "주요 자원 정의"의 내용을 참조하십시오.

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

RSE 디먼 및 서버 로깅과 RSE 감사 데이터가 있는 디렉토리. 기본값은 /var/rdz/logs입니다. 필요한 위치를 강제 실행하도록 변경하십시오. 이 지시문이 주석 처리되면, RSE 디먼에 지정된 사용자 ID의 홈 디렉토리가 사용됩니다. 홈 디렉토리는 사용자 ID의 OMVS 보안 세그먼트에 정의됩니다.

참고: 이 지시문 또는 상대부, 홈 디렉토리에서 절대 경로를 지정하지 않으면 (여기서 경로는 슬래시로 시작하지 않음), 실제 로그 위치는 구성 디렉토리에 상대적입니다(기본값은 /etc/rdz).

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

사용자 고유 로그로 연결되는 디렉토리. 기본값은 /var/rdz/logs입니다. 필요한 위치를 강제 실행하도록 변경하십시오. 이 지시문이 주석 처리되거나 값이 널 문자열인 경우, 클라이언트 사용자 ID의 홈 디렉토리가 사용됩니다. 홈 디렉토리는 사용자 ID의 OMVS 보안 세그먼트에 정의됩니다.

참고:

- 이 지시문 또는 상대부, 홈 디렉토리에서 절대 경로를 지정하지 않으면(경로는 슬래시로 시작하지 않음), 실제 로그 위치는 구성 디렉토리에 상대적입니다(기본값은 /etc/rdz).
- 사용자 로그의 전체 경로는 userlog/dstorelog/\$LOGNAME/입니다. 여기서 userlog는 user.log 지시문 값이고, dstorelog는 DSTORE_LOG_DIRECTORY 지시문 값이며 \$LOGNAME은 대문자로 된 클라이언트 사용자 ID입니다.
- 각 클라이언트가 \$LOGNAME을 작성할 수 있도록 userlog/dstorelog의 권한 비트가 설정되었는지 확인하십시오.

`_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DDSTORE_LOG_DIRECTORY="`

이 디렉토리는 user.log 지시문에 지정된 경로에 추가됩니다. 이와 함께 사용자 특정 로그로 이어지는 경로를 작성합니다. 기본값은 널 문자열입니다. 지정된 디렉토리를 강제로 사용하도록 변경하십시오. 이 지시문이 주석 처리되면, .eclipse/RSE/가 사용됩니다.

참고:

- 사용자 로그의 전체 경로는 userlog/dstorelog/\$LOGNAME/입니다. 여기서 userlog는 user.log 지시문 값이고, dstorelog는 DSTORE_LOG_DIRECTORY 지시문 값이며 \$LOGNAME은 대문자로 된 클라이언트 사용자 ID입니다.
- 여기에 지정된 디렉토리는 user.log에 지정된 디렉토리에 상대적이므로, 슬래시(/)로 시작하지 않습니다.
- 각 클라이언트가 \$LOGNAME을 작성할 수 있도록 userlog/dstorelog의 권한 비트가 설정되었는지 확인하십시오.

_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Dlog.retention.period=5"

디먼 및 사용자 로그를 보관하는 일 수. 기본값은 5입니다. 지정된 일 수가 지난 후 로그를 삭제하려면 이 지시문을 사용자 정의하십시오. 한계를 설정하지 않으려면 0을 지정하십시오. 최대값은 365입니다. 디먼 로그 정리는 디먼 활동을 필요로 하는 다음 조치에서 일어납니다. 사용자 로그는 다음에 사용자가 연결할 때 정리됩니다.

다음 지시문은 기본적으로 주석 처리됩니다.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Dmaximum.clients=30"

하나의 스레드 풀이 서비스하는 최대 클라이언트 수. 기본값은 30입니다. 스레드 풀당 클라이언트의 수를 제한하려면 주석 해제한 후 사용자 정의하십시오. 기타 제한사항으로 인해 RSE가 이 한계에 도달하지 못할 수도 있습니다.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Dmaximum.threads=520"

새 클라이언트를 허용할 수 있는 한 스레드 풀의 최대 활성 스레드 수. 기본값은 520입니다. 사용 중인 스레드 수를 기반으로 각 스레드 풀에서 클라이언트의 수를 제한하려면, 주석 해제한 후 사용자 정의하십시오. 각 클라이언트 연결에서는 여러 스레드(17 이상)를 사용하며 기타 제한사항으로 RSE가 이 한계에 도달하지 못할 수 있습니다.

참고: 이 값은 SYS1.PARMLIB(BPXPRMxx)에서 MAXTHREADS 및 MAXTHREADTASKS에 대한 설정보다 작아야 합니다.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Dminimum.threadpool.process=1"

최대 활성 스레드 풀 수. 기본값은 1입니다. 표시된 스레드 풀 프로세스 수 이상을 시작하려면 주석 해제한 후 사용자 정의하십시오. 스레드 풀 프로세스는 RSE 서버 스레드의 로드 밸런싱에 사용됩니다. 필요한 경우 새 프로세스가 더 시작됩니다. 새 프로세스를 미리 시작하면 연결 지연을 방지할 수 있지만 유휴 시간 동안 더 많은 자원을 사용합니다.

참고: single.logon 지시문이 활성화된 경우, minimum.threadpool.process가 1로 설정되어 있더라도 2 이상의 스레드 풀이 시작됩니다. rsed.envvars에서 single.logon의 기본 설정이 활성화됩니다.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Dmaximum.threadpool.process=100"

최대 활성 스레드 풀 수. 기본값은 100입니다. 스레드 풀 프로세스 수를 제한하려면 주식 해제 후 사용자 정의하십시오. 스레드 풀 프로세스는 RSE 서버 스레드의 로드 밸런싱에 사용되므로 프로세스 수를 제한하면 활성 클라이언트 연결 수가 제한됩니다.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Dip6=true"

TCP/IP 버전. 기본값은 false이며, IPv4 인터페이스가 사용됨을 의미합니다. IPv6 인터페이스를 사용하려면 주식 해제한 후 true를 지정하십시오.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Ddisplay.users=true"

활성 사용자의 자동화된 디스플레이. 기본값은 false입니다. 각 사용자 로그인 및 로그오프의 rseserver.log에서 활성 사용자의 자동화된 디스플레이를 사용하려면 주식 해제한 후 true를 지정하십시오.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Dkeep.all.logs=false"

디먼 및 사용자 로그에 대해 임베디드 시간소인이 있는 파일 이름을 사용합니다. 기본값은 true이며, 이는 log.retention.period 설정에 의해 제거될 때까지 로그가 보관됨을 의미합니다. 고정된 로그 파일 이름을 사용하려면 주식 해제한 후 false를 지정하십시오. 이 이름은 디먼이 시작되거나 사용자가 연결할 때마다 바뀝니다.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Dkeep.last.log=true"

이전 세션에 속하는 호스트 로그 파일의 사본을 보존합니다. 기본값은 false입니다. 서버 시작 및 클라이언트 연결 도중 이전 로그 파일의 이름을 *.last로 변경하려면, 주식 해제 후 true를 지정하십시오. .dstore* 사용자 추적 파일은 클라이언트 재연결 후 자동 제거되지도 않고 keep.last.log 처리의 일부도 아니므로 주의하십시오. 이 파일은 수동으로 제거해야 합니다. keep.last.log를 적용하려면 keep.all.logs 지시문을 false로 설정해야 합니다.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Denable.standard.log=true"

스레드 풀의 stdout 및 stderr 스트림을 로그 파일에 기록하십시오. 기본값은 false입니다. stdout 및 stderr 스트림을 저장하려면 주식 해제 후 true를 지정하십시오. 결과로 생기는 로그 파일은 daemon.log 지시문에서 참조하는 디렉토리에 있습니다.

참고:

- **MODIFY RSESTANDARDLOG** 운영자 명령을 사용하여 스트림 로그 파일의 업데이트를 동적으로 중지 또는 시작할 수 있습니다.
- enable.standard.log 지시문이 활성화된 경우에는 사용자별 stdout.log, stderr.log 로그 파일이 없습니다. 이제 사용자별 데이터는 일치 RSE 스레드 풀 스트림에 기록됩니다.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Denable.port.of.entry=true"

POE(Port Of Entry) 검사 옵션. 기본값은 false입니다. 클라이언트 연결에 대한 POE 검사를 강제로 실행하려면 주석 해제 후 true를 지정하십시오. POE 검사 도중, 클라이언트의 IP 주소는 보안 소프트웨어에 의해 네트워크 액세스 보안 영역으로 매핑됩니다. 클라이언트 사용자 ID에는 보안 영역을 정의하는 프로파일을 사용할 수 있는 권한이 있어야 합니다.

참고:

- POE 검사는 사용자의 보안 제품에서도 사용할 수 있어야 합니다.
- POE 검사를 사용으로 설정하면 INETD와 같은 기타 z/OS UNIX 서비스에 대한 제품도 사용할 수 있습니다.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Denable.certificate.mapping=false"

보안 소프트웨어를 사용하여 X.509 인증서와 함께 로그온을 인증합니다. 기본값은 true입니다. RSE 디먼이 보안 소프트웨어의 X.509 지원에 의존하지 않고 인증을 수행할 수 있도록 하려면 주석 해제 후 false로 지정하십시오.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Denable.automount=true"

z/OS UNIX 자동 마운트에서 작성하는 홈 디렉토리를 지원합니다. 기본값은 false입니다. z/OS UNIX 자동 마운트에서 디렉토리 소유자로서 클라이언트 사용자 ID를 사용하도록 하려면, 주석 해제한 후 true로 지정하십시오.

참고: z/OS UNIX 자동 마운트는 파일 시스템을 작성할 때 서비스를 호출한 프로세스의 사용자 ID를 사용합니다. 이 옵션이 사용 안함으로 설정되면, 이 프로세스는 STCRSE 사용자 ID를 가진 RSE 스레드 풀 서버입니다. 이 옵션이 사용 가능한 경우, 임시 프로세스는 서비스를 호출하기 전에 클라이언트 사용자 ID를 사용하여 작성됩니다.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Denable.audit.log=true"

감사 옵션. 기본값은 false입니다. 클라이언트가 수행한 조치의 자동 로그를 강제로 실행하려면 주석 해제 후 true를 지정하십시오. 감사 로그는 RSE 디먼 로그 위치에 기록됩니다. 위치를 알려면, _RSE_JAVAOPTS 변수의 daemon.log 옵션을 참조하십시오.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Daudit.cycle=30"

1 감사 로그 파일에 저장되는 일 수. 기본값은 30입니다. 1 감사 로그 파일에 기록되는 감사 데이터의 양을 제어하려면 주석 해제 후 사용자 정의하십시오. 최대값은 365입니다.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Daudit.retention.period=0"

감사 로그를 보관하는 일 수. 기본값은 0이며, 한계가 지정되지 않음을 의미합니다. 지정된 일 수가 지난 후 감사 로그를 삭제하려면 주석 해제 후 사용자 정의하십시오. 최대값은 365입니다.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Daudit.log.mode=RW.R."

감사 로그의 액세스 권한 마스크 기본값은 RW.R.이며, 소유자에게 읽기 및 쓰기 액세스 권한을 허용합니다. 소유자의 기본 그룹에는 읽기 액세스 권한이 있고 기타 사용자에게는 액세스 권한이 없습니다. 필요한 액세스 권한을 설정하려면 주식 해제 후 사용자 정의하십시오.

UNIX 표준에는 3가지 사용자 유형(소유자, 그룹, 기타)에 대한 권한을 설정할 수 있는 것으로 규정되어 있습니다. audit.log.mode 마스크에서 필드는 이 순서와 일치하고, 필드는 점(.)으로 구분됩니다. 각 필드는 비어 있거나, 값으로서 R, W 또는 RW가 있습니다. 여기서 R = 읽기, W = 쓰기.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Daudit.action=<user exit>"

감사 로그 파일을 닫을 때 호출되는 사용자 종료의 이름. 기본값은 없지만 /usr/lpp/rdz/samples/process_audit.rex에서 샘플 종료가 제공됩니다. 감사 로그의 사후 처리를 사용할 수 있으려면, 주식 해제 후 사용자 종료 프로그램의 전체 경로 이름을 지정하십시오.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Daudit.action.id=<userid>"

audit.action 변수에서 지정된 종료를 실행할 때 사용할 사용자 ID. 기본값은 RSE 디먼에 지정된 사용자 ID입니다. 감사 사후 처리 종료를 실행하는 데 지정된 ID를 사용하려면, 주식 해제 후 사용자 ID를 지정하십시오.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Dlogon.action=<user exit>"

사용자가 로그인할 때 호출되는 사용자 종료의 이름. 기본값은 없지만, /usr/lpp/rdz/samples/process_logon.sh에서 샘플 종료가 제공됩니다. 로그인의 사후 처리를 사용할 수 있으려면, 주식 해제 후 사용자 종료 프로그램의 전체 경로 이름을 지정하십시오.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Dlogon.action.id=<userid>"

logon.action 변수에서 지정된 종료를 실행할 때 사용할 사용자 ID. 기본값은 RSE 디먼에 지정된 사용자 ID입니다. 로그인 사후 처리 종료를 실행하는 데 지정된 ID를 사용하려면, 주식 해제 후 사용자 ID를 지정하십시오.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Ddeny.nonzero.port=true"

클라이언트가 통신 포트 번호를 선택하지 못하도록 설정합니다. 기본값은 false입니다. 클라이언트가 연결을 위해 RSE 서버에서 사용해야 하는 호스트 시스템 포트를 지정하는 연결을 거부하려면 주식 해제 후 true를 지정하십시오. 자세한 정보는 53 페이지의 『RSE 서버에 사용 가능한 PORTRANGE 정의』의 내용을 참조하십시오.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Dsingle.logon=false"

한 사용자 ID가 여러 번 로그인하지 못하도록 합니다. 기본값은 true입니다. 하나의 사용자 ID로 단일 RSE 디먼에 여러 번 로그인할 수 있도록 하려면 주식 해제 후 false를 지정하십시오.

참고:

- 이 지시문이 활성화되지 않았거나 true로 설정된 경우 두 번째 로그온을 시도 하면 호스트 시스템이 첫 번째 로그온을 취소합니다. 이 취소 조치는 콘솔 메시지 FEK210I를 동반합니다.
- single.logon 지시문이 활성화되면, minimum.threadpool.process가 1로 설정된 경우라도 2개 이상의 스레드 풀이 시작됩니다. rsed.envvars에서 minimum.threadpool.process의 기본 설정은 1입니다.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Dprocess.cleanup.interval=0"

복구 불가능한 오류 상태의 RSE 스레드 풀을 자동으로 제거합니다. 기본적으로 오류가 있는 RSE 스레드 풀은 자동으로 제거되지 않습니다. 매 간격마다 오류가 있는 RE 스레드 풀 서버를 자동으로 제거하려면(여기서 간격 단위는 초임), 주석 해제 후 사용자 정의하십시오. 0을 지정하면 간격 타이머가 시작되지 않지만, RSE 디먼이 새 클라이언트 로그온 또는 DISPLAY PROCESS 명령 도중 RSE 스레드 풀을 검사할 때 오류가 있는 RSE 스레드 풀 서버가 제거되지 않습니다.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Dreject.logon.threshold=1000000"

지정된 크기보다 큰 파일을 여는 스레드 풀은 파일을 로드할 때까지 새 로그온 요청을 승인하지 않습니다. 기본값 파일 크기는 1000000 바이트입니다. 이러한 파일이 열릴 때 스레드 풀이 로그온 요청을 무시하도록 파일 크기를 지정하려면 주석 해제 후 사용자 정의하십시오. 기타 스레드 풀은 계속 새 로그온 요청을 승인할 수 있습니다.

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

이 변수는 C 코드로 된 콘텐츠 지원에 대한 강제 포함 목록이 들어 있는 완전한 z/OS UNIX 파일을 나타냅니다. 강제 포함은 해당 파일 또는 멤버가 프리프로세서 지시문을 사용하여 소스 코드에 포함되는 지에 관계없이 파일이나 디렉토리, 데이터 세트 또는 콘텐츠 지원 오퍼레이션을 수행할 때 구문 분석되는 데이터 세트 멤버로 구성됩니다. 구성 파일의 이름을 지정하려면 주석 해제 후 사용자 정의하십시오.

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

이 변수는 C++ 코드로 된 콘텐츠 지원에 대한 강제 포함 목록이 들어 있는 완전한 z/OS UNIX 파일을 나타냅니다. 강제 포함은 해당 파일 또는 멤버가 프리프로세서 지시문을 사용하여 소스 코드에 포함되는 지에 관계없이 파일이나 디렉토리, 데이터 세트 또는 콘텐츠 지원 오퍼레이션을 수행할 때 구문 분석되는 데이터 세트 멤버로 구성됩니다. 구성 파일의 이름을 지정하려면 주석 해제 후 사용자 정의하십시오.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS

-DCPP_CLEANUP_INTERVAL=60000"

사용되지 않는 C/C++ 헤더 파일의 정리 간격(밀리초 단위). 기본값은 60000이며, 1분을 의미합니다. 정리 간격을 변경하려면 주석 해제 후 사용자 정의하십시오. 0 값을 지정하면 C/C++ 헤더 파일이 캐싱되지 않으므로, 편집기에서 원격 콘텐츠 지원의 성능이 줄어듭니다.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -DRIS_BUFFER=8"

원격 색인 작성 도중 사용되는 버퍼 크기(메가바이트 단위). 기본값은 8MB입니다. 버퍼 크기를 변경하려면, 주석 해제 후 사용자 정의하십시오. 올바른 값은 1 - 2000(양쪽 포함) 사이의 정수입니다. 버퍼가 크면 색인 작성이 가속화되지만, 스레드 풀의 Java 힙에서 더 많은 부분을 사용합니다. 색인 작성이 완료되기 전 버퍼가 가득 차면 자동으로 색인에 비웁니다.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -DAPPLID=FEKAPPL"

RSE 서버 애플리케이션 ID. 기본값은 FEKAPPL입니다. 필요한 애플리케이션 ID를 강제로 실행하려면 주석 해제 후 사용자 정의하십시오.

참고:

- 애플리케이션 ID는 사용자의 보안 소프트웨어에 정의해야 합니다. 정의하지 못하면 클라이언트가 로그인할 수 없습니다.
- 이 값을 변경할 때 보안 상의 의미는 *호스트 구성 참조서* (SA30-4501)에서 "PassTicket 사용"의 내용을 참조하십시오.
- 이 값은 FEJJCNFG 구성 파일에서 JES 작업 모니터에 대해 설정된 애플리케이션 ID와 일치해야 합니다. 이 값이 다른 경우 RSE가 JES 작업 모니터로 클라이언트를 연결할 수 없습니다. JES 작업 모니터의 변수 정의 방법을 알려면 35 페이지의 『FEJJCNFG, JES 작업 모니터 구성 파일』의 내용을 참조하십시오.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -DRSE_DSICALL=TSO"

데이터 세트 정보를 수집할 때 사용되는 메소드. 기본적으로 서비스에서는 사용자가 실행한 TSO 명령에 대해 사용한 동일한 TSO/ISPF Client Gateway 인프라를 사용합니다. 대안적이고 자원 집약적이지 않은 메소드를 사용하여 데이터 세트 정보를 수집하려면, 주석 해제 후 TSO를 지정하십시오. 이 대체 메소드엔드 추가 필수 소프트웨어가 필요합니다.

- FEK.SFEKLOAD(FEKDSI)는 rsed.envvars에 정의된 LINKLIST 또는 STEPLIB에 있어야 합니다.
- REXX 런타임 라이브러리(REXX.*.SEAGLPA)는 rsed.envvars)에 정의된 LPA, LINKLIST 또는 STEPLIB에 있어야 합니다.

TSO 명령 서비스를 지원하기 위해 ISPF의 TSO/ISPF Client Gateway 대신 APPC를 사용할 경우 이 변수는 사용하지 않습니다.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Dsearch.server.limit.hits=0"

색인이 작성되지 못한 파일과 텍스트 검색의 자원 사용을 한정합니다. 기본값은 0(한계 없음)입니다. 지정한 수의 결과를 찾은 후 검색을 중지하도록 이 지시문을 주석 해제하고 사용자 정의합니다.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Dsearch.server.limit.datasets=0"

색인이 작성되지 못한 파일과 텍스트 검색의 자원 사용을 한정합니다. 기본값은 0(한계 없음)입니다. 지정한 수의 데이터 세트를 스캔 한 후 검색을 중지하도록 이 지시문을 주석 해제하고 사용자 정의합니다.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Dsearch.server.limit.lines=0"

색인이 작성되지 못한 파일과 텍스트 검색의 자원 사용을 한정합니다. 기본값은 0(한계 없음)입니다. 지정한 수의 행을 스캔 한 후 검색을 중지하도록 이 지시문을 주석 해제하고 사용자 정의합니다.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -DDISABLE_TEXT_SEARCH=true"

색인 없는 텍스트 검색을 사용하지 않도록 설정합니다. 기본값은 false입니다. 사용자가 호스트에 전체 텍스트 검색을 시작하지 않도록 하려면, 주석 해제 후 true로 지정하십시오.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS

-DDISABLE_REMOTE_INDEX_SEARCH=true"

클라이언트에서 원격 색인 검색 메뉴 항목을 사용할 수 없습니다. 기본값은 false입니다. 사용자가 호스트 시스템 데이터 세트의 원격 색인을 작성하지 못하도록 하려면, 주석 해제 후 true를 지정하십시오. 이 옵션은 클라이언트 버전 8.5.1 이후에서만 작동합니다.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -DDENY_PASSWORD_SAVE=true"

비밀번호 저장 옵션. 기본값은 false입니다. 사용자가 클라이언트에 호스트 비밀번호를 저장하지 않도록 하려면, 주석 해제 후 true로 지정하십시오. 이전에 저장된 비밀번호는 제거됩니다. 이 옵션은 클라이언트 버전 7.1 이후에서만 작동합니다.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -DHIDE_ZOS_UNIX=true"

z/OS UNIX 옵션을 숨깁니다. 기본값은 false입니다. 사용자에게 z/OS UNIX 요소(디렉토리 구조 및 명령행)가 표시되지 않도록 하려면 클라이언트에서 주석 해제 후 true로 지정하십시오. 이 옵션은 클라이언트 버전 7.6 이후에서만 작동합니다.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS

-DDISABLE_DELETE_IN_SUBPROJECT=true"

z/OS 서브프로젝트의 컨텍스트 메뉴에서 삭제 메뉴 항목을 사용할 수 없도록 합니다. 기본값은 false입니다. 사용자가 z/OS ' 서브프로젝트의 컨텍스트 메뉴에서 삭제 메뉴 항목을 사용할 수 없도록 하려면, 주석 해제 후 true를 지정하십시오. 이 옵션은 클라이언트 버전 8.0.1 이후에서만 작동합니다.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS

-DDSTORE_IDLE_SHUTDOWN_TIMEOUT=3600000"

유틸리티 클라이언트의 연결을 끊습니다. 기본적으로 유틸리티 클라이언트는 연결이 끊어집니다. 표시된 밀리초 동안 유틸리티 상태인 클라이언트의 연결을 끊으려면 주식 해제 후 사용자 정의하십시오. 3600000 밀리초는 1시간입니다.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS

-DDSTORE_SSL_ALGORITHM=TLSv1.2"

클라이언트와의 암호화 통신을 위해 SSL 대신 TLS를 사용합니다. 기본값은 SSL입니다. Developer for System z 클라이언트-호스트 통신에 대해 TLS (v1.2) 사용을 시작하려면 주식 해제 후 TLSv1.2를 지정하십시오. 이 옵션은 Java 버전 7.0 이상 및 클라이언트 버전 9.0 이상에서만 작동합니다. 참고로 버전 9.0 클라이언트도 SSL을 사용합니다. 클라이언트에서 TLS 암호화 통신을 사용하려면 -DDSTORE_SSL_ALGORITHM=TLSv1.2가 eclipse.ini에서 지정되어야 합니다.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -DDSTORE_TCP_NO_DELAY=true"

TCP/IP DELAY ACK 기능을 사용 안함으로 설정합니다. 기본값은 false입니다. TCP/IP가 Developer for System z 클라이언트-호스트 통신에 대해 DELAY-ACK를 수행하지 못하도록 중지하려면 주식 해제 후 true로 지정하십시오.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -DDSTORE_TRACING_ON=true"

dstore 추적을 시작합니다. IBM 지원 센터에서 지시할 경우에만 사용하십시오. 결과 .dstoreTrace 로그 파일은 EBCDIC이 아닌 유니코드(ASCII)로 작성됩니다.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS

-DDSTORE_MEMLOGGING_ON=true"

dstore 메모리 더미 추적을 시작합니다. IBM 지원 센터에서 지시할 경우에만 사용하십시오. 결과 .dstoreMemLogging 로그 파일은 EBCDIC이 아닌 유니코드(ASCII)로 작성됩니다.

_RSE_ISPF_OPTS를 사용한 여분의 Java 시작 매개변수 정의

다른 _RSE_*OPTS 지시문을 사용하는 경우 rsed.envvars에는 RSE 프로세스를 시작할 때 Java에 여분의 매개변수를 제공하는 기능이 있습니다. rsed.envvars에 포함된 샘플 옵션은 주식 해제하여 활성화시킬 수 있습니다.

_RSE_ISPF_OPTS 지시문은 RSE 고유의 Java 옵션이며, 기본적으로 Developer for System z에서 ISPF의 TSO/ISPF Client Gateway를 사용하는 경우에만 적용됩니다.

_RSE_ISPF_OPTS=""

변수 초기화입니다. 수정하지 마십시오.

_RSE_ISPF_OPTS="\$_RSE_ISPF_OPTS&ISPPROF=&SYSUID..ISPPROF"

ISPF 초기화에 기존 ISPF 프로파일을 사용하십시오. 지정된 ISPF 프로파일을 사용하려면 데이터 세트 이름을 주석 해제한 후 변경하십시오.

데이터 세트 이름에서 사용할 수 있는 변수는 다음과 같습니다.

- &SYSUID. - 개발자의 사용자 ID를 대체
- &SYSPREF. - 개발자의 TSO 접두부 또는 사용자 ID(TSO 접두부를 판별할 수 없는 경우) 대체
- &SYSNAME. - IEASYMxx parmlib 멤버에 지정된 대로 시스템 이름을 대체

ISPF.conf, ISPF의 TSO/ISPF Client Gateway 구성 파일

ISPF의 TSO/ISPF Client Gateway는 ISPF.conf의 정의를 사용하여 일괄처리 TSO 및 ISPF 명령을 실행하는 데 올바른 환경을 작성합니다. Developer for System z는 이러한 환경을 사용하여 일부 MVS 기반 서비스를 실행합니다. 이 서비스에는 TSO 명령 서비스와 SCLM 개발자 툴킷이 포함됩니다.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정하지 않은 한, ISPF.conf는 /etc/rdz/에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오. TSO **OEDIT** 명령으로 파일을 편집할 수 있습니다.

정의는 1열에서 시작되어야 합니다. US 코드 페이지를 사용할 때 주석 라인은 별표(*)로 시작합니다. 데이터 라인에는 지시문과 할당값만 있을 수 있습니다. 주석은 같은 라인에서 허용되지 않습니다. 라인 연속은 지원하지 않습니다. 데이터 세트 이름을 연결할 때는 동일 행에서 이름을 추가하고 쉼표(,)로 이름을 구분하십시오.

ISPF 데이터 세트에 대해 올바른 이름을 제공할 뿐만 아니라, 다음 예제에 표시된 대로 TSO 명령 서비스 데이터 세트 이름, FEK.SFEKPROC를 SYSPROC 또는 SYSEXEC 명령문에 추가하십시오.

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

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

그림 12. ISPF.conf: ISPF 구성 파일

참고:

- 자체 DD 유사 명령문 및 데이터 세트 집합을 추가하여 TSO 환경을 사용자 정의함으로써 TSO 로그인 프로시저를 흉내낼 수 있습니다. 자세한 정보는 호스트 구성 참조서 (SA30-4501)에 있는 "TSO 환경 사용자 정의"의 내용을 참조하십시오.
- STEPLIB 지시문을 정의할 수 없습니다. 대신에 rsed.envvars의 STEPLIB 지시문을 사용하십시오.
- **ISPSTART**와 같은 ISPF 명령을 방해하는 써드파티 제품을 사용하는 경우 TSO/ISPF Client Gateway가 올바르게 작동하지 않을 수 있습니다. Developer for System z에 대해 사용 안함으로 설정하는 방법을 알려면 해당 제품의 문서를 확인하십시오. 제품에서 DUMMY에 특정 DD 명령문을 할당해야 하는 경우, 해당 DD 명령문을 nullfile로 할당하여 ISPF.conf에서 이 동작을 시뮬레이션할 수 있습니다.

예를 들어, 다음과 같습니다.

```
ISPTRACE=nullfile
```

- allocjob 지시문을 사용할 때, ISPF.conf보다 먼저 DD 정의를 실행 취소하지 않도록 주의하십시오.
- SMFPRMxx parmlib 멤버에서 JWT 매개변수가 ISPF.conf의 ISPF_timeout 값보다 낮은 값으로 설정되면 모듈 ISPZTS0에 대해 시스템 이상 종료 522가 발생할 것으로 예상됩니다. 필요하다면 TSO/ISPF Client Gateway가 자동으로 재시작되므로 이것은 Developer for System z 오퍼레이션에 영향이 없습니다.
- 변경은 모든 새 호출에 대하여 활성화됩니다. 서버를 다시 시작할 필요가 없습니다.

선택적 컴포넌트

이 절에서는 다양한 선택적 사용자 정의 태스크를 결합합니다. 해당 절의 지시사항에 따라 필요한 서비스를 구성하십시오.

Developer for System z용 Developer 독립형 컴포넌트에 대한 사용자 정의:

- 69 페이지의 제 3 장 『(선택사항) CARMA(Common Access Repository Manager)』
- 111 페이지의 제 4 장 『(선택사항) SCLM 개발자 툴킷』
- 119 페이지의 제 5 장 『(선택사항) 애플리케이션 배치 관리자(더 이상 사용되지 않음)』
- 127 페이지의 제 6 장 『(선택사항) 호스트 기반 코드 분석』

Developer for System z 구성 파일에 대한 사용자 정의:

- 131 페이지의 『(선택사항) pushtoclient.properties, 호스트 기반 클라이언트 제어』
- 135 페이지의 『(선택사항) ssl.properties, RSE SSL 암호화』
- 138 페이지의 『(선택사항) rsecomm.properties, RSE 추적』
- 139 페이지의 『(선택사항) include.conf, C/C++ 콘텐츠 지원을 위해 강제로 포함됨』

기타 제품에 대한 Developer for System z 관련 사용자 정의:

- 140 페이지의 『(선택사항) z/OS UNIX 서브프로젝트』
- 141 페이지의 『(선택사항) 프리프로세서 지원 포함』
- 142 페이지의 『(선택사항) Enterprise COBOL 및 PL/I용 xUnit 지원』
- 144 페이지의 『(선택사항) 엔터프라이즈 서비스 도구 지원』
- 144 페이지의 『(선택사항) CICS 양방향 언어 지원』
- 145 페이지의 『(선택사항) 생성된 코드의 진단 IRZ 메시지』
- 150 페이지의 『(선택사항) 문제점 판별 도구 지원』
- 150 페이지의 『(선택사항) DB2 및 IMS 디버그 지원』
- 151 페이지의 『(선택사항) 파일 관리자 지원』
- 152 페이지의 『(선택사항) WORKAREA 및 /tmp 정리』

설치 검증

IVP의 일부는 선택적 컴포넌트에 대한 것이므로 다양한 설치 검증 프로그램(IVP)에 대한 상세한 설명은 153 페이지의 제 8 장 『설치 검증』에 있습니다.

다음 시나리오를 사용하여 기본 기능을 테스트할 수 있습니다.

1. JMON 시작 태스크 또는 사용자 작업을 시작하십시오. DD STDOUT의 시작 정보는 다음 메시지로 끝나야 합니다.

FEJ211I 서버가
연결을 허용할 준비가 되었습니다.

작업이 리턴 코드 66으로 종료되면, FEK.SFEKAUTH에는 APF 권한이 부여되지 않습니다.

2. IVP=IVP 매개변수를 사용하여 RSED 시작 태스크 또는 사용자 작업을 시작하십시오. 이 매개변수를 사용하면 서버가 몇몇 설치 검증 테스트를 수행한 후 종료됩니다. DD STDOUT에서 다음 IVP가 완료되었음을 나타내는 메시지를 확인하십시오.

- Java 시작
- JES 작업 모니터 연결
- TCP/IP 설정

3. IVP 매개변수를 사용하지 않고 RSED 시작 태스크 또는 사용자 작업 시작. 시작에 성공하면 RSE 디먼이 다음과 같은 콘솔 메시지를 발행합니다.

FEK002I RseDaemon이 시작되었음 (포트=4035)

4. 다음 운영자 명령을 실행한 후 결과 콘솔 메시지에서 테스트가 성공적으로 완료되었음을 확인하십시오.

```
F RSED,APPL=IVP PASSTICKET,userid  
F RSED,APPL=IVP DAEMON,userid  
F RSED,APPL=IVP ISPF,userid
```

userid를 올바른 TSO 사용자 ID로 대체하십시오.

제 3 장 (선택사항) CARMA(Common Access Repository Manager)

CARMA(Common Access Repository Manager)는 RAM(Repository Access Manager)용 서버 플랫폼입니다. RAM은 z/OS 기반인 SCM(Software Configuration Manager)용 API(Application Programming Interface)입니다. RAM에서 SCM 기능을 랩핑하면 클라이언트는 단일 API를 사용하여 지원되는 SCM에 액세스할 수 있습니다.

Developer for System z에서는 자체 RAM 작성을 위해 사전 빌드된 여러 RAM 및 소스 코드 예제를 제공합니다.

호스트 시스템 기반 SCM에서는 서비스에 액세스하는 데 단일 사용자 주소 공간이 필요하며, CARMA가 각 사용자에게 대해 CARMA 서버를 시작해야 합니다. 여러 사용자를 지원하는 단일 서버를 작성할 수 없습니다.

요구사항 및 체크리스트

이 사용자 정의 태스크를 완료하려면 보안 관리자 및 TCP/IP 관리자의 도움이 필요하며, 다음 자원 또는 특수 사용자 정의 태스크가 필요합니다.

- (선택사항) 내부 통신을 위한 TCP/IP 포트 범위
- (선택사항) CARMA VSAM 파일로의 개발자 업데이트 기능을 허용하는 보안 규칙
- (선택사항) 사용자가 CRA* 작업을 제출할 수 있는 보안 규칙
- (선택사항) LPA 업데이트

사용자 사이트에서 CARMA를 사용하여 시작하려면 다음 태스크를 수행하십시오. 특별히 언급되지 않는 한 모든 태스크가 필수입니다.

1. CARMA를 시작하는 메소드를 선택하고 활성화되어야 하는 RAM을 선택하십시오. 사전 구성된 설정대로 RAM 및 서버 시작 메소드의 여러 조합을 사용할 수 있습니다. 자세한 정보는 70 페이지의 『서버 시작 메소드 및 활성화 RAM 선택』의 내용을 참조하십시오.
2. CARMA VSAM 데이터 세트를 작성하십시오. 자세한 정보는 94 페이지의 『CARMA VSAM 데이터 세트』 및 96 페이지의 『CARMA RAM(Repository Access Manager)』의 내용을 참조하십시오.
3. CARMA와의 인터페이스에 대한 RSE 구성 파일의 초기 사용자 정의. 완전한 사용자 정의는 CARMA를 시작하기 위해 선택한 메소드에 따라 달라집니다. 자세한 정보는 84 페이지의 『CRASRV.properties, CARMA에 대한 RSE 인터페이스』의 내용을 참조하십시오.

4. 선택한 CARMA 시작 메소드 및 선택한 RAM에 따라 관련 구성 파일의 필수 사용자 정의를 수행하십시오. 세부사항은 다음을 참조하십시오.
 - 88 페이지의 『crastart*.conf, CRASTART 서버 시작』
 - 92 페이지의 『CRASUB*, 일괄처리 제출 서버 시작』
5. 선택적으로, CA Endeavor® SCM 특정 구성 멤버를 사용자 정의하십시오. 세부사항은 98 페이지의 『CRACFG, CRASHOW 및 CRATMAP: CA Endeavor® SCM RAM 구성 파일』 및 100 페이지의 『CA Endeavor® SCM RAM 일괄처리 조치』의 내용을 참조하십시오.
6. 선택적으로 데이터 세트 할당 exec를 업데이트하십시오. 세부사항은 99 페이지의 『CRANDVRA: CA Endeavor® SCM RAM 할당 exec』, 104 페이지의 『CRAALLOC: 사용자 정의 RAM 할당 exec』 및 106 페이지의 『(선택사항) 사용자 정의 할당 exec』의 내용을 참조하십시오.
7. 선택적으로 시작 사용자 엑시트를 작성하십시오. 세부사항은 (선택사항) CARMA 사용자 엑시트를 참조하십시오.
8. 선택적으로, IRXJCL 대신 CRAXJCL을 작성하십시오. 자세한 정보는 108 페이지의 『(선택사항) IRXJCL 대 CRAXJCL』의 내용을 참조하십시오.

참고: 이 장에서 참조하는 샘플 멤버는 FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의 및 제출할 때 다른 위치를 지정한 경우가 아니면 FEK.#CUST.* 및 /etc/rdz에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오.

서버 시작 메소드 및 활성 RAM 선택

Developer for System z는 CARMA 서버를 시작하기 위한 여러 가지 방법을 지원합니다. Developer for System z는 또한 여러 가지 RAM(Repository Access Manager)을 제공합니다. 이 RAM은 두 그룹(프로덕션 RAM과 샘플 RAM)으로 나눌 수 있습니다. 이 서적에서는 RAM과 서버 시작 메소드의 가능한 여러 조합에 대해 설명합니다. 설명한 구성 시나리오 각각은 사전 구성된 설정으로 사용 가능합니다.

CARMA 서버 시작

Developer for System z는 CARMA 서버를 시작하기 위한 여러 가지 방법을 지원합니다. 각 메소드에는 장점과 단점이 있습니다.

CRASTART

"CRASTART" 메소드는 CARMA 서버를 RSE 내 하위 태스크로 시작합니다. 이 메소드는 CARMA 서버를 시작하는 데 필요한 프로그램 호출과 데이터 세트 할당을 정의하는 개별 구성 파일을 사용하여 매우 유연한 설정을 제공합니다. 이 메소드는 최고의 성능을 제공하고 가장 적은 자원을 사용하지만 CRASTART 모듈이 LPA에 있어야 합니다.

일괄처리 제출

"일괄처리 제출" 방법은 작업을 제출하여 CARMA 서버를 시작합니다. 제공된 샘플 구성 파일에서 사용되는 기본 메소드입니다. 이 방법의 이점은 작업 출력에서 CARMA 로그에 쉽게 액세스할 수 있다는 것입니다. 또한 개발자 자신이 유지관리하는 각 개발자에 대한 사용자 정의 서버 JCL을 사용할 수 있습니다. 그러나 이 메소드에서는 CARMA 서버를 시작하는 개발자당 하나의 JES 이니시에이터를 사용합니다.

(더 이상 사용되지 않음) TSO/ISPF Client Gateway

"TSO/ISPF Client Gateway" 메소드는 ISPF의 TSO/ISPF Client Gateway를 사용하여 TSO 또는 ISPF 환경을 작성하고, 여기서 CARMA 서버가 시작됩니다. 이 메소드는 ISPF.conf의 기능을 이용하여 탄력적으로 데이터 세트를 할당할 수 있습니다. 그러나, 이 메소드는 보통 TSO 또는 ISPF 오퍼레이션을 방해하는 SCM으로 액세스하는 데 적합하지 않습니다.

참고: "TSO/ISPF Client Gateway" 연결 메소드는 더 이상 사용되지 않음으로 표시됩니다. 계속 지원되는 경우라도, 이 기능은 더 이상 개선되지 않고 문서는 Developer for System z 라이브러리(<http://www-01.ibm.com/software/awdtools/rdz/library/>)에서 사용 가능한 *Using ISPF Client Gateway to provide CARMA services* (SC14-7292)로 이동했습니다.

프로덕션 RAM

프로덕션 유형의 RAM은 프로덕션 환경에서 SCM에 액세스할 때 사용할 수 있는 완전히 동작하는 사전 빌드된 RAM입니다.

CA Endeavor® SCM RAM

IBM Rational Developer for System z Interface for CA Endeavor® Software Configuration Manager는 Developer for System z 클라이언트에 CA Endeavor® SCM에 대한 직접 액세스를 제공합니다. 지금부터는 IBM Rational Developer for System z Interface for CA Endeavor® SCM을 CA Endeavor® SCM RAM으로 축약합니다.

CA Endeavor® SCM 패키지 RAM

CA Endeavor® SCM 패키지 RAM은 Developer for System z 클라이언트에게 CA Endeavor® SCM 패키지에 대한 직접 액세스를 제공합니다.

샘플 RAM

샘플 RAM은 사용자 CARMA 환경의 구성을 테스트하기 위한 목적으로 그리고 자체 RAM 개발을 위한 예제로서 제공됩니다. 소스 코드가 포함됩니다.

경고: 프로덕션 환경에서 제공된 샘플 RAM을 사용하지 마십시오.

PDS RAM

PDS RAM은 원격 시스템 보기에 있는 MVS 파일 -> 내 데이터 세트와 유사한 데이터 세트 목록을 제공합니다.

스켈레톤 RAM

스켈레톤 RAM은 자체 RAM 개발을 위한 시작점으로서 사용할 수 있는 기능적 프레임워크를 제공합니다.

SCLM RAM

SCLM RAM은 ISPF의 소프트웨어 구성 관리자인 SCLM의 기초 항목을 제공합니다. 기본적으로 SCLM RAM은 사용할 수 없습니다.

사전 구성된 RAM 및 서버 시작 조합

사전 구성된 설정대로 RAM 및 서버 시작 메소드의 여러 조합을 사용할 수 있습니다. 표시된 시나리오는 사용자 환경에 적합하도록 약간만 사용자 정의하면 됩니다.

- 『CA Endeavor® SCM RAM을 사용한 CRASTART』
- 75 페이지의 『샘플 RAM을 사용한 CRASTART』
- 78 페이지의 『CA Endeavor® SCM RAM을 사용한 일괄처리 제출』
- 81 페이지의 『샘플 RAM을 사용한 일괄처리 제출』

각각 시나리오의 여러 단계에 대한 자세한 정보는 84 페이지의 『CARMA 구성 세부 사항』에 있습니다.

지금 또는 이후에 CARMA 설정으로 RAM을 추가할 수 있습니다. 기존 설정으로 RAM 추가에 대한 자세한 정보는 105 페이지의 『(선택사항) 다중 RAM 자원』의 내용을 참조하십시오.

CA Endeavor® SCM RAM을 사용한 CRASTART

이 절의 정보에서는 다음 스펙으로 CARMA를 설정하는 방법에 대해 설명합니다.

- 서버 시작: CRASTART 방법. 이 방법에서는 CRASTART가 LPA에 있어야 합니다.
- RAM: CA Endeavor® SCM RAM

다른 스펙을 가진 기타 시나리오 중 하나를 사용하려는 경우 이 사용자 정의 단계를 생략할 수 있습니다.

CARMA VSAM 데이터 세트 작성

CARMA 관련 VSAM 데이터 세트를 정의하고 채우려면 다음 JCL 작업을 사용자 정의하고 제출하십시오. 사용자 정의 지시사항은 멤버 내의 문서를 참조하십시오. 기존 VSAM 데이터 세트는 대체됩니다.

이 단계에 대한 자세한 내용은 94 페이지의 『CARMA VSAM 데이터 세트』의 내용을 참조하십시오.

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

CRASRV.properties 사용자 정의

RSE 서버는 /etc/rdz/CRASRV.properties의 설정을 사용하여 CARMA 서버를 시작 및 연결합니다. TSO **OEDIT** 명령으로 파일을 편집할 수 있습니다. 변경된 내용을 적용하려면, RSED 시작 태스크를 다시 시작하십시오.

기본 파일 위치를 사용하는 경우, clist.dsname 지시문 값을 *CRASTART로 바꾸고 crastart.configuration.file 값을 /etc/rdz/crastart.endevor.conf로 바꾸면 됩니다. 다른 지시문에 대한 자세한 정보는 84 페이지의 『CRASRV.properties, CARMA에 대한 RSE 인터페이스』의 내용을 참조하십시오.

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

그림 13. CRASRV.properties: CA Endevor® SCM RAM을 사용하는 CRASTART

crastart.endevor.conf 사용자 정의

CRASTART는 /etc/rdz/crastart.endevor.conf에 있는 정의를 사용하여 CARMA 서버를 시작하는 데 올바른 TSO/ISPF 환경을 작성합니다. TSO **OEDIT** 명령으로 파일을 편집할 수 있습니다. 변경사항은 업데이트 후 시작된 모든 CARMA 서버에 적용됩니다.

사용자 정의 지시사항은 파일 내의 문서를 참조하십시오. CRASTART 시작 메소드에 대한 자세한 정보는 88 페이지의 『crastart*.conf, CRASTART 서버 시작』의 내용을 참조하십시오.

참고: 페이지 너비 제한사항으로 인해, 다음 샘플의 일부 라인은 다음 라인으로 줄바꿈됩니다. 들여쓰기로 시작하는 모든 라인은 이전 라인 끝에 추가되어야 합니다.

```

* RAM에서 사용하는 DD
TYPEMAP = FEK.#CUST.PARMLIB(CRATMAP)
SHOWVIEW= FEK.#CUST.PARMLIB(CRASHOW)
CRACFG = FEK.#CUST.PARMLIB(CRACFG)
* CRABCFG 및 CRABSKEL을 주석 해제하여 일괄처리 조치 사용
*CRABCFG = FEK.#CUST.PARMLIB(CRABCFG)
*CRABSKEL= FEK.#CUST.CNTL
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

* CARMA 서버에서 사용하는 DD(CRASERV)
* TASKLIB 사용 시 APF 권한 부여에 주의
TASKLIB = FEK.SFEKLOAD,CA.NDVR.CSIQAUTH,CA.NDVR.CSIQAUTU
CRADEF = FEK.#CUST.CRADEF
CRAMSG = FEK.#CUST.CRAMSG
CRASTRS = FEK.#CUST.CRASTRS
CARMALOG= SYSOUT(H)
SYSPRINT= SYSOUT(H)

* ISPF에서 사용하는 DD(NDVRC1을 통해)
-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

* TSO에서 사용하는 DD(IKJEFT01)
SYSPROC = FEK.SFEKPROC
SYSTSIN = DUMMY
SYSTSPRT= SYSOUT(H)

* CRANDVRA

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

```

그림 14. *crastart.endevor.conf*: CA Endeavor® SCM RAM 사용하는 CRASTART

(선택사항) 추가적 CA Endeavor® SCM RAM 사용자 정의

CA Endeavor® SCM RAM에는 필요한 경우 사용자 정의할 수 있는 추가 컴포넌트가 있습니다.

- CARMA 시작 처리에는 선택적 사용자 엑시트가 있습니다. 자세한 정보는 (선택사항) CARMA 사용자 엑시트를 참조하십시오.
- CA Endeavor® SCM RAM에는 사용자 정의할 수 있는 여러 구성 파일 (FEK.#CUST.PARMLIB(CRACFG), FEK.#CUST.PARMLIB(CRASHOW) 및 FEK.#CUST.PARMLIB(CRATMAP))이 있습니다. 자세한 정보는 98 페이지의 『CRACFG, CRASHOW 및 CRATMAP: CA Endeavor® SCM RAM 구성 파일』의 내용을 참조하십시오.
- CA Endeavor® SCM RAM에는 사용자 정의할 수 있는 할당 exec(FEK.SFEKPROC (CRANDVRA))가 있습니다. 자세한 정보는 99 페이지의 『CRANDVRA: CA Endeavor® SCM RAM 할당 exec』의 내용을 참조하십시오.
- CA Endeavor® SCM RAM은 일괄처리 모드에서 CA Endeavor® SCM 조치 수행을 지원합니다. 일괄처리 조치에는 사용자 정의해야 하는 하나의 구성 파일 (FEK.#CUST.PARMLIB(CRABCFG)) 및 스켈레톤 JCL(FEK.#CUST.CNTL(CRABATCA))이 필요합니다. 자세한 정보는 100 페이지의 『CA Endeavor® SCM RAM 일괄처리 조치』의 내용을 참조하십시오.

샘플 RAM을 사용한 CRASTART

이 절의 정보에서는 다음 스펙으로 CARMA를 설정하는 방법에 대해 설명합니다.

- 서버 시작: CRASTART 방법. 이 방법에서는 CRASTART가 LPA에 있어야 합니다.
- RAM: 샘플 RAM, 프로덕션 용도에 사용할 수 없습니다.

다른 스펙을 가진 기타 시나리오 중 하나를 사용하려는 경우 이 사용자 정의 단계는 건너뛸 수 있습니다.

CARMA VSAM 데이터 세트 작성

다음 JCL 작업을 사용자 정의 및 제출하여 CARMA 관련 VSAM 데이터 세트를 정의하고 채웁니다. 사용자 정의 지시사항은 멤버 내의 문서를 참조하십시오. 기존 VSAM 데이터 세트는 대체됩니다.

이 단계에 대한 자세한 내용은 94 페이지의 『CARMA VSAM 데이터 세트』 및 96 페이지의 『CARMA RAM(Repository Access Manager)』의 내용을 참조하십시오.

CARMA

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

샘플 RAM

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

CRASRV.properties 사용자 정의

RSE 서버는 /etc/rdz/CRASRV.properties의 설정을 사용하여 CARMA 서버를 시작 및 연결합니다. TSO **OEDIT** 명령으로 파일을 편집할 수 있습니다. 변경된 내용을 적용하려면, RSED 시작 태스크를 다시 시작해야 합니다.

기본 파일 위치를 사용하는 경우, clist.dsname 지시문 값을 *CRASTART로 바꾸기만 하면 됩니다. 다른 지시문에 대한 자세한 정보는 84 페이지의 『CRASRV.properties, CARMA에 대한 RSE 인터페이스』의 내용을 참조하십시오.

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

그림 15. CRASRV.properties: 샘플 RAM을 사용하는 CRASTART

crastart.conf 사용자 정의

CRASTART는 /etc/rdz/crastart.conf의 정의를 사용하여 CARMA 서버를 시작하는 데 올바른 TSO/ISPF 환경을 작성합니다. TSO **OEDIT** 명령으로 파일을 편집할 수 있습니다. 변경사항은 업데이트 후 시작된 모든 CARMA 서버에 적용됩니다.

사용자 정의 지시사항은 파일 내의 문서를 참조하십시오. CRASTART 시작 메소드에 대한 자세한 정보는 88 페이지의 『crastart*.conf, CRASTART 서버 시작』의 내용을 참조하십시오.


```

* RAM에서 사용하는 DD
CRARAM1 = FEK.#CUST.CRARAM1
* CARMA 서버에서 사용하는 DD(CRASERV)
TASKLIB = FEK.SFEKLOAD
CRADEF = FEK.#CUST.CRADEF
CRAMSG = FEK.#CUST.CRAMSG
CRASTRS = FEK.#CUST.CRASTRS
CARMALOG= SYSOUT(H)
SYSPRINT= SYSOUT(H)

* ISPF에서 사용하는 DD(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

* TS0에서 사용하는 DD(IKJEFT01)
SYSPROC = #hlq.SFEKPROC
SYSTSIN = DUMMY
SYSTSPRT= SYSOUT(H)

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

```

그림 16. *crastart.conf*: 샘플 RAM을 사용하는 *CRASTART*

참고: 페이지 폭 제한으로 인해 샘플의 일부 라인은 다음 라인으로 줄바꿉니다. 들여쓰기로 시작하는 모든 라인은 이전 라인 끝에 추가되어야 합니다.

(선택사항) 추가 사용자 설치 RAM의 사용자 정의

사용자 설치 RAM에는 필요한 경우 사용자 정의할 수 있는 추가 컴포넌트가 있습니다.

- CARMA 시작 처리에는 선택적 사용자 엑시트가 있습니다. 자세한 정보는 (선택사항) CARMA 사용자 엑시트를 참조하십시오.
- 사용자 설치 RAM 시작에는 사용자 정의할 수 있는 할당 `exec(FEK.SFEKPROC(CRAALLOC))`가 있습니다. 자세한 정보는 `CRAALLOC`, 사용자 설치 RAM 할당 `exec`를 참조하십시오.

CA Endeavor® SCM RAM을 사용한 일괄처리 제출

이 절의 정보에서는 다음 스펙으로 CARMA를 설정하는 방법에 대해 설명합니다.

- 서버 시작: 일괄처리 제출 방법. 이 방법에서는 JES 이니시에이터가 필요합니다.
- RAM: CA Endeavor® SCM RAM

다른 스펙을 가진 기타 시나리오 중 하나를 사용하려는 경우 이 사용자 정의 단계를 생략할 수 있습니다.

CARMA VSAM 데이터 세트 작성

다음 JCL을 사용자 정의 및 제출하여 CARMA 관련 VSAM 데이터 세트를 정의 및 채웁니다. 사용자 정의 지시사항은 멤버 내의 문서를 참조하십시오. 기존 VSAM 데이터 세트는 대체됩니다.

이 단계에 대한 자세한 내용은 94 페이지의 『CARMA VSAM 데이터 세트』의 내용을 참조하십시오.

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

CRASRV.properties 사용자 정의

RSE 서버는 /etc/rdz/CRASRV.properties의 설정을 사용하여 CARMA 서버를 시작 및 연결합니다. TSO **OEDIT** 명령으로 파일을 편집할 수 있습니다. 변경된 내용을 적용하려면, RSED 시작 태스크를 다시 시작해야 합니다.

기본 파일 위치를 사용하는 경우, clist.dsname 지시문의 값을 FEK.#CUST.CNTL(CRASUBCA)로 바꾸기만 하면 됩니다. 다른 지시문에 대한 자세한 정보는 84 페이지의 『CRASRV.properties, CARMA에 대한 RSE 인터페이스』의 내용을 참조하십시오.

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

그림 17. CRASRV.properties: CA Endeavor® SCM RAM을 사용하는 일괄처리 제출

CRASUBCA 사용자 정의

FEK.#CUST.CNTL(CRASUBCA) CLIST 및 임베디드 JCL은 CARMA 서버를 제출합니다. 변경사항은 업데이트 후 시작된 모든 CARMA 서버에 적용됩니다.

사용자 정의 지시사항은 멤버 내의 문서를 참조하십시오. 일괄처리 제출 시작 메소드에 대한 자세한 정보는 92 페이지의 『CRASUB*, 일괄처리 제출 서버 시작』의 내용을 참

조하십시오.

```

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
//*
//* RAM에서 사용하는 DD
//TYPEMAP DD DISP=SHR,DSN=FEEK.#CUST.PARMLIB(CRATMAP)
//SHOWVIEW DD DISP=SHR,DSN=FEEK.#CUST.PARMLIB(CRASHOW)
//CRACFG DD DISP=SHR,DSN=FEEK.#CUST.PARMLIB(CRACFG)
//* CRABCFG 및 CRABSKEL을 주석 해제하여 일괄처리 조치 사용
//*CRABCFG DD DISP=SHR,DSN=FEEK.#CUST.PARMLIB(CRABCFG)
//*CRABSKEL DD DISP=SHR,DSN=FEEK.#CUST.CNTL
//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))
//C1MSG1 DD SYSOUT(H)
//C1EXMSG1 DD SYSOUT(H)
//MSG3FILE DD DUMMY
//*
//* CARMA 서버에서 사용하는 DD(CRASERV)
//* STEPLIB 사용 시 APF 권한 부여에 주의
//STEPLIB DD DISP=SHR,DSN=FEEK.SFEKLOAD
//          DD DISP=SHR,DSN=CA.NDVR.CSIQAUTH
//          DD DISP=SHR,DSN=CA.NDVR.CSIQAUTU
//CRADEF DD DISP=SHR,DSN=FEEK.#CUST.CRADEF
//CRAMSG DD DISP=SHR,DSN=FEEK.#CUST.CRAMSG
//CRASTRS DD DISP=SHR,DSN=FEEK.#CUST.CRASTRS
//CARMALOG DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//*
//* ISPF에서 사용하는 DD(NDVRC1을 통해)
//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
//*
//* TS0에서 사용하는 DD(IKJEFT01)
//SYSPROC DD DISP=SHR,DSN=FEEK.SFEKPROC * CRANDVRA
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
%CRANDVRA NDVRC1 PGM(CRASERV) PARM(&CRAPRM1 &CRAPRM2 &CRAPRM3 +
&CRAPRM4 &CRAPRM5 &CRAPRM6 &CRAPRM7 &CRAPRM8 )
$$
EXIT CODE(0)

```

그림 18. CRASUBCA: CA Endeavor® SCM RAM을 사용한 일괄처리 제출

(선택사항) 추가적 CA Endeavor® SCM RAM 사용자 정의

CA Endeavor® SCM RAM에는 필요한 경우 사용자 정의할 수 있는 추가 컴포넌트가 있습니다.

- CA Endeavor® SCM RAM에는 사용자 정의할 수 있는 여러 구성 파일 (FEK.#CUST.PARMLIB(CRACFG), FEK.#CUST.PARMLIB(CRASHOW) 및 FEK.#CUST.PARMLIB(CRATMAP))이 있습니다. 자세한 정보는 98 페이지의 『CRACFG, CRASHOW 및 CRATMAP: CA Endeavor® SCM RAM 구성 파일』의 내용을 참조하십시오.
- CA Endeavor® SCM RAM에는 사용자 정의할 수 있는 할당 exec(FEK.SFEKPROC (CRANDVRA))가 있습니다. 자세한 정보는 99 페이지의 『CRANDVRA: CA Endeavor® SCM RAM 할당 exec』의 내용을 참조하십시오.
- CA Endeavor® SCM RAM은 일괄처리 모드에서 CA Endeavor® SCM 조치 수행을 지원합니다. 일괄처리 조치에는 사용자 정의해야 하는 구성 파일(FEK.#CUST.PARMLIB (CRABCFG)), 스켈레톤 JCL(FEK.#CUST.CNTL(CRABATCA)) 및 선택적 기본 작업 카드(FEK.#CUST.CNTL(CRABJOBC))가 필요합니다. 자세한 정보는 100 페이지의 『CA Endeavor® SCM RAM 일괄처리 조치』의 내용을 참조하십시오.

샘플 RAM을 사용한 일괄처리 제출

이 절의 정보에서는 다음 스펙으로 CARMA를 설정하는 방법에 대해 설명합니다.

- 서버 시작: 일괄처리 제출 방법, 여기에서는 JES 이니시에이터가 필요합니다.
- RAM: 샘플 RAM이며, 프로덕션 용도에 사용할 수 없습니다.

다른 스펙을 가진 기타 시나리오 중 하나를 사용하려는 경우 이 사용자 정의 단계를 생략할 수 있습니다.

VSAM 데이터 세트 작성

다음 JCL 작업을 사용자 정의 및 제출하여 CARMA 관련 VSAM 데이터 세트를 정의하고 채웁니다. 사용자 정의 지시사항은 멤버 내의 문서를 참조하십시오. 기존 VSAM 데이터 세트는 대체됩니다.

이 단계에 대한 자세한 내용은 94 페이지의 『CARMA VSAM 데이터 세트』 및 96 페이지의 『CARMA RAM(Repository Access Manager)』의 내용을 참조하십시오.

CARMA

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

샘플 RAM

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

CRASRV.properties 사용자 정의

RSE 서버는 /etc/rdz/CRASRV.properties의 설정을 사용하여 CARMA 서버를 시작 및 연결합니다. TSO **OEDIT** 명령으로 파일을 편집할 수 있습니다. 변경된 내용을 적용하려면, RSED 시작 태스크를 다시 시작해야 합니다.

Developer for System z의 기본 시나리오이므로, 수정되지 않은 파일 사본에서 시작할 때 아무것도 변경할 필요가 없습니다. 다른 지시문에 대한 자세한 정보는 84 페이지의 『CRASRV.properties, CARMA에 대한 RSE 인터페이스』의 내용을 참조하십시오.

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

그림 19. CRASRV.properties: 샘플 RAM을 사용하는 일괄처리 제출

CRASUBMT 사용자 정의

FEK.#CUST.CNTL(CRASUBMT) CLIST 및 임베디드 JCL은 CARMA 서버를 제출합니다. 변경사항은 업데이트 후 시작된 모든 CARMA 서버에 적용됩니다.

사용자 정의 지시사항은 멤버 내의 문서를 참조하십시오. 일괄처리 제출 시작 메소드에 대한 자세한 정보는 92 페이지의 『CRASUB*, 일괄처리 제출 서버 시작』의 내용을 참조하십시오.

```

PROC 8 CRAPRM1 CRAPRM2 CRAPRM3 CRAPRM4 CRAPRM5 CRAPRM6 CRAPRM7 CRAPRM8
SUBMIT * END($$)
//CRA&PORT JOB CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1)
//*
//RUN      EXEC PGM=IKJEFT01,DYNAMNBR=125,REGION=0M,TIME=NOLIMIT
//*
//* RAM에서 사용하는 DD
//CRARAM1 DD DISP=SHR,DSN=FEEK.#CUST.CRARAM1          * PDS RAM
//*
//* CARMA 서버에서 사용하는 DD(CRASERV)
//STEPLIB DD DISP=SHR,DSN=FEEK.SFEKLOAD
//CRADEF  DD DISP=SHR,DSN=FEEK.#CUST.CRADEF
//CRAMSG  DD DISP=SHR,DSN=FEEK.#CUST.CRAMSG
//CRASTRS DD DISP=SHR,DSN=FEEK.#CUST.CRASTRS
//CARMALOG DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//*
//* ISPF에서 사용하는 DD(ISPSTART)
//ISPPROF DD DISP=(NEW,DELETE,DELETE),UNIT=SYSALLDA,
//          LRECL=80,RECFM=FB,SPACE=(TRK,(1,1,5))
//ISPCTL0 DD DISP=(NEW,DELETE,DELETE),UNIT=SYSALLDA,
//          LRECL=80,RECFM=FB,SPACE=(TRK,(5,5))
//ISPCTL1 DD DISP=(NEW,DELETE,DELETE),UNIT=SYSALLDA,
//          LRECL=80,RECFM=FB,SPACE=(TRK,(5,5))
//ISPMLIB DD DISP=SHR,DSN=ISP.SISPMENU
//ISPPLIB DD DISP=SHR,DSN=ISP.SISPPENU
//ISPSLIB DD DISP=SHR,DSN=ISP.SISPSENU
//ISPTLIB DD DISP=SHR,DSN=ISP.SISPTENU
//*
//* TS0에서 사용하는 DD(IKJEFT01)
//SYSPROC DD DISP=SHR,DSN=#hlq.SFEKPROC          * CRAALLOC
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
%CRAALLOC ISPSTART PGM(CRASERV) PARM(&CRAPRM1 &CRAPRM2 &CRAPRM3 +
                                     &CRAPRM4 &CRAPRM5 &CRAPRM6 &CRAPRM7 &CRAPRM8 )
$$
EXIT CODE(0)

```

그림 20. CRASUBMT: 샘플 RAM을 사용하는 일괄처리 제출

(선택사항) 추가 사용자 설치 RAM의 사용자 정의

사용자 설치 RAM에는 필요한 경우 사용자 정의할 수 있는 추가 컴포넌트가 있습니다.

- CARMA 시작 처리에는 선택적 사용자 엑시트가 있습니다. 자세한 정보는 (선택사항) CARMA 사용자 엑시트를 참조하십시오.
- 사용자 설치 RAM 시작에는 사용자 정의할 수 있는 할당 exec(FEEK.SFEKPROC (CRAALLOC))가 있습니다. 자세한 정보는 CRAALLOC, 사용자 설치 RAM 할당 exec를 참조하십시오.

CARMA 구성 세부사항

본 서적에서 설명하는 다른 구성 시나리오에서는 많은 CARMA 구성 파일을 공유합니다. 이러한 구성 파일의 세부사항에 대해서는 여기에서 문서화하고 여러 시나리오에서 참조합니다.

CRASRV.properties, CARMA에 대한 RSE 인터페이스

CARMA 서버는 호스트 시스템을 사용하여 하나 이상의 SCM(Software Configuration Manager)에 액세스하는 기타 제품을 위한 표준 API를 제공합니다. 그러나, 클라이언트 컴퓨터와의 직접 통신을 위한 메소드는 제공하지 않습니다. 이 통신의 경우 RSE 서버와 같은 기타 제품을 사용합니다. RSE 서버는 CRASRV.properties의 설정을 사용하여 CARMA 서버를 시작 및 연결합니다.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정하지 않는 한, CRASRV.properties는 /etc/rdz/에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오. TSO **OEDIT** 명령으로 파일을 편집할 수 있습니다.

참고: 변경된 내용을 적용하려면, RSED 시작 태스크를 다시 시작해야 합니다.

```
# CRASRV.properties - CARMA 구성 옵션
#
port.start=0
#port.range=100
#user.exit='FEK.SFEKSAMP(CRAEXIT)'
startup.script.name=carma.startup.rex
clist.dsname='FEK.#CUST.CNTL(CRASUBMT)'
crastart.configuration.file=crastart.conf
#crastart.stub=/usr/lpp/rdz/bin/CRASTART
#crastart.syslog=Partial
#crastart.timeout=420
#crastart.steplib=FEK.SFEKLPA
#crastart.tasklib=TASKLIB
```

그림 21. CRASRV.properties - CARMA 구성 파일

port.start

port.start 값이 0(0)일 때, CARMA는 CARMA와 RSE 서버 사이의 통신에 임시 포트를 사용합니다. 이 시나리오에서, TCP/IP는 사용하지 않는 임의 포트 번호를 지정합니다. port.start 값이 0이 아닐 경우 CARMA와 RSE 서버 사이의 통신에 사용되는 포트 범위의 시작점으로 해석됩니다. 이 경우 port.range 변수도 정의해야 합니다. 기본 포트는 0입니다. 이 포트에 대한 통신은 사용자 호스트 시스템에만 한정됩니다.

참고: 포트를 선택하기 전에, 포트가 **NETSTAT** 및 **NETSTAT PORTL** 명령을 이용하여 사용자 시스템에서 이용 가능한지 확인하십시오. 자세한 정보는 호스트 구성 참조서 (SA30-4501)의 "예약된 TCP/IP 포트"의 내용을 참조하십시오.

#port.range

port.start가 0이 아닐 경우 CARMA 통신에 사용되는 포트 범위로서 port.start에서 시작합니다. 기본값은 100입니다. 포트 범위의 크기를 지정하려면 주석 해제한 후 사용자 정의하십시오. 예를 들어, port.start가 5227이고 port.range가 100일 때, 포트 5227부터 5326까지(양쪽 포함) CARMA에서 사용할 수 있습니다. 각 CARMA 연결에서는 포트를 독점적으로 사용하므로, 포트 범위를 지정하면 동시 CARMA 세션의 최대 수가 한정됩니다.

#user.exit

CARMA 시작 중에 실행할 사용자 지정 코드를 정의합니다. 실행할 코드의 데이터 세트 이름을 주석 해제하고 지정하십시오.

따옴표(')를 사용하는 데이터 세트 이름은 절대 참조이고, 따옴표(')가 없는 데이터 세트 이름은 TSO 접두부가 아니라 클라이언트의 사용자 ID가 접두부입니다. 후자의 경우 모든 CARMA 사용자가 자신의 종료 코드를 유지보수해야 합니다.

샘플 사용자 엑시트는 SFEKSAMP(CRAEXIT)로 제공됩니다. 또한 이 샘플은 사용자 엑시트로 전달된 시작 인수를 설명합니다. 자세한 정보는 (선택사항) CARMA 사용자 엑시트를 참조하십시오.

startup.script.name

CARMA 시작 스크립트를 정의합니다. 기본값은 carma.startup.rex 입니다. 이 REXX exec는 CARMA 서버의 시작을 트리거합니다. 파일 이름은 여러 방법으로 지정될 수 있습니다.

- 널 문자열. 변수가 지정되지 않았음을 의미합니다. 이 경우에, 기본값이 사용됩니다.
- 파일 이름만. 이것이 기본 방법입니다. CARMA는 파일을 찾기 위해 PATH 환경 변수에 있는 디렉토리를 검색합니다. Developer for System z 실행 파일(기본값은 /usr/lpp/rdz/bin)을 보유하는 디렉토리는 PATH 환경 변수에 자동으로 추가됩니다.
- 상대 경로. 디렉토리 및 파일 이름이며 앞에 슬래시(/)가 없습니다. CARMA는 제공된 경로에 구성 디렉토리(기본값은 /etc/rdz/)를 추가하여 절대 경로를 작성합니다.
- 절대 경로. 디렉토리 및 파일 이름이며 앞에 슬래시(/)가 있습니다. CARMA는 지정된 파일 위치를 사용합니다.

clist.dsname

CARMA 서버의 시작 방법을 정의합니다. 다른 시작 방법에 대한 자세한 정보는 70 페이지의 『서버 시작 메소드 및 활성화 RAM 선택』의 내용을 참조하십시오.

- *CRASTART는 CARMA 서버가 CRASTART를 사용하여 RSE 내부의 하위 태스크로서 시작되어야 함을 표시합니다. *CRASTART를 지정하면 crastart.* 지시문도 지정하거나 기본값을 사용해야 합니다.
- *ISPF는 CARMA 서버가 ISPF의 TSO/ISPF Client Gateway를 사용하여 시작되어야 함을 표시합니다. 이 시작 방법은 더 이상 사용되지 않습니다.
- 기타 값은 TSO와 유사한 이름 지정 규칙을 사용하여 CRASUBMT CLIST의 위치를 정의합니다. 작은따옴표(')를 사용하는 데이터 세트 이름은 절대 참조이고, 작은따옴표(')가 없는 데이터 세트 이름은 TSO 접두부가 아니라 클라이언트의 사용자 ID가 접두부입니다. 후자의 경우 모든 CARMA 사용자가 자신의 CRASUBMT CLIST를 유지보수해야 합니다.

기본값은 'FEK.#CUST.CNTL(CRASUBMT)'입니다. 일괄처리 제출 방법을 이용하여 연결을 개설하면 이 CLIST가 CARMA 서버를 시작합니다.

crastart.configuration.file

CRASTART 구성 파일의 이름을 지정합니다. 기본값은 crastart.conf입니다. 이 파일은 CARMA 서버를 시작할 때 필요한 데이터 세트 할당 및 프로그램 호출을 지정합니다. clist.dsname 지시문에 *CRASTART 값이 있는 경우에만 이 지시문을 사용합니다. 파일 이름은 여러 방법으로 지정될 수 있습니다.

- 널 문자열. 변수가 지정되지 않았음을 의미합니다. 기본값이 사용됩니다.
- 파일 이름만. 이것이 기본 방법입니다. CARMA는 파일을 찾기 위해 사용자 구성 디렉토리(기본값은 /etc/rdz)를 검색합니다.
- 상대 경로. 디렉토리와 파일 이름이며 앞에 슬래시(/)가 없습니다. CARMA는 제공된 경로에 구성 디렉토리(기본값은 /etc/rdz/)를 추가하여 절대 경로를 작성합니다.
- 절대 경로. 디렉토리와 파일 이름이며 앞에 슬래시(/)가 있습니다. CARMA는 지정된 파일 위치를 사용합니다.

#crastart.stub

SRASTART 호출을 위한 z/OS UNIX 스텝. 기본값은 CRASTART입니다. 이 스텝은 z/OS UNIX 프로세스에서 MVS 기반 CRASTART 로드 모듈을 이용 가능하도록 합니다. 특정 경로를 지정하려면 주석을 해제하고 사용자 정의하십시오. clist.dsname 지시문에 *CRASTART 값이 있는 경우에만 이 지시문을 사용합니다. 파일 이름은 여러 방법으로 지정될 수 있습니다.

- 널 문자열. 변수가 지정되지 않았음을 의미합니다. 기본값이 사용됩니다.

- 파일 이름만. 이것이 기본 방법입니다. CARMA는 파일을 찾기 위해 PATH 환경 변수에 있는 디렉토리를 검색합니다. Developer for System z 실행 파일(기본값은 /usr/lpp/rdz/bin)을 보유하는 디렉토리는 PATH 환경 변수에 자동으로 추가됩니다.
- 상대 경로. 디렉토리와 파일 이름이며 앞에 슬래시(/)가 없습니다. CARMA는 제공된 경로에 구성 디렉토리(기본값은 /etc/rdz/)를 추가하여 절대 경로를 작성합니다.
- 절대 경로. 디렉토리와 파일 이름이며 앞에 슬래시(/)가 있습니다. CARMA는 지정된 파일 위치를 사용합니다.

#crastart.syslog

CRASTART가 CARMA 서버를 시작하는 동안 시스템 로그에 기록되는 정보의 양을 지정합니다. 기본값은 Partial입니다. 올바른 값은 다음 표에 나열되어 있습니다.

A(모두)	모든 추적 정보가 SYSLOG에 기록됩니다.
P(일부)	연결, 연결 끊기 및 오류 정보만 SYSLOG에 기록됩니다.
기타	오류 조건만 SYSLOG에 기록됩니다.

시스템 로그 메시지의 필수 상세 레벨을 지정하려면 주석을 해제하고 사용자 정의하십시오. clist.dsname 지시문에 *CRASTART 값이 있는 경우에만 이 지시문을 사용합니다.

#crastart.timeout

활동 부족으로 인해 CARMA 서버가 종료한 전 시간(초)입니다. 기본값은 420(7분)입니다. 필요한 제한시간 값을 지정하려면 주석을 해제하고 사용자 정의하십시오. clist.dsname 지시문에 *CRASTART 값이 있는 경우에만 이 지시문을 사용합니다.

참고: SMFPRMxx parmlib 멤버에서 JWT 매개변수가 CRASRV.properties의 crastart.timeout 값보다 낮은 값으로 설정되면 모듈 CRASERV에 대해 시스템 이상 종료 522가 발생합니다. 필요한 경우 서버가 자동으로 재시작되므로 이 발생은 CARMA 오퍼레이션에 영향이 없습니다.

#crastart.steplib

rsed.envvars의 STEPLIB 지시문을 통해 액세스할 때 CRASTART 모듈의 위치. 기본값은 FEK.SFEKLPA입니다. CRASTART 모듈이 LPA 또는 LINKLIST의 일부가 될 수 없는 경우 이 지시문의 주석을 해제하고 사용자 정의하십시오. CRASTART 모듈이 LPA에 없으면 프로그램 제어 및 APF 문제가 발생할 수 있습니다. clist.dsname 지시문에 *CRASTART 값이 있는 경우에만 이 지시문을 사용합니다.

#crastart.tasklib

crastart.conf의 TASKLIB DD 이름에 대한 대체 이름입니다. 기본값은 TASKLIB입니다. DD 이름 TASKLIB가 SCM 또는 RAM에 대해 특별한 의미가 있고 STEPLIB 대체로서 사용할 수 없는 경우, 이 지시문을 주석 해제하고 사용자 정의하십시오. clist.dsnname 지시문에 *CRASTART 값이 있는 경우에만 이 지시문을 사용합니다.

crastart*.conf, CRAFTSTART 서버 시작

RSE는 CRAFTSTART 로드 모듈을 시작하고, 여기에서는 crastart*.conf에 있는 정의를 사용하여 일괄처리 TSO 및 ISPF 명령을 실행하는 데 올바른 환경을 작성합니다. Developer for System z는 이 환경을 사용하여 CARMA 서버, CRAFTSERV를 실행합니다.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 crastart*.conf는 /etc/rdz/에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오. TSO **OEDIT** 명령으로 파일을 편집할 수 있습니다.

참고: 변경사항은 업데이트 후 시작된 모든 CARMA 서버에 적용됩니다.

Developer for System z에서는 여러 crastart*.conf 구성 파일을 제공합니다. 이러한 샘플 파일 각각은 특정 사용자 정의 시나리오를 위해 미리 구성됩니다.

- crastart.endevor.conf는 CA Endevor® SCM RAM을 사용한 CRAFTSTART 시작에 맞도록 구성됩니다.
- crastart.conf는 샘플 RAM을 사용한 CRAFTSTART 시작에 맞도록 구성됩니다.

crastart*.conf 파일의 기능은 JCL 작업 스트림의 개념과 비슷하지만, 더 제한적입니다.

- 다음과 같은 샘플은 유효한 라인 형식을 보여줍니다.

```
- * comment
- ddname=dsn1,dsn2,dsn3          * comment
- ddname=SYSOUT(c)                * comment
- ddname=DUMMY                    * comment
- -COMMAND=<any bpxwdyn command> * comment
- PROGRAM = progname parms        * comment
```

참고: **BXPWDYN** 명령은 *Using REXX and z/OS UNIX System Services* (SA22-7806)에서 설명하며 복합 할당 생성이 가능합니다.

- 모든 입력은 대문자로 변경됩니다.
- 라인 연속은 지원하지 않습니다.

- 한 라인의 길이 제한은 없습니다.
- 등호(=) 주위에는 하나 이상의 공백이 허용됩니다.
- DD 할당은 관련된 PROGRAM 명령문보다 선행해야 합니다.
- 여기에서 할당된 DD 이름은 프로그램 실행 종료시 해제됩니다. 누적되지 않습니다.
- 피호출 프로그램에서 할당한 DD 이름은 해제되지 않습니다.
- 여러 데이터 세트가 DD 이름에 연결될 수 있습니다. 데이터 세트 이름은 쉼표(,)로 구분되어야 하고 표시된 순서대로 접합을 검색합니다.
- 모든 데이터 세트 할당은 -COMMAND를 사용하여 수행된 할당을 제외하고, DISP=SHR로 수행됩니다.
- 인라인 데이터는 지원되지 않습니다. 모든 데이터는 카탈로그 파일에 있어야 합니다.
- 변수는 등호(=) 오른쪽에서만 사용될 수 있습니다.
- 다음 변수가 지원됩니다.

&CRAUSER.	클라이언트 사용자 ID
&CRADATE.	Dyyyyddd 형식(7자 율리우스)의 Current® 날짜
&CRATIME.	Thhmmss 형식(시 분 초)의 현재 시간
&CRAPRM1.	포트 번호
&CRAPRM2.	제한시간 값
시스템 기호	SYS1.PARMLIB(IEASYMxx) 시스템 기호
-<ddname>	하이픈 뒤에 오는 이전에 정의된 DD 이름은 JCL의 *.ddname 후방 참조처럼 작용합니다. 원래 DD는 -COMMAND 명령문을 사용하여 할당해야 합니다.

참고: 구성 파일을 해석할 때 TSO가 활성화되지 않으므로 TSO 접두부에 대한 변수는 없습니다. 사용 불가능한 기타 변수 또는 TSO 접두부가 필요한 경우, 106 페이지의 『(선택사항) 사용자 정의 할당 exec』의 내용을 참조하십시오.

90 페이지의 그림 22에서는 ISPF 서비스를 포함하는 기본 crastart*.conf 스케레톤을 보여줍니다.

* RAM에서 사용하는 DD

* CARMA 서버에서 사용하는 DD(CRASERV)

```
TASKLIB = FEK.SFEKLOAD
CRADEF = FEK.#CUST.CRADEF
CRAMSG = FEK.#CUST.CRAMSG
CRASTRS = FEK.#CUST.CRASTRS
CARMALOG= SYSOUT(H)
SYSPRINT= SYSOUT(H)
```

* ISPF에서 사용하는 DD(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
```

* TSO에서 사용하는 DD(IKJEFT01)

```
SYSPROC = #hlq.SFEKPROC * CRAALLOC
SYSTSIN = DUMMY
SYSTSPRT= SYSOUT(H)
```

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

그림 22. *crastart*.conf*: *CRASTART*를 사용하는 *CARMA* 서버 시작

참고:

- 페이지 폭 제한으로 인해 샘플의 일부 라인은 다음 라인으로 줄바꿈니다. 들여쓰기로 시작하는 모든 라인은 이전 라인 끝에 추가되어야 합니다.
- **PROGRAM** 행을 변경하는 경우, 문자열 처리를 단순화하기 위해 **PARM()** 명령문의 닫기 괄호(")") 앞에 최소 하나의 공백이 있는지 확인하십시오.
- 자체 DD 명령문 및 데이터 세트 접합을 추가하여 **CARMA TSO** 환경을 사용자 정의할 수 있으므로, TSO 로그인 프로시저와 유사합니다.
- DD 이름 **TASKLIB**는 JCL의 **STEPLIB**와 유사하게 동작합니다. DD 이름은 **CRASRV.properties**에서 **crastart.tasklib**에 대해 지정된 값과 일치해야 하며, 여기에 대해서는 84 페이지의 『**CRASRV.properties**, **CARMA**에 대한 RSE 인터페이스』에서 설명합니다.
- 정규 APF 규칙은 **TASKLIB** 할당에 적용됩니다. APF의 허가를 받지 않은 라이브러리가 접합의 일부인 경우 라이브러리도 APF의 권한을 잃게 됩니다.
- **SMFPRMxx parmlib** 멤버의 **JWT** 매개변수가 **CRASRV.properties**의 **crastart.timeout** 값보다 낮은 값으로 설정되면 모듈 **CRASERV**에 대해 시스템 이

상 종료 522가 발생합니다. 필요한 경우 서버가 자동으로 다시 시작되므로 시스템 이상 종료는 CARMA 오퍼레이션에 영향이 없습니다.

- 서버가 끝날 때 CARMA 서버 시작에 대한 세부사항은 rsecomm.log에 표시됩니다. rsecomm.log의 상세 레벨 설정에 대한 자세한 정보는 138 페이지의 『(선택사항) rsecomm.properties, RSE 추적』의 내용을 참조하십시오.

CRASTART 로그 파일 수집

CRASTART는 RSE의 하위 프로세스로서 TSO 환경을 작성하며, 별도의 주소 공간에서 실행됩니다. 사소한 조치에서는 SYSOUT(*)에 보내진 CARMA 결과물을 유지할 필요가 없으며, 로그 파일 수집이 복잡해집니다. 이 어려움은 다음 샘플 할당에 표시한 대로 사용자 특정 데이터 세트에 로그 파일을 기록하면 해결할 수 있습니다.

```
-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)
```

참고:

- 페이지 폭 제한으로 인해 샘플의 일부 라인은 다음 라인으로 줄바꿈됩니다. 들여쓰기로 시작하는 모든 라인은 이전 라인 끝에 추가되어야 합니다.
- 사용자 특정 로그 파일을 작성할 수 있도록 이 로그 파일은 -COMMAND 명령문을 사용하여 할당되어야 합니다.
- 더 많은 유연성이 필요한 경우 할당 exec에 로그 데이터 세트를 할당할 수도 있습니다. 예를 들어, 특정 사용자의 데이터 세트에 로그를 전송하는 경우. 할당 exec에 대한 자세한 정보는 106 페이지의 『(선택사항) 사용자 정의 할당 exec』의 내용을 참조하십시오.

SYSOUT에 로그 파일을 기록하는 경우, z/OS UNIX 프로세스가 할당하는 SYSOUT는 JES에서 특수 출력으로 처리됨을 기억하십시오. 이것은 APPC 트랜잭션에서 할당하는 SYSOUT과 유사합니다.

- CARMA 서버가 계속 활동하는 동안, SDSF에서 DA 명령을 사용하여 출력을 볼 수 있습니다. 작업에는 작업 이름 및 STC 작업 ID로서 뒤에 임의의 1자리 숫자가 오는 사용자의 사용자 ID가 있습니다. 사용자는 작업 소유자입니다.
- 출력이 HOLD 출력 클래스로 기록된 경우, CARMA 서버가 끝날 때 연결을 종료하는 사용자 또는 비활성으로 인해 출력은 SDSF에서 APPC ON 및 HALL 명령을 사용하여 출력을 볼 수 있습니다. 작업 이름, 작업 ID 및 작업 소유자는 그대로 유지됩니다. 각 DD에서는 별개의 스푼 파일로 표시되며, DD임을 나타내는 표시는 없습니다.
- SEARCHALL=ON이 FEJJCNFG에서 활성화되고 출력이 HOLD 출력 클래스의 스푼에 상주하는 경우 JES 작업 모니터도 출력을 표시합니다. SEARCHALL 지시문에 대한 자세한 정보는 35 페이지의 『FEJJCNFG, JES 작업 모니터 구성 파일』의 내용을 참조하십시오.

CRASUB*, 일괄처리 제출 서버 시작

RSE는 CLIST CRASUB*를 시작하고, 그 결과 임베디드 JCL을 제출하여 일괄처리 TSO 및 ISPF 명령을 실행하기에 올바른 환경을 작성합니다. Developer for System z는 이 환경을 사용하여 CARMA 서버, CRASERV를 실행합니다.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 CRASUB*는 FEK.#CUST.CNTL에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오.

참고: 변경사항은 업데이트 후 시작된 모든 CARMA 서버에 적용됩니다.

Developer for System z에서는 여러 CRASUB* JCL 작업을 제공합니다. 이러한 샘플 파일 각각은 특정 사용자 정의 시나리오를 위해 미리 구성됩니다.

- CRASUBCA는 CA Endevor® SCM RAM을 사용한 일괄처리 시작에 맞게 구성됩니다.
- CRASUBMT는 샘플 RAM을 사용한 일괄처리 시작에 맞게 구성됩니다.

93 페이지의 그림 23에서는 ISPF 서비스를 포함하는 기본 CRASUB* 스켈레톤을 표시합니다.


```

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

```

그림 23. CRASUB*: 일괄처리 제출을 사용하는 CARMA 시작

참고:

- SYSTSIN 데이터를 변경하는 경우, 문자열 처리를 단순화하기 위해 PARM() 명령문의 닫기 괄호(")") 앞에 최소 하나의 공백이 있는지 확인하십시오.
- 자체 DD 명령문 및 데이터 세트 접합을 추가하여 CARMA TSO 환경을 사용자 정의할 수 있으므로, TSO 로그온 프로시저와 유사합니다.
- 선택적으로 CRASUB* CLIST에서 the SET CRAPRM2=420 행을 주석 해제하고 수정하여 CARMA의 제한시간 값을 변경할 수 있습니다. 제한시간 값은 CARMA가 클라이언트로부터 다음 명령을 기다리는 시간(초)입니다. 0 값을 설정하면 기본 제한시간 값이 현재 420초(7분)이 됩니다.
- 서버가 끝날 때 CARMA 시작 프로세스에 대한 세부사항은 rsecomm.log에 표시됩니다. rsecomm.log의 상세 레벨 설정에 대한 자세한 정보는 138 페이지의 『(선택사항) rsecomm.properties, RSE 추적』의 내용을 참조하십시오.

CARMA VSAM 데이터 세트

CARMA 서버에는 세 VSAM 데이터 세트에 대한 읽기 액세스 권한이 필요합니다. FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 이 VSAM 데이터 세트를 작성하고 채우는 샘플 멤버는 FEK.#CUST.JCL에 위치합니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오.

참고:

- (사용자 정의) RAM의 정의를 기존 VSAM 구성과 병합해야 하는 경우, FEK.#CUST.JCL(CRA#UADD) 샘플 작업을 참조하십시오. 이 작업은 변하는 각 CARMA VSAM 파일에 맞게 사용자 정의되고 제출되어야 합니다. 다른 CARMA VSAM 파일을 사용하는 레코드 구조에 대한 자세한 정보는 *Common Access Repository Manager Developer's Guide*(SC23-7660)를 참조하십시오.
- VSAM에서 순차 데이터 세트로 활성 정의를 추출하려면 FEK.#CUST.JCL(CRA#UQRY) 샘플 작업을 사용하십시오.

CRADEF, 구성 데이터 세트

이 VSAM 데이터 세트는 정의된 RAM에서 지원하는 기능에 대해 설명합니다. RAM 개발자는 이 데이터 세트에 대한 업데이트 액세스가 필요합니다. 데이터 세트는 이러한 샘플 작업 중 하나에서 작성될 수 있습니다.

- CRA\$VCAD는 CA Endevor® SCM RAM 데이터로 데이터 세트를 채웁니다.
- CRA\$VDEF는 샘플 RAM 데이터로 데이터 세트를 채웁니다.

언급된 샘플 작업을 사용하여 VSAM 작성 도중 정의된 RAM을 사용 안함으로 설정합니다. 그렇게 설정하면 단일 마스터 입력 파일을 이용하여 사용자 정의한 CARMA 설정을 작성할 수 있으며, 이것은 IBM에서 제공하거나 RAM 개발자가 사용자 정의하는 설정일 수 있습니다.

CRAMSG, 메시지 데이터 세트

이 VSAM 데이터 세트는 CARMA 서버 자체에서 실행한 메시지를 보유하고 있습니다. 데이터 세트는 이러한 샘플 작업 중 하나에서 작성될 수 있습니다.

- CRA\$VMSG는 일반 서버 데이터로 데이터 세트를 채웁니다.

CRASTRS, 사용자 정의 문자열 데이터 세트

이 VSAM 데이터 세트는 정의된 RAM에서 실행된 메시지를 보유하고 있습니다. RAM 개발자는 이 데이터 세트에 대한 업데이트 액세스가 필요합니다. 데이터 세트는 이러한 샘플 작업 중 하나에서 작성될 수 있습니다.

- CRA\$VCAS는 CA Endevor® SCM RAM 데이터로 데이터 세트를 채웁니다.
- CRA\$VSTR은 샘플 RAM 데이터로 데이터 세트를 채웁니다.

CARMA VSAM 마이그레이션 노트

1. 버전 7.6.1을 시작으로, Developer for System z에서는 메시지 길이 제한을 제거하기 위해 CARMA 사용자 정의 정보 VSAM 데이터 세트(CRASTRS)의 새 데이터 구조 레이아웃을 지원합니다.

Developer for System z 버전 7.6.1 이전에는 CARMA 사용자 정의 정보 VSAM 데이터 세트에서 정의된 문자열이 사전 정의된 길이로 제한됩니다. 이 제한은 RAM 개발자에게 기술적 문자열을 줄이거나 클라이언트측 플러그인을 사용하여 전체 길이 문자열을 표시하도록 합니다.

새 VSAM 레코드 구조는 CARMA 사용자 정의 정보 VSAM 데이터 세트(CRASTRS)에 대한 가변-길이 데이터 구조 레이아웃을 지원하며, 여기서 문자열은 고정 길이 대신에 구분 문자로 구분됩니다.

새로운 가변 길이 형식으로 기존의 고정 길이, CARMA 사용자 정의 정보 VSAM 데이터 세트(CRASTRS)를 변환하도록 FEK.SFEKSAMP(CRA#VS2) JCL을 사용자 정의하고 제출하십시오.

참고:

- 버전 7.6.1부터 샘플 CARMA 사용자 정의 정보 VSAM 데이터 세트는 가변 길이 형식으로 제공됩니다.
 - 버전 7.6.1부터 CARMA 로드 모듈(CRASERV)은 CARMA 사용자 정의 정보 VSAM 데이터 세트에 대해 고정 길이 형식 및 가변 길이 형식 모두를 지원합니다.
 - 가변 길이 CARMA 사용자 정의 정보 VSAM 데이터 세트와 함께 사용될 경우 CARMA 로드 모듈의 이번 버전에서는 가변 길이 형식을 지원하고 왜곡된 문자열을 생산하지 않습니다.
2. CA Endevor® SCM 패키지 RAM이 도입되면서 CA Endevor® SCM 요소 RAM에서 패키지 관련 조치를 제공할 필요가 없어졌습니다. 그러므로 CA Endevor® SCM RAM에서는 더 이상 패키지 관련 조치를 제공하지 않으며 버전 8.5에서 CARMA VSAM 데이터 세트는 이 기능에 필요한 정의를 더 이상 보유하지 않습니다.

CARMA VSAM 데이터 세트로 제거된 항목을 다시 병합하도록 FEK.SFEKSAMP(CRA#UADD) JCL을 사용자 정의하고 제출하여 이 기능을 복원할 수 있습니다.

- FEK.SFEKVSM2(CRA0VPKD)를 FEK.#CUST.CRADEF로 병합하십시오.
- FEK.SFEKVSM2(CRA0VPKS)를 FEK.#CUST.CRASTRS로 병합하십시오.

제품 유지보수 동안 VSAM 데이터 세트를 업데이트할 때마다 이 병합 조치가 필요합니다.

CARMA RAM(Repository Access Manager)

RAM은 z/OS 기반인 SCM(Software Configuration Manager)의 API(Application Programming Interface)입니다. 결국 Developer for System z 또는 사용자 작성 애플리케이션은 RAM을 로드하고 지원되는 SCM에 액세스하기 위한 표준 인터페이스를 제공하는 CARMA 서버를 시작할 수 있습니다.

CARMA 서버는 LINKLIST 또는 STEPLIB/TASKLIB를 통해 RAM 로드 모듈을 찾을 수 있어야 합니다.

Developer for System z의 SMP/E 설치 도중 다른 상위 레벨 규정자를 사용하지 않으면 Developer for System z에서 제공하는 CRAR* RAM 로드 모듈은 FEK.SFEKLOAD에 위치하며, 샘플 소스 코드 및 컴파일 작업은 FEK.SFEKSAMP에 위치합니다.

다음 절에는 Developer for System z와 함께 사용 가능한 RAM의 사용자 정의 노트가 있습니다. FEK.SFEKSAMP(FEKSETUP) 샘플 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 참조되는 샘플 멤버는 FEK.#CUST.*에 위치합니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오.

CARMA에 대한 깊이 있는 지식 및 제공되는 샘플 소스 코드와 샘플 RAM에 대한 자세한 정보는 *Common Access Repository Manager Developer's Guide*(SC23-7660)를 참조하십시오.

CA Endeavor® SCM RAM

- CA Endeavor® SCM RAM은 프로덕션 유형의 RAM입니다.
- CA Endeavor® SCM RAM은 Developer for System z 클라이언트에게 CA Endeavor® SCM 요소에 대한 직접 액세스를 제공합니다.
- 로드 모듈 이름은 CRARNDVR입니다.
- CA Endeavor® SCM RAM에는 기존 CARMA 설정과 비교하여 많은 추가적 설정이 있습니다. 시작점으로서 CA Endeavor® SCM RAM을 지원하는 사전 구성된 설정 중 하나를 이용하고, 필요한 대로 사용자 정의하십시오.
- 더 이상 사용되지 않는 TSO/ISPF Client Gateway 시작 메소드는 CA Endeavor® SCM RAM과 함께 사용될 수 없습니다.
- CA Endeavor® SCM RAM에는 사용자 정의할 수 있는 여러 구성 파일(FEK.#CUST.PARMLIB(CRACFG), FEK.#CUST.PARMLIB(CRASHOW) 및 FEK.#CUST.PARMLIB(CRATMAP))이 있습니다. 자세한 정보는 98 페이지의 『CRACFG, CRASHOW 및 CRATMAP: CA Endeavor® SCM RAM 구성 파일』의 내용을 참조하십시오.
- CA Endeavor® SCM RAM에는 사용자 정의할 수 있는 할당 exec(FEK.SFEKPROC(CRANDVRA))가 있습니다. 자세한 정보는 99 페이지의 『CRANDVRA: CA Endeavor® SCM RAM 할당 exec』의 내용을 참조하십시오.

- CA Endeavor® SCM RAM은 일괄처리 모드에서 백그라운드로 CA Endeavor® SCM 조치 수행을 지원합니다. 일괄처리 조치에는 사용자 정의해야 하는 하나의 구성 파일(FEK.#CUST.PARMLIB(CRABCFG)) 및 스켈레톤 JCL(FEK.#CUST.CNTL(CRABATCA))이 필요합니다. 자세한 정보는 100 페이지의 『CA Endeavor® SCM RAM 일괄처리 조치』의 내용을 참조하십시오.

CA Endeavor® SCM 패키지 RAM

- CA Endeavor® SCM 패키지 RAM은 프로덕션 유형의 RAM입니다.
- CA Endeavor® SCM 패키지 RAM은 Developer for System z 클라이언트에 CA Endeavor® SCM 패키지에 대한 직접 액세스를 제공합니다.
- 로드 모듈 이름은 CRARPKGS입니다.
- CA Endeavor® SCM 패키지 RAM에는 사용자 정의할 수 있는 설정이 없으므로 CA Endeavor® SCM RAM과 함께 사용되어야 합니다.

PDS RAM

- PDS RAM은 샘플 RAM입니다. 프로덕션 환경에서는 사용하지 마십시오.
- PDS RAM은 원격 시스템 보기에 있는 MVS 파일 -> 내 데이터 세트와 유사한 데이터 세트 목록을 제공합니다.
- 로드 모듈 이름은 CRARPDS입니다.
- PDS RAM에서는 ISPF 서비스를 사용할 수 있어야 합니다.
- PDS RAM에서는 추가 VSAM 데이터 세트를 DD CRARAM1에 할당해야 합니다. 이 VSAM 데이터 세트는 할당되어 FEK.#CUST.JCL(CRA#VPDS) 샘플 작업을 준비할 수 있습니다. 사용자 정의 지시사항은 멤버 내의 문서를 참조하십시오.
- 소스 코드와 컴파일 작업은 FEK.SFEKSAMP에서 이용할 수 있습니다. 자세한 정보는 *Common Access Repository Manager Developer's Guide*(SC23-7660)를 참조하십시오.

스켈레톤 RAM

- 스켈레톤 RAM은 샘플 RAM입니다. 프로덕션 환경에서는 사용하지 마십시오.
- 스켈레톤 RAM은 자체 RAM 개발을 위한 시작점으로서 사용할 수 있는 기능적 프레임워크를 제공합니다.
- 로드 모듈 이름은 CRARTEST입니다.
- 소스 코드와 컴파일 작업은 FEK.SFEKSAMP에서 이용할 수 있습니다. 자세한 정보는 *Common Access Repository Manager Developer's Guide*(SC23-7660)를 참조하십시오.

SCLM RAM

- SCLM RAM은 샘플 RAM입니다. 프로덕션 환경에서는 사용하지 마십시오.

- SCLM RAM은 ISPF의 소프트웨어 구성 관리자인 SCLM의 기초 항목을 제공합니다. 기본적으로 이 RAM은 사용할 수 없습니다.
- 로드 모듈 이름은 CRARSCLM입니다.
- SCLM RAM에서는 ISPF 서비스를 사용할 수 있어야 합니다.
- SCLM RAM에서는 추가 VSAM 데이터 세트가 DD CRARAM2에 할당되어야 합니다. 이 VSAM 데이터 세트가 할당되어 FEK.#CUST.JCL(CRA#VSLM) 샘플 작업을 준비할 수 있습니다. 사용자 정의 지시사항은 멤버 내의 문서를 참조하십시오.
- SCLM RAM에는 다양한 사용자 고유의 데이터 세트가 존재해야 합니다. FEK.#CUST.JCL(CRA#ASLM)을 사용자 정의하여 이러한 데이터 세트를 할당하십시오. 사용자 정의 지시사항은 멤버 내의 문서를 참조하십시오. 각 사용자는 SCLM RAM이 있는 CARMA를 사용하기 전 한 번 CRA#ASLM을 제출해야 합니다. 그렇지 않으면 할당 오류가 발생합니다.
- 기본적으로 SCLM RAM은 사용할 수 없습니다. RAM을 사용할 수 있으려면, DD CRADEF 및 CRASTRS에서 참조하는 CARMA VSAM 데이터 세트에 정의되어야 합니다. FEK.#CUST.JCL(CRA#UADD) 샘플 작업을 사용하여 FEK.SFEKVSM2(CRA0SLMD)를 CRADEF에, FEK.SFEKVSM2(CRA0SLMS)를 CRASTRS에 병합하십시오. 사용자 정의 지시사항은 멤버 내의 문서를 참조하십시오.
- 소스 코드와 컴파일 작업은 FEK.SFEKSAMP에서 이용할 수 있습니다. 자세한 정보는 *Common Access Repository Manager Developer's Guide(SC23-7660)*를 참조하십시오.

CRACFG, CRASHOW 및 CRATMAP: CA Endeavor® SCM RAM 구성 파일

다음 CA Endeavor® SCM RAM 특정 CARMA 컴포넌트는 선택된 서버 시작 방법과 관계없이 사용자 정의될 수 있습니다. FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 아래 참조되는 샘플 멤버는 FEK.#CUST.PARMLIB에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오.

CRACFG: SCM과 CA Endeavor® SCM RAM의 상호작용

CRACFG는 CA Endeavor® SCM RAM이 CA Endeavor® SCM과 상호작용하는 방법을 지정합니다. 기본값을 변경하려는 경우 사용자 정의 지시사항에 대해서는 멤버 내의 문서를 참조하십시오.

```
# ENTRY-STAGE-COPY-MODE = RETRIEVE-ADD
```

그림 24. CRACFG: SCM과 CA Endeavor® SCM RAM의 상호작용

CRASHOW, CA Endeavor® SCM RAM 기본 필터

CRASHOW는 CA Endeavor® SCM 환경, 시스템 등을 위한 기본 필터를 정의합니다. 기본값을 변경하려는 경우 사용자 정의 지시사항에 대해서는 멤버 내의 문서를 참조하십시오.

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

그림 25. CRASHOW - CA Endeavor® SCM RAM 기본 필터

참고: FILTER-DEP는 일반 CA Endeavor® SCM 변수가 아니지만, 풋프린트가 있는 요소에 대한 종속성 스캔을 제어하는 Developer for System z 고유 변수는 기타 CA Endeavor® SCM 저장소 위치를 참조합니다.

CRATMAP, CA Endeavor® SCM RAM 파일 확장자 맵핑

CRATMAP는 CA Endeavor® SCM 유형을 파일 확장자 맵핑에 대체합니다. 기본값으로 변경하려는 경우 멤버 내 문서에서 사용자 정의 지시사항을 참조하십시오.

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

그림 26. CRATMAP: CA Endeavor® SCM RAM 기본 필터

CRANDVRA: CA Endeavor® SCM RAM 할당 exec

일괄처리 제출과 CRASTART 시작 방법 모두는 CRANDVRA REXX exec를 호출하여 CA Endeavor® SCM RAM에 사용되는 사용자 고유 데이터 세트를 할당합니다. exec는 일괄처리 제출 CRASUBCA JCL과 CRASTART crastart.endevor.conf 구성 파일 내에서 가능한 그 이상의 유연성을 제공하므로 할당은 별도의 exec에서 완료됩니다. 할당 exec는 선택적 사용자 엑시트 호출도 담당합니다.

DD	데이터 세트 이름	유형
DEPEND	&SYSPREF..&SYSUID.. &SYSNAME..CRA\$NDVR.DEPEND	영구적

DD	데이터 세트 이름	유형
BROWSE	&SYSPREF..&SYSUID.. &SYSNAME..CRA\$NDVR.BROWSE	임시
CIPRINT	&SYSPREF..&SYSUID.. &SYSNAME..CRA\$NDVR.LISTING	임시
SPCLLIST	&SYSPREF..&SYSUID.. &SYSNAME..CRA\$NDVR.SPCLLIST	임시
PKGSCLS	&SYSPREF..&SYSUID.. &SYSNAME..CRA\$NDVR.PKGSCLS	임시
CRABJCLO	&SYSPREF..&SYSUID.. &SYSNAME..CRA\$NDVR.CRABJCLO	임시
ENHCEDIT	&SYSPREF..&SYSUID.. &SYSNAME..CRA\$NDVR.ENHCEDIT	임시
CRAPARM	&SYSPREF..&SYSUID.. &SYSNAME..CRA\$NDVR.CRAPARM	임시

데이터 세트 이름과 같은 특정 기본값이 사용자의 사이트 표준과 일치하지 않으면 이 할당 REXX exec 사본을 사용자 정의할 수 있습니다. Developer for System z의 SMP/E 설치 도중 다른 상위 레벨 규정자를 사용한 경우가 아니면 CRANDVRA는 FEK.SFEKPROC에 위치합니다.

사용자 정의 지시사항은 멤버 내의 문서를 참조하십시오. 할당 exec에 대한 자세한 정보는 106 페이지의 『(선택사항) 사용자 정의 할당 exec』의 내용을 참조하십시오.

참고: 샘플 할당 REXX를 새 데이터 세트로 복사하고 유지보수를 적용할 때 겹쳐 쓰여지지 않도록 이 사본을 사용자 정의해야 합니다. 이를 위해서는 새 데이터 세트 이름과 일치하도록 사용자가 선택한 CARMA 시작 방법의 SYSEXEC DD에서 SFEKPROC에 대한 참조를 업데이트해야 합니다.

CA Endevor® SCM RAM 일괄처리 조치

일반적으로 “요소 생성”과 같은 CA Endevor® SCM 조치는 CARMA 서버 주소 공간에서 “온라인”으로 실행됩니다. TSO는 이미 활성화되어 있으므로 즉 SYSTSIN 및 SYSTSPRT와 같이 필요한 DD를 사용 중임을 의미하므로 CA Endevor® SCM 프로시저가 TSO를 호출하면 이 동작으로 인해 문제점이 발생합니다.

이 문제를 해결하기 위해 CA Endevor® SCM RAM은 버전 8.0.3부터 “일괄처리 조치”를 지원합니다. 일괄처리 조치가 사용되면, CA Endevor® SCM RAM은 사용자 정의할 수 있는 일괄처리 작업을 제출하여 “요소 생성”과 같은 조치를 수행합니다. 일괄처리 작업을 사용하면 제출된 JCL에서 TSO를 활성화할 필요가 없으므로 CA Endevor® SCM 프로시저에서 SYSTSIN 및 SYSTSPRT와 같은 DD를 할당합니다.

CA Endevor® SCM RAM 일괄처리 조치는 백그라운드 CA Endevor® SCM 조치와 동일한 Developer for System z입니다.

일괄처리 조치에서 지원하는 조치를 실행하는 요청을 실행하면, CA Endeavor® SCM RAM은 CRASUBCA 또는 crastart.endevor.conf에서 CRABCFG DD가 있는지 확인한 후 이 DD에 대한 설정이 올바른지 확인합니다. CRABCFG가 존재하고 설정이 올바르면, 조치는 일괄처리로 수행됩니다. CRABCFG가 없으면 조치는 온라인으로 수행됩니다. 버전 8.0.3 이후 클라이언트에는 이 동작을 대체하는 기능이 있습니다.

예를 들어, 다음과 같습니다.

```
//* CRABCFG 및 CRABSKEL을 주석 해제하여 일괄처리 조치 사용
//*CRABCFG DD DISP=SHR,DSN=FEK.#CUST.PARMLIB(CRABCFG)
//*CRABSKEL DD DISP=SHR,DSN=FEK.#CUST.CNTL
```

참고:

- TSO 없는 환경은 선택된 CA Endeavor® SCM 조치에 대해서만 사용할 수 있습니다. 일괄처리 조치는 이 범위를 벗어나는 TSO 없는 환경을 지원하지 않습니다.
- CRABCFG 구성 파일은 지원되는 CA Endeavor® SCM 조치에 대해 설명합니다.
- 일괄처리 조치를 실행하기 위한 기능적 샘플 작업 FEK.#CUST.CNTL(CRABATCA)이 제공되지만, 일괄처리 조치의 목적은 현재 CA Endeavor® SCM 프로시저를 시작할 수 있도록 이 샘플을 사용자 정의하는 것입니다.
- 일괄처리 조치 JCL을 제출할 때 사용되는 클래스에서 사용할 수 있는 충분한 JES 이니시에이터가 있는지 확인하십시오.
- SYSPLEX 환경에서 JES를 사용 중인 경우, 작업이 현재 시스템에서 실행되는지 또는 CA Endeavor® SCM RAM이 상태를 확인할 수 있도록 Developer for System z를 호스트하는 시스템으로 완료 정보가 다시 라우트되는지 확인하십시오.
- Developer for System z와 호스트 시스템 모두가 버전 8.5.1 이상이면, 클라이언트는 제출하기 전에 일괄처리 조치 JCL에 사용자 정의된 JOB 카드와 추가 JCL 명령문을 제공할 수 있습니다.

CRABCFG, CA Endeavor® SCM RAM 일괄처리 조치 구성

CRABCFG는 CA Endeavor® SCM RAM 일괄처리 조치와 연관된 구성 변수를 정의합니다.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 CRABCFG는 FEK.#CUST.PARMLIB에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오.

다음 CRABCFG 샘플 파일을 참조하십시오. 사용자의 시스템 환경에 맞도록 사용자 정의되어야 합니다. US 코드 페이지를 사용할 때 주석 행은 숫자 부호(#)로 시작합니다. 지시문과 할당값 뒤에 있는 주석은 지원됩니다. 등호(=) 주위에 있는 공백도 지원됩니다. 라인 연속은 지원하지 않습니다.

참고: 변경사항은 업데이트 후 시작된 모든 CARMA 서버에 적용됩니다.

```
# 일괄처리 조치 JCL의 위치
SKELETON-DD = CRABSKEL
#
# SKELETON-DD 내에서의 일괄처리 조치 JCL 멤버
DEFAULT-JOBCARD = CRABJOBC
ADD-ELEMENT     = CRABATCA
GENERATE-ELEMENT = CRABATCA
MOVE-ELEMENT    = CRABATCA
DELETE-ELEMENT  = CRABATCA
RETRIEVE-ELEMENT = CRABATCA
SIGNIN-ELEMENT  = CRABATCA
PRINT-ELEMENT   = CRABATCA
PRINT-MEMBER    = CRABATCA
#
# 일괄처리 조치 JCL 내에서의 명령 대체 키
BSTIPT01-KEY = <CRA_BSTIPT01>
```

그림 27. CRABCFG: CA Endeavor® SCM RAM 일괄처리 조치 구성

SKELETON-DD

일괄처리 조치 스켈레톤 JCL을 보유하고 있는 하나 이상의 PDS(E) 데이터 세트를 참조하는 DD 명령문의 이름입니다. 샘플 값은 CRABSKEL입니다. 필요한 경우 변경할 수 있습니다. 이 DD는 CRASUBCA 또는 crastart.endevor.conf에서 CARMA 서버에 대해 정의해야 합니다.

DEFAULT-JOBCARD

기본 JOB 카드를 보유하고 있는 멤버의 이름입니다. 버전 8.5.1 이상의 Developer for System z 클라이언트에 저장된 사용자 고유의 JOB 카드에서 반대하지 않으면, 이 기본 JOB 카드를 사용하여 스켈레톤 JCL에서 <JOBCARD> 키를 대체합니다. 필요한 경우 변경할 수 있습니다.

GENERATE-ELEMENT 및 기타 CA Endeavor® SCM 조치

키 이름은 일괄처리 조치에서 지원하고, 변경할 수 없는 CA Endeavor® SCM 조치를 나타냅니다. 각 키에 지정된 값은 관련 스켈레톤 JCL의 멤버명입니다. 샘플 값은 모든 키에 대해 CRABATCA입니다. 필요한 경우 변경할 수 있습니다.

BSTIPT01-KEY

실제 CA Endeavor® SCM 명령 문자열의 대체 키입니다. 샘플 값은 <CRA_BSTIPT01>입니다. 필요한 경우 변경할 수 있습니다. 스켈레톤 JCL에 있는 이 대체 키의 첫 번째 발생(주석에는 없음)은 CA Endeavor® SCM에 요청된 요소에 대해 요청된 조치를 수행하도록 알리는 명령 문자열로 대체됩니다.

CRABATCA, CA Endeavor® SCM RAM 일괄처리 조치 JCL

CRABATCA는 일괄처리 조치에 사용되는 샘플 스켈레톤 JCL입니다. 기본값을 변경하려면 멤버 내 문서에서 사용자 정의 지시사항을 참조하십시오.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 CRABATCA는 FEK.#CUST.CNTL에 위치합니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오.

변경은 모든 새 호출에 대하여 활성화됩니다. 서버를 다시 시작할 필요가 없습니다.

```
//<JOB CARD>
//*
//CRABATCA EXEC PGM=NDVRC1,DYNAMNBR=1500,REGION=4096K,PARM='C1BM3000'
//STEPLIB DD DISP=SHR,DSN=CA.NDVRU.AUTHLIB * NDVR R12
// DD DISP=SHR,DSN=CA.NDVR.AUTHLIB * NDVR R12
//* DD DISP=SHR,DSN=CA.NDVR.CSIQAUTU * NDVR R14
//* DD DISP=SHR,DSN=CA.NDVR.CSIQAUTH * NDVR R14
//CONLIB DD DISP=SHR,DSN=CA.NDVR.CONLIB * NDVR R12
//*CONLIB DD DISP=SHR,DSN=CA.NDVR.CSIQLOAD * NDVR R14
//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>
//*
```

그림 28. CRABATCA: CA Endeavor® SCM RAM 일괄처리 조치 JCL

CRABJOBC, CA Endeavor® SCM RAM 일괄처리 조치 JOB 카드

CRABJOBC는 <JOB CARD> 키를 지정하는 일괄처리 조치 스퀘레톤 JCL에 사용된 샘플 기본 JOB 카드입니다. 기본값을 변경하려면 멤버 내 문서에서 사용자 정의 지시사항을 참조하십시오.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 CRABJOBC는 FEK.#CUST.CNTL에 위치합니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오.

변경은 모든 새 호출에 대하여 활성화됩니다. 서버를 다시 시작할 필요가 없습니다.

```
//<USERID>B JOB CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1)
//*PROCS JCLLIB ORDER=(COBOL.V4R1M0.SIGYPROC,CBC.SCCNPRC)
```

그림 29. CRABJOBC: CA Endeavor® SCM RAM 일괄처리 조치 JOB 카드

CRAALLOC: 사용자 정의 RAM 할당 exec

일괄처리 제출 및 CRASTART 시작 메소드는 둘 다 CRAALLOC REXX exec를 호출하여 사용자 작성 RAM에서 사용할 수 있는 사용자별 데이터 세트를 할당합니다. exec는 일괄처리 제출 CRASUBMT JCL 및 CRASTART crastart.conf 구성 파일 내에서 가능한 그 이상의 유연성을 허용하므로 할당은 별도의 exec에서 완료됩니다. 할당 exec는 선택적 사용자 엑시트 호출도 담당합니다.

DD	데이터 세트 이름	유형
CRAPARM	&SYSPREF..&SYSUID.. &SYSNAME..CRA\$CUST.CRAPARM	임시

데이터 세트 이름과 같은 특정 기본값이 사용자의 사이트 표준과 일치하지 않으면 이 할당 REXX exec 사본을 사용자 정의할 수 있습니다. Developer for System z의 SMP/E 설치 도중 다른 상위 레벨 규정자를 사용한 경우가 아니면 CRAALLOC는 FEK.SFEKPROC에 위치합니다.

사용자 정의 지시사항은 멤버 내의 문서를 참조하십시오. 할당 exec에 대한 자세한 정보는 (선택사항) 사용자 정의 할당 exec를 참조하십시오.

참고: 샘플 할당 REXX를 새 데이터 세트로 복사하고 유지보수를 적용할 때 겹쳐 쓰여지지 않도록 이 사본을 사용자 정의해야 합니다. 이를 위해서는 새 데이터 세트 이름과 일치하도록 사용자가 선택한 CARMA 시작 방법의 SYSEXEC DD에서 SFEKPROC에 대한 참조를 업데이트해야 합니다.

CARMA 리턴 코드

CARMA는 다양한 오류 코드를 클라이언트에 또는 호스트 시스템 로그에 보고할 수 있습니다. 오류와 함께 제공되는 세부사항 및 표 12의 정보는 오류를 찾아 해결하는 데 도움이 될 수 있습니다.

표 12. CARMA 리턴 코드

오류 범위	오류 유형
4-99	일반 CARMA 오류
100-199	일반 RAM 오류
200-399	CRASERV(CARMA 서버) 오류
400-499	RSE(CARMA 마이너) 오류
500-899	RAM 특정 오류
900-999	TSO 및 TCP/IP 오류

몇 가지 공통 리턴 코드는 다음과 같습니다.

- 220: CARMA 서버가 비활동 제한시간으로 인해 종료합니다. 이것은 오류가 아닙니다.

- 990: CARMA 서버가 CARMA 마이너가 청취하고 있는 포트에 연결될 수 없습니다.

(선택사항) 다중 RAM 지원

CARMA에는 다중 RAM을 정의하고 이들을 동시에 실행하는 기능이 있습니다. 그러나 한 사용자에게 대해 하나의 CARMA 서버만 활성화할 수 있으므로, 여러 RAM이 있는 경우라도 이 설정을 작동시키려면 일부 구성을 변경해야 합니다.

RAM은 CARMA 구성 VSAM 데이터 세트, CRADEF에서 RAM 개발자가 정의합니다. 시작 도중, CARMA 서버, CRASERV는 정의된 모든 RAM을 식별하고 정보를 CRMA 클라이언트로 전송합니다. 그러면 사용자가 하나 이상의 RAM을 선택할 수 있고, 이것이 CARMA 서버로 로드됩니다.

RAM은 CARMA 서버의 플러그인으로서 활성화되므로, CARMA 서버의 주소 공간에서 각 RAM에 대해 데이터 세트 할당과 같은 모든 필수 소프트웨어를 사용할 수 있는지 확인하십시오. 이 요구사항으로 인해 Developer for System z에 포함된 CRASUBMT 또는 crastart.conf와 같은 CARMA 구성 샘플을 변경해야 합니다.

예제

다음 예제에서는 CRASTART 시작 메소드를 사용하여 CA Endeavor® SCM RAM을 사용하는 기존 설정에서 시작하고 샘플 PDS RAM을 추가합니다.

CA Endeavor® SCM RAM에 대한 정의:

- FEK.SFEKVSM2(CRA0VCAD): CRADEF 정의
- FEK.SFEKVSM2(CRA0VCAS): CRASTRS 정의
- /etc/rdz/crastart.endevor.conf: CRASTART 구성 파일

PDS RAM을 위한 정의:

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

프로세스는 RAM 개발자가 시스템 프로그래머가 설정을 완료하는 데 필요한 데이터 및 정보를 수집하는 것으로 시작합니다.

1. SFEKVSM2 멤버에서 PDS RAM에만 해당하는 데이터를 추출하십시오. 이러한 멤버는 PDS RAM뿐만 아니라 모든 샘플 RAM에 대한 정의를 보유합니다.
2. CA Endeavor® SCM RAM SFEKVSM2 멤버와 이 데이터를 병합하십시오.
3. PDS RAM 특정 필수 소프트웨어 목록을 작성하십시오.
 - DD CRARAM1, FEK.#CUST.CRARAM1에 링크됩니다.

- TSO 환경

시스템 프로그래머는 이 데이터를 사용하여 업데이트된 CARMA VSAM 데이터 세트를 작성하고 이 필수 소프트웨어 정보를 사용하여 RAM 모드를 지원할 수 있는 CRASTART 구성 파일을 작성합니다.

1. CRA\$VDEF 및 CRA\$VSTR 작업에 대한 입력으로서 결합된 데이터를 사용하여 업데이트된 CARMA 구성 및 사용자 정의 정보 VSAM 데이터 세트(CRADEF 및 CRASTRS)를 작성합니다. CRAMSG VSAM은 CARMA 서버에만 해당되므로 RAM 모드에서 동일합니다.

2. crastart.endevor.conf에 CRARAM1 정의를 추가하십시오.

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

3. crastart.endevor.conf의 PROGRAM 명령문을 보고 두 RAM에 필요한 환경을 제공할 수 있는지 확인하십시오.

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

- IKJEFT01: TSO, 비승인 환경에서 승인된 특정 호출을 허용하는 데 사용되고, CA Endevor® SCM RAM 사전 할당 exec를 실행하는 환경으로 사용됩니다.
- %CRANDVRA: CA Endevor® SCM RAM 사전 할당 exec, FEK.SFEKPROC에 있으며 임시 사용자 특정 작업 데이터 세트를 할당합니다.
- NDVRC1: CA Endevor® 백엔드, TSO 및 ISPF 명령을 실행하는 내장형 메커니즘이 있습니다.
- PGM(CRASERV): CARMA 서버를 시작하는 명령으로, ISPF 명령 형식입니다.
- PARM(&CRAPRM1. &CRAPRM2.): CRASERV의 매개변수이며, ISPF 명령 형식입니다. &CRAPRM1은 사용될 포트이고 &CRAPRM2가 제한시간 값입니다.

CA Endevor® SCM RAM은 ISPF 환경에서 활성화되며, 이는 PDS RAM에 필요한 TSO 환경도 사용 가능함을 의미합니다.

(선택사항) 사용자 정의 할당 exec

모든 CARMA 서버 시작 메소드에는 데이터 세트 할당에 관한 제한이 있습니다. 예를 들어, TSO 접두부 대체는 JCL 또는 CRASTART에서 사용할 수 없습니다.

그러나 TSO 또는 ISPF가 시작된 후 그리고 CARMA가 시작되기 전에 호출되는 exec를 작성하면, TSO 또는 ISPF에서 사용 가능한 전체 변수 및 서비스 범위를 사용하여 필요한 할당을 수행할 수 있습니다.

Developer for System z는 이 장 앞부분에서 설명한 사전 구성된 각 설정에서 할당 exec를 사용합니다. FEK.SFEKPROC(CRANDVRA)는 CA Endevor® SCM RAM에 대한 할당 exec이고 FEK.SFEKPROC(CRAALLOC)은 사용자 정의 RAM에 대한 할당 exec입니다.

니다. exec는 사용자의 TSO 접두부를 상위 레벨 규정자로 갖는 카탈로그화된 임시 데이터 세트를 할당합니다. 할당 exec는 선택적 사용자 엑시트 호출도 담당합니다.

사용자 정의 지시사항은 exec 내에 설명되어 있습니다. 할당 exec 변경은 지원되지만 PTF 서비스가 exec를 업데이트할 때 사용자 정의를 다시 실행해야 하므로 권장되지 않습니다. 가능한 경우, 『(선택사항) CARMA 사용자 엑시트』에서 설명된 CARMA 사용자 엑시트를 대신 사용하십시오.

참고:

- 할당 exec를 업데이트할 때, CRASTART 또는 사용자의 시작 JCL에 의해 CARMA 시작 프로세스에서 이전에 수행된 할당이 영구 삭제되지 않도록 하십시오.
- 할당 exec에서 생성된 결과물은 CARMA 서버의 DD SYSTSPRT에 표시됩니다.

할당 exec를 업데이트할 때, CRASTART 또는 사용자의 시작 JCL에 의해 CARMA 시작 프로세스에서 이전에 수행된 할당이 영구 삭제되지 않도록 하십시오.

다음 샘플은 TSO만 필요한 할당 exec 시작 방법을 보여줍니다.

```
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. )  
//*
```

(선택사항) CARMA 사용자 엑시트

CARMA는 CARMA 서버의 시작 중 특수 초기화 및 시스템 종료 중 특수 정리에 대비하여 사용자 종료의 호출을 지원합니다. 사용자 종료를 사용하면 PTF 서비스로 유지 보수되는 할당 exec를 변경할 필요성이 줄어듭니다.

사용자 종료를 할당 exec에 의해 호출되고 두 번 실행됩니다. 초기화 호출은 임시 데이터 세트 할당 후와 CARMA 서버 호출 전입니다. 정리 호출은 CARMA 서버 종료 후와 임시 파일 제거 전입니다. 첫 번째 호출이 리턴 코드 99 이상으로 종료되면 CARMA 시작이 인터럽트됩니다. 이는 CARMA 서버 및 이 사용자 종료를 두 번째 호출이 모두 실행되지 않음을 의미합니다.

Developer for System z의 SMP/E 설치 중에 다른 상위 레벨 규정자를 사용하지 않으면 샘플 사용자 종료를 FEK.SFEKSAMP(CRAEXIT)로 제공됩니다. 이 샘플 사용자 엑시트는 사용자 엑시트로 전달된 시작 인수를 자세히 설명합니다.

시작 인수	설명
(STARTUP) (ENDING)	엑시트 호출이 CARMA 서버 호출 전인지 후인지를 나타내는 표시기입니다.
EXIT_RC=rc	이전 엑시트 호출의 리턴 코드입니다. (STARTUP) 호출 중에 rc는 항상 0입니다.
CARMA_RC=rc	CARMA 서버 호출의 리턴 코드입니다. (STARTUP) 호출 중에 rc는 항상 0입니다.
...	CARMA 서버 시작 명령 및 시작 인수입니다. 예를 들면, ISPSTART PGM(CRASERV) PARM(1312 420 EXIT=CRAEXIT CLIENT=9.0.1 . . .)

사용자 엑시트에서 생성된 결과물은 CARMA 서버의 DD SYSTSPRT에 표시됩니다.

(선택사항) IRXJCL 대 CRAXJCL

CARMA 서버가 TSO(IKJEFTxx)를 사용하여 시작되면, 차례로 IRXJCL REXX 일괄처리 인터페이스를 호출하는 RAM 호출 서비스에서 문제점이 발생할 수 있습니다. 이전에 RAM이 호출한 프로세서가 TSO 없이 실행된 경우 또는 온라인 TSO에서는 DD SYSTSIN 또는 SYSTSPRT를 동적으로 할당한 경우 문제점이 발생할 수 있습니다. 샘플 프로그램, CRAXJCL이 이 문제점을 해결하기 위해 제공됩니다.

이미 CARMA에 필요한 일괄처리 TSO에 해당하는 DD 이름이 할당되어 열려 있으므로 IRXJCL에 필요한 SYSTSIN 또는 SYSTSPRT를 할당하려 시도하면 프로세서가 실패할 수 있습니다. CRAXJCL 교체 모듈은 DUMMY에 SYSTSIN 및 SYSTSPRT를 할당하려 하지만, 할당이 실패할 경우 발생하는 오류는 무시합니다. 그런 다음 IRXJCL을 호출하여 실제 작업을 수행합니다.

이는 사용자 프로세서가 TSO에서 시작된 CARMA 환경에서 실행될 경우 SYSTSIN 및 SYSTSPRT로의 할당이 CARMA에서 사용하는 할당과 동일함을 의미합니다. 프로세서가 TSO/CARMA 외부에서 실행되는 경우, SYSTSIN 및 SYSTSPRT 할당은 CRAXJCL에서 작성됩니다. 그러므로 프로세서는 SYSTSIN에 할당된 데이터 세트 컨테이너에 의존해서는 안 됩니다.

*TSO/E REXX Reference(SA22-7790)*에서 설명한 대로 IRXJCL 호출에서는 PARM 필드를 사용하여 REXX 이름 및 시작 매개변수를 전달하는 것으로 가정합니다. 이는 CARMA가 SYSTSIN을 안전하게 사용할 수 있음을 의미합니다. IRXJCL이 SYSTSPRT로 전송한 출력은 CARMA의 로그에 기록됩니다.

CRAXJCL 대체 모듈을 호출하는 프로세서는 CRAXJCL을 호출하기 전에 DD SYSTSIN 또는 SYSTSPRT 할당을 시도해서는 안 됩니다.

CRAXJCL 작성

CRAXJCL 대체 모듈은 사용자가 SYSTSPRT에 대해 사용할 특정 할당을 지정하도록 사용자 정의해야 하므로 소스 형식으로 제공됩니다. SYSTSIN을 위한 할당은 보통 더미 데이터 세트에 대한 것이어야 합니다.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의 및 제출할 때 다른 위치를 지정한 경우가 아니면 샘플 어셈블러 소스 코드 및 샘플 컴파일/바인드 작업은 FEK.#CUST.ASM (CRAXJCL) 및 FEK.#CUST.JCL(CRA#CIRX)로 사용할 수 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오.

멤버 내의 문서를 사용하여, 필요에 따라 CRAXJCL 어셈블러 소스 코드를 사용자 정의하십시오. 이후에는 CRA#CIRX JCL을 사용자 정의 및 제출하여 CRAXJCL 로드 모듈을 작성하십시오. 사용자 정의 지시사항은 멤버 내의 문서를 참조하십시오.

필요하면, IRXJCL의 이름을 다른 것으로 변경할 수 있습니다. CRAXJCL 소스를 조정하여 IRXJCL에 대한 이 새 이름을 호출한 후 컴파일하고, CRAXJCL 로드 모듈의 이름을 IRXJCL로 변경하십시오. 이 설정은 모든 호출을 IRXJCL로 변경하는 것보다 더 쉽습니다.

제 4 장 (선택사항) SCLM 개발자 툴킷

SCLM 개발자 툴킷에서는 SCM의 기능을 클라이언트로 확장하는 데 필요한 도구를 제공합니다. SCLM 그 자체는 ISPF 일부로 포함되는 호스트 시스템 기반 소스 코드 관리자입니다.

SCLM 개발자 툴킷에는 SCLM과 인터페이스로 접속하며 상속 코드 개발을 위한 모든 SCLM 프로세스에 액세스할 수 있고 메인프레임의 Java EE 코드 빌드, 어셈블 및 배치를 포함하여 메인프레임의 SCLM과 동기화된 워크스테이션에서 전체 Java 및 Java EE 개발을 위한 지원을 제공하는 Eclipse 기반 플러그인이 있습니다.

요구사항 및 체크리스트

이 사용자 정의 태스크를 완료하려면 SCLM 관리자 및 보안 관리자(선택적)의 도움이 필요하며, 다음 자원 및 특수 사용자 정의 태스크가 필요합니다.

- APF 및 LINKLIST 업데이트
- Java EE 지원을 위한 SCLM 언어 변환기 정의
- Java EE 지원을 위한 SCLM 유형 정의
- (선택사항) SCLM VSAM으로의 사용자 업데이트를 허용하는 보안 규칙
- (선택사항) Ant 설치

사용자 사이트에서 SCLM 개발자 툴킷 사용을 시작하려면 다음 태스크를 수행해야 합니다. 특별히 언급되지 않는 한 모든 태스크가 필수입니다.

1. 필수 소프트웨어와 PARMLIB 업데이트를 검증하고 조정하십시오. 자세한 정보는 112 페이지의 『전제조건』의 내용을 참조하십시오.
2. Developer for System z 구성 파일을 사용자 정의하십시오. 세부사항은 다음을 참조하십시오.
 - 113 페이지의 『SCLMDT의 ISPF.conf 업데이트』
 - 113 페이지의 『SCLMDT에 대한 rsed.envvars 업데이트』
3. 선택적으로 긴/짧은 이름 변환 지원을 정의하십시오. 자세한 정보는 114 페이지의 『(선택사항) 긴/짧은 이름 변환』의 내용을 참조하십시오.
4. 선택적으로 Java EE 빌드 지원을 사용하도록 Ant를 설치하고 사용자 정의하십시오. 자세한 정보는 116 페이지의 『(선택사항) Ant 설치 및 사용자 정의』의 내용을 참조하십시오.
5. SCLM을 업데이트하여 SCLMDT 특정 파트를 정의하십시오. 자세한 정보는 118 페이지의 『SCLMDT에 대한 SCLM 업데이트』의 내용을 참조하십시오.

6. 정기적으로 SCLMDT 작업 영역을 정리하도록 선택적으로 자동화를 설정하십시오. 자세한 정보는 118 페이지의 『WORKAREA 및 /tmp에서 이전 파일 제거』의 내용을 참조하십시오.

전제조건

필요한 SCLM 유지보수 목록은 *IBM Rational Developer for System z 전제조건 (SA30-3650)*을 참조하십시오.

이 문서는 SCLM 개발자 툴킷에서 Java EE 빌드에 필요한 Ant 스펙에 대해서도 설명합니다.

경고: SCLM 개발자 툴킷은 ISPF의 TSO/ISPF Client Gateway를 사용하며, 이는 z/OS 1.8 이상이 필요함을 의미합니다.

19 페이지의 『PARMLIB 변경사항』에서 설명한 대로, SCLM 개발자 툴킷에는 시스템 설정의 추가 사용자 정의가 필요합니다. 이 변경사항에는 다음 항목이 포함됩니다.

- (BPXPRMxx) z/OS UNIX 사용자 ID당 최대 프로세스 수를 늘립니다.
- (PROGxx) APF에서 SYS1.LINKLIB 및 REXX 런타임, REXX.V1R4M0.SEAGLPA 또는 REXX.V1R4M0.SEAGALT에 대해 권한을 부여합니다.
- (PROGxx/LPALSTxx) ISP.SISPLPA, ISP.SISpload, SYS1.LINKLIB 및 REXX 런타임을 LINKLIST/LPALIB에 배치합니다.

또한 SCLM 개발자 툴킷에서는 SDSF 또는 TSO **OUTPUT** 명령을 사용하여 작업 완료 상태 및 작업 출력을 검색합니다. 양쪽 메소드 모두 추가 주의사항이 있습니다.

- SDSF는 개별적으로 순서 지정, 설치 및 구성되어야 합니다. SDSF에서도 JES2가 필요합니다.
- TSO **OUTPUT** 명령의 기본 설정을 사용하면 사용자가 해당하는 특정 사용자 ID로 시작하는 작업 출력만 검색할 수 있습니다. **OUTPUT** 기능 전체를 사용하려면, 사용자가 해당 사용자가 소유하는 작업 출력을 검색할 수 있지만 해당 사용자의 사용자 ID를 시작할 수 없도록 샘플 TSO/E 종료 IKJEFF53을 수정해야 합니다. 이 종료에 대한 자세한 정보는 *TSO/E Customization(SA22-7783)*을 참조하십시오.

사용자는 z/OS UNIX 디렉토리 /tmp/ 및 /var/rdz/WORKAREA/에 대한 읽기, 쓰기 및 실행 권한이 필요합니다. FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 WORKAREA 디렉토리는 /var/rdz/에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오.

SCLMDT의 ISPF.conf 업데이트

SCLM 개발자 툴킷에서는 표준 ISPF/SCLM 스켈레톤을 사용하므로, ISP.SISPSLIB 스켈레톤 라이브러리가 ISPF.conf의 ISPSLIB 접합에 할당되어 있는지 확인하십시오. ISP.SISPSENU 데이터 세트 사용은 선택사항입니다.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 ISPF.conf는 /etc/rdz/에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오. TSO **OEDIT** 명령으로 파일을 편집할 수 있습니다.

참고: 변경은 업데이트 뒤에 호스트 시스템에 연결되는 모든 클라이언트에 적용됩니다.

다음 샘플 코드는 ISPF.conf 파일을 표시하며, 사용자의 시스템 환경에 맞도록 사용자 정의되어야 합니다. 주석 행은 별표(*)로 시작합니다. 동일한 라인의 접합에 데이터 세트를 추가하고 쉼표(,)로 이름을 구분하십시오. ISPF.conf 사용자 정의에 대한 자세한 정보는 64 페이지의 『ISPF.conf, ISPF의 TSO/ISPF Client Gateway 구성 파일』의 내용을 참조하십시오.

```
* REQUIRED:
sysproc=ISP.SISPCLIB,FEK.SFEKPROC
ispmllib=ISP.SISPMENU
isptlib=ISP.SISPSTENU
ispplib=ISP.SISPSENU
ispslib=ISP.SISPSLIB
ispllib=ISP.SISPLLOAD

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

그림 30. SCLMDT의 ISPF.conf 업데이트

참고:

- 자체 DD 유사 명령문 및 데이터 세트 접합을 추가하여 TSO 환경을 사용자 정의함으로써 TSO 로그인 프로시저를 흉내낼 수 있습니다. 자세한 정보는 호스트 구성 참조서 (SA30-4501)에 있는 "TSO 환경 사용자 정의"의 내용을 참조하십시오.
- 일괄처리 빌드를 수행 중인 경우 사용자 정의된 FLMLIBS 스켈레톤 버전이 ISPF/SCLM 스켈레톤 라이브러리보다 먼저 연결되도록 하십시오.

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

SCLMDT에 대한 rsed.envvars 업데이트

SCLM 개발자 툴킷은 rsed.envvars에서 몇 가지 지시문 세트를 사용하여 데이터 세트 및 디렉토리를 찾습니다.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 `rsed.envvars`는 `/etc/rdz/`에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오. TSO **OEDIT** 명령으로 파일을 편집할 수 있습니다.

참고: 변경된 내용을 적용하려면, RSED 시작 태스크를 다시 시작하십시오.

다음 코드 샘플에서는 `rsed.envvars`에서 `SCLMDT` 지시문을 표시하며, 이는 사용자의 시스템 환경에 맞도록 사용자 정의해야 합니다. `rsed.envvars` 사용자 정의에 대한 자세한 정보는 41 페이지의 『`rsed.envvars`, RSE 구성 파일』의 내용을 참조하십시오.

```
_SCLMDT_CONF_HOME=/var/rdz/sc1mdt
#STEPLIB=$STEPLIB:FEK.SFEKAUTH:FEK.SFEKLOAD
#_SCLMDT_TRANTABLE=FEK.#CUST.LSTRANS.FILE
#ANT_HOME=/usr/lpp/apache/Ant/apache-ant-1.7.1
_SCLMDT_BASE_HOME=$RSE_HOME
_SCLMDT_WORK_HOME=$CGI_ISPHOME
CGI_DTWORk=$_SCLMDT_WORK_HOME
```

그림 31. `SCLMDT`에 대한 `rsed.envvars` 업데이트

(선택사항) 긴/짧은 이름 변환

SCLM 개발자 툴킷은 긴 이름 파일을 SCLM에 저장할 수 있는 기능을 제공합니다. 긴 파일 이름은 이름이 8자 이상이거나 대소문자가 혼합된 파일입니다. 긴 파일 이름은 SCLM에서 사용되는 8자 멤버명으로 긴 파일 이름이 맵핑되는 VSAM 파일을 사용하면 저장할 수 있습니다.

참고:

- z/OS 1.8 이전 버전의 경우, 이 기능은 APAR OA11426을 나타내는 기본 ISPF/SCLM PTF를 통해 제공됩니다.
- 긴/짧은 이름 변환은 IBM SCLM 관리자 툴킷과 같은 기타 SCLM 관련 제품에서도 사용됩니다.

LSTRANS.FILE 작성, 긴/짧은 이름 변환 VSAM

긴/짧은 이름 변환 VSAM을 작성하려면 ISP.SISPSAMP ISPF 샘플 라이브러리에서 샘플 FLM02LST 멤버를 사용자 정의하고 제출하십시오. 본 서적의 구성 단계에서는 다음 샘플 설정 JCL에 표시된 대로 VSAM의 이름이 `FEK.#CUST.LSTRANS.FILE`로 변경되어야 합니다.

```

//FLM02LST JOB <job parameters>
//*
/* 주의: 이것은 JCL 프로시저도 완료 작업도 아닙니다.
/* 이 샘플을 사용하기 전에,
/* 다음과 같이 수정하십시오.
/* 1. 시스템 요구사항에 맞도록 작업 매개변수를 변경하십시오.
/* 2. VSAM을 보유할 볼륨으로 *****를 변경하십시오.
/* 3. SCLM 변환 VSAM의 이름 지정 규칙에 맞도록 FEK.#CUST.LSTRANS.FILE의 모든 참조를
/*   변경하십시오.
/*
//CREATE EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
DELETE FEK.#CUST.LSTRANS.FILE
SET MAXCC=0
DEFINE CLUSTER(NAME(FEK.#CUST.LSTRANS.FILE) -
              VOLUMES(*****)) -
              RECORDSIZE(58 2048) -
              SHAREOPTIONS(3 3) -
              CYLINDERS(1 1) -
              KEYS(8 0) -
              INDEXED) -
DATA (NAME(FEK.#CUST.LSTRANS.FILE.DATA)) -
INDEX (NAME(FEK.#CUST.LSTRANS.FILE.INDEX))

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

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

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

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

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

```

그림 32. FLM02LST: 긴/짧은 이름 변환 설정 JCL

참고: 호스트 구성 참조서 (SA30-4501)의 "보안 고려사항"에서 설명한 대로 사용자에게는 이 VSAM 데이터 세트에 업데이트 권한이 필요합니다.

긴/짧은 이름 변환을 위한 rsed.envvars 업데이트

긴/짧은 이름 변환을 사용하기 전에, 긴/짧은 이름 변환 VSAM의 이름과 일치하도록 rsed.envvars 환경 변수 _SCLMDT_TRANTABLE을 주석 해제한 후 설정하십시오.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 rsed.envvars는 /etc/rdz/에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오. TSO **OEDIT** 명령으로 파일을 편집할 수 있습니다.

참고: 변경된 내용을 적용하려면, RSED 시작 태스크를 다시 시작하십시오.

(선택사항) Ant 설치 및 사용자 정의

이 단계는 SCLM에서 Java EE 빌드 지원을 사용하려는 경우에만 필요합니다.

Apache Ant는 오픈 소스 Java 빌드 도구이며 <http://ant.apache.org/>에서 다운로드할 수 있습니다. Ant는 ASCII 형식으로 배포되는 스크립트 및 텍스트 파일로 구성되며, z/OS UNIX에서 실행하려면 ASCII/EBCDIC 변환을 수행해야 합니다.

다음 단계를 수행하여 z/OS에서 Ant를 구현하고, Developer for System z에 정의하십시오.

- 바이너리 형식으로 최신 Ant 압축 파일을 z/OS UNIX 파일 시스템에 다운로드하십시오. ANT의 .zip 버전을 다운로드하십시오.
- 예를 들어, TSO OMVS 명령을 사용하여 z/OS UNIX 명령행 세션을 열고 설치를 계속하십시오.
- **mkdir -p /home-dir** 명령을 사용하여 Ant 설치의 홈 디렉토리를 작성하고 **cd /home-dir** 명령을 사용하여 이를 현재 디렉토리로 작성하십시오.
- JAR 추출 명령, **jar -xf apache-ant-1.7.1.zip**을 사용하여 현재 디렉토리로 파일을 추출하십시오. **jar** 명령을 사용하려면 Java 바이너리 디렉토리가 로컬 z/OS UNIX PATH에 있어야 합니다. 반면 Java 바이너리 위치에 대해서는 완전한 명령을 입력하십시오(예: /usr/lpp/java/J6.0/bin/jar -xf apache-ant-1.7.1.zip).
- 선택적으로 /usr/lpp/rdz/samples/BWBTRANT 샘플 스크립트를 사용자 정의 및 실행하여 Ant 텍스트 파일 모두를 EBCDIC으로 변환하십시오.

참고: 이 스크립트는 한 번만 실행하십시오. 여러 번 실행하면 Ant 설치가 손상됩니다.

- 성공적으로 변환되었는지 확인하려면, Ant 디렉토리에서 텍스트 파일(예: apache-ant-1.7.1/README)을 찾아 여십시오. 파일을 읽을 수 있으면 변환이 완료된 것입니다.
- 모든 사용자가 ANT 디렉토리에서 파일을 읽고 실행할 수 있게 하려면 **chmod -R 755 *** 명령을 사용하십시오.
- Ant를 사용하기 전에, rsed.envvars 환경 변수 JAVA_HOME과 ANT_HOME을 설정하십시오.
 - JAVA_HOME은 Java 홈 디렉토리를 나타내는 데 필요합니다. 예를 들어, 다음과 같습니다.
 JAVA_HOME=/usr/lpp/java/IBM/J6.0
 - ANT_HOME은 Ant 홈 디렉토리를 나타내는 데 필요합니다. 예를 들어, 다음과 같습니다.
 ANT_HOME=/usr/lpp/Apache/Ant/apache-ant-1.7.1

예를 들어, 다음과 같습니다.

- TSO OMVS
- mkdir -p /usr/lpp/Apache/Ant
- cd /usr/lpp/Apache/Ant
- jar -xf /u/userid/apache-ant-1.7.1
- /usr/lpp/rdz/samples/BWBTRANT
- cat ./apache-ant-1.7.1/README
- chmod -R 755 *
- oedit /etc/rsed.envvars

Ant 초기화가 완료되었는지 테스트하려면 다음을 수행하십시오.

- Ant 및 Java 바이너리 디렉토리를 환경 변수 PATH에 추가하십시오.

예를 들어, 다음과 같습니다.

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

- 설치 후 버전을 표시하려면 ant -version을 실행하십시오.

예를 들어, 다음과 같습니다.

```
ant -version
```

참고: 이러한 방식으로 PATH 명령문을 설정하는 것은 단지 테스트용이며 운영을 위한 것은 아닙니다.

SCLMDT에 대한 SCLM 업데이트

SCLM 자체에서도 SCLM 개발자 툴킷에 대해 작업하기 위한 사용자 정의가 필요합니다. 필요한 사용자 정의 태스크에 대한 자세한 정보를 알려면, *IBM Rational Developer for System z SCLM Developer Toolkit Administrator's Guide(SC23-9801)*를 참조하십시오.

- Java EE 지원을 위한 언어 변환기 정의
- Java EE 지원을 위한 SCLM 유형 정의

사용자 정의 및 프로젝트 정의 태스크를 완료하려면, 표 13에 설명한 대로 SCLM 관리자가 여러 가지 Developer for System z 사용자 정의 가능한 값을 알고 있어야 합니다.

표 13. SCLM 관리자 체크리스트

설명	<ul style="list-style-type: none"> • 기본값 • 답을 찾을 수 있는 위치 	값
Developer for System z 샘플 라이브러리	<ul style="list-style-type: none"> • FEK.SFEKSAMV • SMP/E 설치 	
Developer for System z 샘플 디렉토리	<ul style="list-style-type: none"> • /usr/lpp/rdz/samples • SMP/E 설치 	
Java 바이너리 디렉토리	<ul style="list-style-type: none"> • /usr/lpp/java/J6.0/bin • rsed.envvars - \$JAVA_HOME/bin 	
Ant 바이너리 디렉토리	<ul style="list-style-type: none"> • /usr/lpp/Apache/Ant/apache-ant-1.7.1/bin • rsed.envvars - \$ANT_HOME/bin 	
WORKAREA 홈 디렉토리	<ul style="list-style-type: none"> • /var/rdz • rsed.envvars - \$CGI_ISPWORK 	
SCLMDT 프로젝트 구성 홈 디렉토리	<ul style="list-style-type: none"> • /var/rdz/sclmdt • rsed.envvars - \$_SCLMDT_CONF_HOME 	
긴/짧은 이름 변환 VSAM	<ul style="list-style-type: none"> • FEK.#CUST.LSTRANS.FILE • rsed.envvars - \$_SCLMDT_TRANTABLE 	

WORKAREA 및 /tmp에서 이전 파일 제거

SCLM 개발자 툴킷 및 ISPF의 TSO/ISPF Client Gateway는 동일한 WORKAREA와 /tmp 디렉토리를 공유하며, 둘 다 정기적으로 정리해야 합니다. 이 태스크에 대한 자세한 정보는 152 페이지의 『(선택사항) WORKAREA 및 /tmp 정리』의 내용을 참조하십시오.

제 5 장 (선택사항) 애플리케이션 배치 관리자(더 이상 사용되지 않음)

참고: 애플리케이션 배치 관리자는 더 이상 사용되지 않는 것으로 표시되었습니다. 계속 지원되더라도 이 기능은 더 이상 개선되지 않습니다.

Developer for System z는 애플리케이션 배치 관리자의 특정 기능을 다양한 컴포넌트의 공통 배치 방법으로 사용합니다. 다음 기능이 사용되는 경우 이 장에 수록된 사용자 정의 단계가 필요합니다.

- 엔터프라이즈 서비스 도구
- BMS 화면 디자이너
- MFS 화면 디자이너
- CICSTS 코드 생성

참고: 엔터프라이즈 서비스 도구에는 서비스 플로우 모델러(SFM) 및 엔터프라이즈용 XML 서비스와 같은 여러 도구가 포함됩니다.

애플리케이션 배치 관리자를 사용자 정의하면 CICS 자원 정의(CRD) 서버가 추가되며, 이 서버는 다음 기능을 지원하기 위해 z/OS에서 CICS 애플리케이션으로서 실행됩니다.

- CICS 자원 조회
- CICS 자원 정의는 CICSplex SM과 비CICSplex SM 환경 모두에서 요청을 설치 및 설치 제거합니다.
- 프로그램 및 맵 세트 단계별 요청 도입
- 파이프라인 스캔 요청
- Manifest 내보내기, 가져오기 및 업데이트 요청

CICS 관리자는 *호스트 구성 참조서 (SA30-4501)*에서 "CICSTS 고려사항"의 CRD 서버에 대한 자세한 정보를 발견할 수 있습니다.

요구사항 및 체크리스트

이 사용자 정의 태스크를 완료하려면 CICS 관리자, TCP/IP 관리자 및 보안 관리자의 도움이 필요하며, 다음과 같은 자원 또는 특수 사용자 정의 태스크가 필요합니다.

- 외부 통신을 위한 TCP/IP 포트 정의
- CICS 리전 JCL 업데이트
- CICS 리전 CSD 업데이트
- CICS 리전에 그룹 정의

- 관리자가 애플리케이션 배치 관리자 VSAM을 업데이트할 수 있도록 하는 보안 규칙 작성
- CICSTS 보안 설정
- (선택사항) CICS 트랜잭션 이름 정의
- (선택사항) 사용자가 애플리케이션 배치 관리자 VSAM을 업데이트할 수 있도록 하는 보안 규칙 작성

사용자 사이트에서 애플리케이션 배치 관리자 사용을 시작하려면 다음 태스크를 수행하십시오. 특별히 언급되지 않는 한 모든 태스크가 필수입니다.

1. CRD 저장소를 작성하십시오. 자세한 정보는 『CRD 저장소』의 내용을 참조하십시오.
2. 사용할 CICS 인터페이스(RESTful 또는 웹 서비스)를 선택하십시오. 인터페이스는 공존할 수 있습니다. 자세한 정보는 121 페이지의 『RESTful 대 웹 서비스』의 내용을 참조하십시오.
3. 필요할 경우 REST 서비스 특정 사용자 정의를 수행하십시오. 자세한 정보는 122 페이지의 『RESTful 인터페이스를 사용하는 CRD 서버』의 내용을 참조하십시오.
 - CICS 기본 연결 리전에 CRD 서버를 정의하십시오.
 - 선택적으로, CICS 비기본 연결 리전에 CRD 서버를 정의하십시오.
 - 선택적으로, CRD 서버 트랜잭션 ID를 사용자 정의하십시오.
4. 필요할 경우 웹 서비스 특정 사용자 정의를 수행하십시오. 자세한 정보는 123 페이지의 『웹 서비스 인터페이스를 사용하는 CRD 서버』의 내용을 참조하십시오.
 - 파이프라인 메시지 핸들러를 CICS RPL 집합에 추가하십시오.
 - CICS 기본 연결 리전에 CRD 서버를 정의하십시오.
 - 선택적으로, CICS 비기본 연결 리전에 CRD 서버를 정의하십시오.
5. 선택적으로, Manifest 저장소를 작성하십시오. 자세한 정보는 126 페이지의 『(선택 사항) Manifest 저장소』의 내용을 참조하십시오.

CRD 저장소

ADNVCRD 작업을 사용자 정의 및 제출하여 CRD 저장소 VSAM 데이터 세트를 할당 및 초기화하십시오. 사용자 정의 지시사항은 멤버 내의 문서를 참조하십시오.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 ADNVCRD는 FEK.#CUST.JCL에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오.

CICS 기본 연결 리전 각각에 대해 별도의 저장소를 작성하십시오. 저장소 공유는 관련된 모든 CICS 리전이 저장소에 저장된 동일한 값을 사용할 것임을 의미합니다.

참고:

- 기존 CRD 서버 저장소는 Developer for System z 버전 7.6.1의 관리 유틸리티에 URIMAP 지원이 추가될 수 있도록 강화되어야 합니다. 자세한 정보는 *호스트 구성 참조서 (SA30-4501)*에 있는 "관리 유틸리티 마이그레이션 참고사항"의 내용을 참조하십시오.
- 특별히 공지하지 않으면 사용자 정의된 값이 들어 있는 현재 CRD 서버 저장소가 Developer for System z 릴리스에서 다시 사용될 수 있습니다.

사용자는 CRD 저장소에 대한 읽기 액세스 권한이 필요하고, CICS 관리자에게는 업데이트 액세스 권한이 필요합니다.

CICS 관리 유틸리티

Developer for System z는 CICS 관리자가 CICS 자원 정의의 기본값을 제공할 수 있도록 하는 관리 유틸리티를 제공합니다. 이 기본값은 읽기 전용이거나 애플리케이션 개발자가 편집할 수 있습니다.

관리 유틸리티를 ADNJSAPU 샘플 작업이라고 합니다. 이 유틸리티를 사용하려면 CRD 저장소에 대한 업데이트 액세스가 필요합니다.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 ADNJSAPU는 FEK.#CUST.JCL에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오.

자세한 정보는 *호스트 구성 참조서 (SA30-4501)*의 "CICSTS 고려사항"에 있습니다.

RESTful 대 웹 서비스

CICS 트랜잭션 서버 버전 4.1 이상에는 RESTful(Representational State Transfer) 원칙을 사용하여 디자인된 HTTP 인터페이스가 있습니다. 이 RESTful 인터페이스는 클라이언트 애플리케이션이 사용하는 전략적 CICSTS 인터페이스입니다. 이전 웹 서비스 인터페이스는 안정화되었으며 기능 보강은 RESTful 인터페이스에만 해당됩니다.

애플리케이션 배치 관리자는 이 지시 명령문을 준수하며 Developer for System 버전 7.6 이상의 새 서비스 모두에 RESTful CRD 서버가 필요합니다.

RESTful 및 웹 서비스 인터페이스는 필요한 경우 동시에 단일 CICS 리전에서 활성화될 수 있습니다. 이런 경우 두 개의 CRD 서버가 리전에서 활성화됩니다. 두 서버는 동일한 CRD 저장소를 공유합니다. CICS는 리전에 두 번째 인터페이스가 정의될 때 중복 정의에 대한 경고를 발행합니다.

RESTful 인터페이스를 사용하는 CRD 서버

이 절의 정보에서는 RESTful 인터페이스를 사용하여 Developer for System z 클라이언트와 통신하는 CRD 서버 정의 방법에 대해 설명합니다.

RESTful 및 웹 서비스 인터페이스는 필요한 경우 동시에 단일 CICS 리전에서 활성화될 수 있습니다. 이런 경우 두 개의 CRD 서버가 리전에서 활성화됩니다. 두 서버는 동일한 CRD 저장소를 공유합니다. CICS는 리전에 두 번째 인터페이스가 정의될 때 중복 정의에 대한 경고를 발행합니다.

CICS 기본 연결 리전

CRD 서버는 기본 연결 리전에 정의해야 합니다. 이것은 Developer for System z로부터의 웹 서비스 요청을 처리하는 WOR(Web Owning Region)입니다.

- FEK.SFEKLOAD(ADNCRD*, ADNANAL and ADNREST) 로드 모듈을 CICS 기본 연결 리전의 CICS RPL 집합(DD 문 DFHRPL)에 배치하십시오. 적용된 유지보수가 CICS에서 자동으로 사용될 수 있도록 집합에 설치 데이터 세트를 추가하여 배치해야 합니다.
- CICS 기본 연결 리전의 CICS 시스템 정의(CSD)를 업데이트하도록 ADNCSDRS 작업을 사용자 정의 및 제출하십시오. 사용자 정의 지시사항은 멤버 내의 문서를 참조하십시오.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 ADNCSDRS는 FEK.#CUST.JCL에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오.

- 이 리전에 대한 애플리케이션 배치 관리자 그룹을 설치하려면 해당하는 CEDA 명령을 사용하십시오. 예를 들어, 다음과 같습니다.

```
CEDA INSTALL GROUP(ADNPCRGP)
```

CICS 비기본 연결 리전

CRD 서버는 또한 하나 이상의 추가적 비기본 연결 리전, 주로 AOR(Application Owning Regions)과 함께 사용될 수 있습니다.

참고: CICSplex® SM BAS(Business Application Services)를 사용하여 CICS 자원 정의를 관리하는 경우 이 단계를 수행할 필요가 없습니다.

- FEK.SFEKLOAD(ADNCRD*) 애플리케이션 배치 관리자 로드 모듈을 이러한 비기본 연결 리전의 CICS RPL 집합(DD 문 DFHRPL)에 배치하십시오. 적용된 유지보수가 CICS에서 자동으로 사용될 수 있도록 집합에 설치 데이터 세트를 추가하여 배치하십시오.
- 이러한 비기본, 연결 리전의 CSD를 업데이트하도록 ADNCSRAR 작업을 사용자 정의 및 제출하십시오. 사용자 정의 지시사항은 멤버 내의 문서를 참조하십시오.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 ADNCSDAR은 FEK.#CUST.JCL에 위치합니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오.

- 이러한 리전에 대해 애플리케이션 배치 관리자 그룹을 설치하려면 적절한 CEDA 명령을 사용하십시오. 예를 들어, 다음과 같습니다.

CEDA INSTALL GROUP(ADNARRGP)

(선택사항) CRD 서버 트랜잭션 ID 사용자 정의

Developer for System z는 CICS 자원을 정의 및 조회할 때 CRD 서버가 사용하는 여러 트랜잭션을 지원합니다.

표 14. 기본 CRD 서버 트랜잭션 ID

트랜잭션	설명
ADMS	Manifest 처리 도구로부터 CICS 자원 변경 요청의 경우. 일반적으로 CICS 관리자를 대상으로 합니다.
ADMI	CICS 자원을 정의, 설치 또는 설치 제거하는 요청의 경우
ADMR	CICS 환경 또는 자원 정보를 검색하는 기타 모든 요청의 경우

다음 단계에 따라 사용자의 사이트 표준과 일치하도록 트랜잭션 ID를 변경할 수 있습니다.

1. ADNTXNC를 사용자 정의 및 제출하여 로드 모듈 ADNRCUST를 작성하십시오. 사용자 정의 지시사항은 멤버 내의 문서를 참조하십시오.
2. CRD 서버가 정의되어 있는 CICS 리전의 CICS RPL 집합(DD 명령문 DFHRPL)에 결과 ADNRCUST 로드 모듈을 배치하십시오.
3. CRD 서버가 정의되어 있는 CICS 리전에 대한 프로그램으로서 ADNRCUST를 정의하도록 ADNCSDTX를 사용자 정의하고 제출하십시오. 사용자 정의 지시사항은 멤버 내의 문서를 참조하십시오.

참고: RESTful CRD 서버는 항상 ADNRCUST 로드 모듈을 로드하려 합니다. 그러므로, 트랜잭션 ID를 변경하지 않은 경우라도 ADNRCUST 로드 모듈을 작성 및 정의하여 약간의 성능상 장점을 얻을 수 있습니다.

웹 서비스 인터페이스를 사용하는 CRD 서버

이 절의 정보에서는 웹 서비스 인터페이스를 사용하여 Developer for System z 클라이언트와 통신하는 CRD 서버 정의 방법에 대해 설명합니다.

RESTful 및 웹 서비스 인터페이스는 필요한 경우 동시에 단일 CICS 리전에서 활성화될 수 있습니다. 이런 경우 두 개의 CRD 서버가 리전에서 활성화됩니다. 두 서버는 동일한 CRD 저장소를 공유합니다. CICS는 리전에 두 번째 인터페이스가 정의될 때 중복 정의에 대한 경고를 발행합니다.

파이프라인 메시지 핸들러

파이프라인 메시지 핸들러(ADNTMSGH)는 SOAP 헤더에서 사용자 ID 및 비밀번호를 처리하여 보안에 사용됩니다. ADNTMSGH는 샘플 파이프라인 구성 파일에서 참조되므로, CICS RPL 접합에 배치되어야 합니다. 파이프라인 메시지 핸들러 및 필요한 보안 설정에 대해 알려면, 호스트 구성 참조서 (SA30-4501)에서 "CICSTS 고려사항"의 내용을 참조하십시오.

Developer for System z는 CICS 자원을 정의하고 조회할 때 CRD 서버가 사용하는 다중 트랜잭션을 제공합니다. 이 트랜잭션 ID는 요청된 오퍼레이션에 따라 ADNTMSGH에서 설정합니다. ADNTMSGH에 대해 사이트 고유의 사용자 정의를 허용하기 위해 샘플 COBOL 소스 코드가 제공됩니다.

표 15. 기본 CRD 서버 트랜잭션 ID

트랜잭션	설명
ADMS	Manifest 처리 도구로부터 CICS 자원 변경 요청의 경우. 일반적으로 CICS 관리자를 대상으로 합니다.
ADMI	CICS 자원을 정의, 설치 또는 설치 제거하는 요청의 경우
ADMR	CICS 환경 또는 자원 정보를 검색하는 기타 모든 요청의 경우

기본값 사용:

- FEK.SFEKLOAD(ADNTMSGH) 로드 모듈을 CICS 기본 연결 리전의 CICS RPL 접합(DD 명령문 DFHRPL)에 배치하십시오. 적용된 유지보수가 CICS에서 자동으로 사용될 수 있도록 접합에 설치 데이터 세트를 추가하여 배치하십시오.

ADNTMSGH 사용자 정의:

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 샘플 ADNMSGH* 멤버는 FEK.#CUST.JCL 및 FEK.#CUST.COBOL에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오.

- 사용자의 사이트 표준에 맞도록 샘플 파이프라인 메시지 핸들러(COBOL) 소스 코드, FEK.#CUST.COBOL(ADNMSGHS)를 사용자 정의하십시오.
- FEK.#CUST.JCL(ADNMSGHC) 작업을 사용자 정의 및 제출하여 사용자 정의된 ADNMSGHS 소스를 컴파일하십시오. 사용자 정의 지시사항에 대해서는 ADNMSGHC 내의 문서를 참조하십시오. 결과로 생기는 로드 모듈의 이름은 ADNTMSGH여야 합니다.
- 결과로 생기는 ADNTMSGH 로드 모듈을 CICS 기본 연결 리전의 CICS RPL 접합(DD 명령문 DFHRPL)에 배치하십시오.

참고: 사용자 정의된 ADNTMSGH 로드 모듈이 FEK.SFEKLOAD에 대한 참조 앞에 위치하는지 확인하십시오. 그렇지 않으면 기본값이 사용됩니다.

CICS 기본 연결 리전

CRD 서버는 기본 연결 리전에 정의해야 합니다. 이것은 Developer for System z의 서비스 요청을 처리하는 리전입니다.

- FEK.SFEKLOAD(ADNCRD*, ADNANAL and ADNREST) 로드 모듈을 CICS 기본 연결 리전의 CICS RPL 접합(DD 문 DFHRPL)에 배치하십시오. 적용된 유지보수가 CICS에서 자동으로 사용될 수 있도록 접합에 설치 데이터 세트를 추가하여 배치하십시오. 124 페이지의 『파이프라인 메시지 핸들러』에 설명한 대로 파이프라인 메시지 핸들러 로드 모듈, ADNTMSGH도 RPL 접합에 배치해야 합니다.
- ADNCSDWS 작업을 사용자 정의 및 제출하여 CICS 기본 연결 리전의 CICS 시스템 정의(CSD)를 업데이트하십시오. 사용자 정의 지시사항은 멤버 내의 문서를 참조하십시오. 이 작업에서 사용되는 트랜잭션 ID는 파이프라인 메시지 핸들러가 사용하는 ID와 일치해야 하며, 사용자 정의될 수도 있습니다.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 ADNCSDWS는 FEK.#CUST.JCL에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오.

- 이 리전에 대한 애플리케이션 배치 관리자 그룹을 설치하려면 해당하는 CEDA 명령을 사용하십시오. 예를 들어, 다음과 같습니다.

CEDA INSTALL GROUP(ADNPCRGP)

CICS 비기본 연결 리전

CRD 서버는 또한 하나 이상의 추가적 비기본 연결 리전, 주로 AOR(Application Owning Regions)과 함께 사용될 수 있습니다.

참고: CICSplex SM BAS(Business Application Services)를 사용하여 CICS 자원 정의를 관리하는 경우 이 단계를 수행할 필요가 없습니다.

- FEK.SFEKLOAD(ADNCRD*) 애플리케이션 배치 관리자 로드 모듈을 이 비기본 연결 리전의 CICS RPL 접합(DD 명령문 DFHRPL)에 배치하십시오. 적용된 유지보수가 CICS에서 자동으로 사용될 수 있도록 접합에 설치 데이터 세트를 추가하여 배치해야 합니다.
- 이러한 비기본, 연결 리전의 CSD를 업데이트하도록 ADNCSDAR 작업을 사용자 정의 및 제출하십시오. 사용자 정의 지시사항은 멤버 내의 문서를 참조하십시오.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 ADNCSDAR은 FEK.#CUST.JCL에 위치합니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오.

- 이러한 리전에 대해 애플리케이션 배치 관리자 그룹을 설치하려면 적절한 CEDA 명령을 사용하십시오. 예를 들어, 다음과 같습니다.

CEDA INSTALL GROUP(ADNARRGP)

(선택사항) Manifest 저장소

Developer for System z에서는 클라이언트가 선택한 CICS 자원에 대해 설명하는 Manifest를 찾아보고 선택적으로 변경할 수 있습니다. CICS 관리자가 설정한 권한에 따라, CICS 관리자가 추가 처리할 수 있도록 변경사항은 Manifest 저장소에서 직접 수행되거나 Manifest 저장소로 내보낼 수 있습니다.

참고:

- 이 단계는 Manifest 처리 도구에서 처리할 수 있도록 Developer for System z에서 Manifest를 내보내는 경우에만 필요합니다.
- Manifest 처리 도구는 IBM CICS Explorer용 플러그인입니다.

Manifest 저장소 VSAM 데이터 세트를 할당 및 초기화한 후 이를 CICS 기본 연결 리전에 정의하려면 ADNVMFST 작업을 사용자 정의한 후 제출하십시오. 사용자 정의 지시사항은 멤버 내의 문서를 참조하십시오. CICS 기본 연결 리전 각각에 대해 별도의 Manifest 저장소를 작성해야 합니다. 모든 사용자에게는 Manifest 저장소에 대한 업데이트 액세스가 필요합니다.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 ADNVMFST는 FEK.#CUST.JCL에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오.

제 6 장 (선택사항) 호스트 기반 코드 분석

Developer for System z 클라이언트와 유사하게, Developer for System z 호스트는 별도의 제품으로 제공하는 코드 분석 도구인 Rational Developer for System z 호스트 유틸리티의 실행을 지원합니다. 호스트에서 코드 분석하면 매일 일괄처리 작업에 통합할 수 있다는 이점이 있습니다.

다음 코드 분석 도구는 호스트에서 사용 가능합니다.

- 코드 검토: 서로 다른 심각도의 규칙을 사용하여 코드 검토는 소스 코드를 스캔하고 규칙 위반을 보고합니다.
- 코드 적용: 실행 프로그램을 분석하고 실행되는 행을 실행 가능한 행의 총 수와 비교하여 보고서를 생성합니다.

요구사항 및 체크리스트

사용자 사이트에서 호스트 기반 코드 분석 도구를 사용하기 위해 다른 관리자의 도움은 필요하지 않지만 다음 태스크는 필요합니다. 특별히 언급되지 않는 한 모든 태스크가 필수입니다.

1. IBM Rational Developer for System z 호스트 유틸리티 프로그램 디렉토리 (GI13-2864) 문서에 따라 Rational Developer for System z 호스트 유틸리티를 설치합니다. 제공된 기본값을 사용할 때 이 제품은 상위 레벨 규정자 AKG 및 z/OS UNIX 경로 /usr/lpp/rdzutil를 사용하여 설치됩니다.
2. AKG.SAKGSAMP(AKGSETUP)을 사용자 정의하고 제출하여 제공한 샘플의 사용자 정의할 수 있는 사본을 작성합니다. 이 작업은 다음 태스크를 수행합니다.
 - AKG.#CUST.PROCLIB를 작성하여 샘플 SYS1.PROCLIB 멤버로 채웁니다.
 - AKG.#CUST.JCL을 작성하여 샘플 구성 JCL로 채웁니다.

코드 검토

코드 검토는 여러 다른 심각도 레벨의 규칙을 사용하여 코드 검토는 소스 코드를 스캔하고 규칙 위반을 보고합니다. 이 도구는 Cobol 및 PL/I에 대한 규칙 제공자와 함께 제공하지만 다른 규칙 제공자를 추가할 수 있습니다.

Developer for System z 호스트 유틸리티에서는 샘플 프로시저 AKGCR를 제공하여 일괄처리 모드에서 코드 리뷰 서비스 호출을 단순화시킵니다. AKG.SAKGSAMP(AKGSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정하지 않으면 AKGCR는 AKG.#CUST.PROCLIB에 있습니다.

멤버 내에 설명된 대로 샘플 프로시저 AKG.#CUST.PROCLIB(AKGCR)를 사용자 정의한 후 SYS1.PROCLIB로 복사하십시오.

AKGCR 프로시저를 시스템 프로시저 라이브러리로 복사할 수 없으면, Developer for System z 사용자에게 JOB 카드 바로 다음의 JCLLIB 카드를 호출 작업에 추가하도록 요청하십시오.

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

코드 검토 처리 수정

Developer for System z 코드 검토는 썬드파티 코드가 검토 프로세스에 참여할 수 있게 해 줍니다. 예를 들어, C/C++ 코드를 분석하기 위한 규칙 제공자를 제공하거나, 또는 사이트 별 코드 작업 규정을 인식하기 위해 Cobol 규칙 제공자를 개선할 수 있습니다.

호스트 기반 코드 검토는 Developer for System z 클라이언트와 마찬가지로 Eclipse 프로세스입니다. 그러므로 클라이언트에서 코드 검토를 위한 개발 지원 팀이 완료한 개선사항은 호스트에서 재 사용될 수 있습니다.

개선사항은 Eclipse 플러그인 또는 Eclipse 기능으로 구성됩니다. 이들을 활성화 하려면 AKGCRADD 구성 작업에 기록된 대로 기존 코드에 사용 가능하게 만들어야 합니다. AKG.SAKGSAMP(AKGSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정하지 않으면 AKGCRADD는 AKG.#CUST.JCL에 있습니다.

코드 적용

코드 적용은 실행 프로그램을 분석하고 실행 행의 총 수에 비해 실행되는 라인에 대한 보고서를 생성합니다. 코드 적용은 필요한 데이터를 수집하기 위해 통합 디버거를 사용하여 임시 포트 TCP/IP 연결을 설정함을 주의합니다. 필요에 따라 통합 디버거 대신 z/OS용 IBM 디버거 도구를 사용할 수도 있습니다.

Developer for System z 호스트 유틸리티에서는 샘플 프로시저 AKGCC를 제공하여 일괄처리 모드에서 코드 적용 서비스 호출을 단순화시킵니다. AKG.SAKGSAMP(AKGSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정하지 않으면 AKGCC는 AKG.#CUST.PROCLIB에 있습니다.

멤버 내에 설명된 대로 샘플 프로시저 AKG.#CUST.PROCLIB(AKGCC)를 사용자 정의한 후 SYS1.PROCLIB로 복사하십시오.

AKGCC 프로시저를 시스템 프로시저 라이브러리로 복사할 수 없으면, Developer for System z 사용자에게 JOB 카드 바로 다음의 JCLLIB 카드를 호출 작업에 추가하도록 요청하십시오.

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

코드 적용 결과물

코드 적용의 결과물은 Developer for System z 클라이언트로 가져오기를 위한 것이므로 z/OS UNIX 파일로 작성됩니다. 또한, 코드 적용은 이전 실행의 결과를 사용할 수 있고 이를 현재의 실행과 결합하여 여러 코드 경로에 대한 단일 보고서를 작성합니다.

이러한 이유로 Developer for System z는 코드 적용 실행의 결과물을 제거하려 하지 않으므로 결과물은 시간의 경과에 따라 누적됩니다.

z/OS UNIX에서는 현재 있는 디렉토리와 이들의 기간에 따라 파일을 삭제하는 셸 스크립트 *skulker*를 제공합니다. 지정된 날짜 및 시간에 명령을 실행하는 z/OS UNIX *cron* 디먼과 함께 결합하여 주기적으로 대상 디렉토리를 정리하는 자동화된 도구를 설정할 수 있습니다. *skulker* 스크립트와 *cron* 디먼에 대한 자세한 정보는 *UNIX 시스템 서비스 명령 참조서(SA22-7802)*를 확인하십시오.

제 7 장 (선택사항) 기타 사용자 정의 태스크

이 절에서는 다양한 선택적 사용자 정의 태스크를 결합합니다. 필요한 서비스를 구성하려면 해당 절의 지시사항을 수행하십시오.

Developer for System z 구성 파일에 대한 사용자 정의:

- 『(선택사항) pushtoclient.properties, 호스트 기반 클라이언트 제어』
- 135 페이지의 『(선택사항) ssl.properties, RSE SSL 암호화』
- 138 페이지의 『(선택사항) rsecomm.properties, RSE 추적』
- 139 페이지의 『(선택사항) include.conf, C/C++ 콘텐츠 지원을 위해 강제로 포함됨』

기타 제품에 대한 Developer for System z 관련 사용자 정의:

- 140 페이지의 『(선택사항) z/OS UNIX 서브프로젝트』
- 141 페이지의 『(선택사항) 프리프로세서 지원 포함』
- 142 페이지의 『(선택사항) Enterprise COBOL 및 PL/I용 xUnit 지원』
- 144 페이지의 『(선택사항) 엔터프라이즈 서비스 도구 지원』
- 144 페이지의 『(선택사항) CICS 양방향 언어 지원』
- 145 페이지의 『(선택사항) 생성된 코드의 진단 IRZ 메시지』
- 146 페이지의 『(선택사항) 통합 디버거』
- 150 페이지의 『(선택사항) 문제점 판별 도구 지원』
- 150 페이지의 『(선택사항) DB2 및 IMS 디버그 지원』
- 151 페이지의 『(선택사항) 파일 관리자 지원』
- 152 페이지의 『(선택사항) WORKAREA 및 /tmp 정리』

(선택사항) pushtoclient.properties, 호스트 기반 클라이언트 제어

이 사용자 정의 태스크에서는 기본 설정에 대한 도움, 특수 자원 또는 특수 사용자 정의 태스크가 필요하지 않습니다.

그룹 지원을 사용할 수 있는 경우, 이 사용자 정의 태스크를 완료하려면 보안 관리자 또는 LDAP 관리자의 도움이 필요하며, 다음 자원 또는 특수 사용자 정의 태스크가 필요합니다.

- FEK.PTC.* 프로파일에 대한 사용자 액세스를 허용하는 보안 규칙
 - 또는 FEK.PTC.* LDAP 그룹의 사용자 멤버십 정의
-

Developer for System z 클라이언트 버전 8.0.1 이후에서는 연결 시 호스트 시스템에서 클라이언트 구성 파일 및 제품 업데이트 정보를 가져올 수 있으므로, 모든 클라이언트가 공통 설정을 갖고 최신 상태를 유지합니다.

z/OS 프로젝트는 클라이언트의 z/OS 프로젝트 퍼스펙티브를 통해 각각 정의되거나 호스트 시스템에서 중앙집중식으로 정의하고 각 사용자 단위로 클라이언트에 전달될 수 있습니다. 이러한 호스트 기반 프로젝트는 해당 구조, 멤버 및 특성을 클라이언트가 수정할 수 없고 호스트 시스템에 연결될 때만 액세스 가능한 점을 제외하면 클라이언트에 정의된 프로젝트와 똑같은 모습이며 동일하게 작동합니다.

pushtoclient.properties는 해당 기능을 사용하는지 여부와 관련 데이터 저장 위치를 클라이언트에 알려줍니다. 데이터는 Developer for System z 클라이언트 관리자 또는 개발 프로젝트 관리자가 유지보수합니다.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 pushtoclient.properties는 /etc/rdz/에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오. TSO **OEDIT** 명령으로 파일을 편집할 수 있습니다. 변경된 내용을 적용하려면, RSED 시작 태스크를 다시 시작하십시오.

버전 8.0.3부터는 클라이언트 관리자가 다양한 개발자 그룹의 요구에 맞는 여러 클라이언트 구성 세트와 여러 클라이언트 업데이트 시나리오를 작성할 수 있습니다. 이러한 다중 설정과 시나리오는 LDAP 그룹의 멤버십과 같은 기준을 기반으로, 사용자에게 사용자 정의된 설정을 제공하거나 보안 프로파일을 허용하는 데 사용될 수 있습니다. 다수의 그룹 지원에 대한 자세한 정보는 *호스트 구성 참조서(SA30-4501)*에서 "클라이언트로 푸시 고려사항"을 참조하십시오.

다음 코드 샘플은 pushtoclient.properties 파일을 표시하며, 사용자의 시스템 환경에 맞도록 사용자 정의되어야 합니다. US 코드 페이지를 사용할 때 주석 행은 숫자 부호(#)로 시작합니다. 데이터 라인에는 지시문과 할당값만 있을 수 있습니다. 주석은 같은 라인에서 허용되지 않습니다. 라인 연속은 지원하지 않습니다.

```
#
# 호스트 기반 클라이언트 제어
#
config.enabled=false
product.enabled=false
reject.config.updates=false
reject.product.updates=falseaccept.product.license=false
primary.system=false
pushtoclient.folder=/var/rdz/pushtoclient
default.store=com.ibm.ftt.configurations.USS
file.permission=RWX.RWX.RX
```

그림 33. pushtoclient.properties: 호스트 기반 클라이언트 제어 구성 파일

config.enabled

호스트 기반 클라이언트 제어가 구성 파일에 사용되는지 여부를 표시합니다. 기본값은 false입니다. 올바른 값은 true, false, LDAP 또는 SAF입니다. 이러한 값의 의미는 134 페이지의 표 16의 내용을 참조하십시오.

product.enabled

호스트 기반 클라이언트 제어가 제품 업데이트에 사용되는지를 표시합니다. 기본값은 false입니다. 올바른 값은 true, false, LDAP 또는 SAF입니다. 이러한 값의 의미는 134 페이지의 표 16의 내용을 참조하십시오.

reject.config.updates

사용자가 클라이언트로 푸시되는 구성 업데이트를 거절할 수 있을지 나타냅니다. 기본값은 false입니다. 올바른 값은 true, false, LDAP 또는 SAF입니다. 이러한 값의 의미는 134 페이지의 표 16의 내용을 참조하십시오.

reject.product.updates

사용자가 클라이언트에 푸시되는 제품 업데이트를 거절할 수 있을지 나타냅니다. 기본값은 false입니다. 올바른 값은 true, false, LDAP 또는 SAF입니다. 이러한 값의 의미는 134 페이지의 표 16의 내용을 참조하십시오.

accept.product.license

클라이언트로 푸시에서 시작된 업데이트 중에 제품 라이선스가 자동으로 승인되는지 여부를 표시합니다. 가능해지면, IBM Installation Manager는 클라이언트 업데이트 도중 라이선스를 승인하도록 요청하지 않습니다. 기본값은 false입니다. 올바른 값은 true 및 false입니다.

primary.system

호스트 기반 클라이언트 제어는 각 시스템에 대한 시스템 고유 데이터 저장을 지원하며, 단일 시스템에 공통 데이터를 유지보수하여 관리 노력을 줄입니다. 이 지시문은 글로벌하고 시스템 공용인 클라이언트 정의를 저장하는 시스템인지 여부를 표시합니다. 기본값은 false입니다. 올바른 값은 true 및 false입니다.

참고: 기본 시스템으로 하나의 시스템만 정의되었는지 확인하십시오. Developer for System z 클라이언트 관리자는 대상 시스템이 기본 시스템인 경우가 아니면 글로벌 구성 데이터를 내보낼 수 없습니다. Developer for System z 클라이언트는 동기화되지 않은 구성으로 여러 기본 시스템에 연결할 때 불규칙적인 동작을 나타낼 수 있습니다.

pushtoclient.folder

호스트 기반 클라이언트 제어 정의를 위한 기본 디렉토리. 기본값은 /var/rdz/pushtoclient입니다.

default.store

호스트 기반 클라이언트 제어는 클라이언트로 푸시되는 데이터를 저장하기 위한 여러 메소드를 지원합니다. 이 지시문은 데이터 액세스에 사용되는 드라이

버 또는 저장소를 식별합니다. 기본값은 `com.ibm.ftt.configurations.USS`이며, z/OS UNIX 플랫 파일에 저장되는 데이터를 지원합니다.

Developer for System z는 `com.ibm.ftt.configurations.USS` 저장소를 제공합니다. 데이터가 기타 위치에 있는 경우 써드파티 저장소가 필요합니다.

file.permission

`com.ibm.ftt.configurations.USS` 저장소는 `file.permission`을 사용하여 저장소에서 작성한 파일의 필수 액세스 권한을 판별합니다. 기본값은 `RWX.RWX.RX`이며, 소유자와 소유자의 기본 그룹에 디렉토리 구조 및 파일에 대한 읽기 및 쓰기 액세스 권한을 허용합니다. 다른 파일과 디렉토리는 디렉토리 구조와 해당 파일에 대한 읽기 액세스 권한만 보유합니다.

UNIX 표준에 따라, 세 가지 유형의 사용자(소유자, 그룹 및 기타)에 대해 권한을 설정할 수 있습니다. `file.permission` 마스크에서 필드는 이 순서와 일치하고 필드는 점(.)으로 구분됩니다. 각 필드는 비어 있거나, 값으로서 R, W, RW, X, RX, WX 또는 RWX를 가지고 있습니다(여기서 R = 읽기, W = 쓰기, X = 디렉토리 콘텐츠 실행 또는 나열).

표 16. 클라이언트로 푸시 그룹 지원

키 값	관련된 클라이언트로 푸시 기능을 사용할 수 있습니까?
false	아니오, 사용할 수 없습니다.
true	예, 모두에서 가능합니다.
LDAP	예, 그러나 사용 가능성은 LDAP 그룹의 멤버십으로 제어됩니다.
SAF	예, 그러나 사용 가능성은 보안 프로파일일에 대한 허용으로 제어됩니다.

참고:

- 호스트 기반 클라이언트 제어를 활성화하려면 `keymapping.xml` 파일이 `/var/rdz/pushtoclient`에 있어야 합니다. 이 파일은 Developer for System z 클라이언트 관리자가 작성하고 유지보수합니다.
- 호스트 기반 프로젝트, 호스트 기반 클라이언트 구성 및 업그레이드 제어에 대한 자세한 정보는 호스트 구성 참조서(SA30-4501)에서 "클라이언트로 푸시 고려사항"을 참조하십시오.
- 파일이 작성될 때, z/OS UNIX에서는 기본적으로 스레즈 작성에 유효한 UID와, 스레드 작성에 유효한 GID가 아닌 소유 디렉토리의 GID(그룹 ID)를 사용합니다. 이 동작의 변경 방법 또는 필요한 GID 할당을 받기 위해 호스트 기반 클라이언트 제어 설정을 조정하는 방법에 대한 자세한 정보는 호스트 구성 참조서(SA30-4501)의 "Z/OS UNIX 디렉토리 구조"를 참조하십시오.

(선택사항) ssl.properties, RSE SSL 암호화

이 사용자 정의 작업을 완료하려면 보안 관리자의 도움이 필요하며, 다음과 같은 자원 또는 특수 사용자 정의 작업이 필요합니다.

- LINKLIST 업데이트
- 프로그램으로 제어되는 데이터 세트를 추가하는 보안 규칙
- (선택사항) SSL의 인증을 추가하는 보안 규칙

외부, 클라이언트-호스트 통신은 SSL(Secure Socket Layer)을 사용하여 암호화할 수 있습니다. 이 기능은 기본적으로 사용되지 않으며 ssl.properties의 설정으로 제어됩니다.

참고: X.509 인증서를 사용한 클라이언트 인증에서는 SSL 암호화 통신을 사용해야 합니다.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 ssl.properties는 /etc/rdz/에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오. TSO **OEDIT** 명령으로 파일을 편집할 수 있습니다. 변경된 내용을 적용하려면 RSE를 다시 시작해야 합니다.

클라이언트는 연결 설정 도중 RSE 디먼과 통신하고 실제 세션 동안은 RSE 서버와 통신합니다. SSL이 사용되는 경우 데이터 스트림 모두가 암호화됩니다.

RSE 디먼과 RSE 서버는 상호 간의 아키텍처 차이로 인해 다른 인증서 저장 메커니즘을 지원합니다. 이는 RSE 디먼과 RSE 서버 모두에 SSL 정의가 필요함을 의미합니다. RSE 디먼과 RSE 서버가 동일한 인증서 관리 메소드를 사용하는 경우에는 공유 인증서를 사용할 수 있습니다.

표 17. SSL 인증서 스토리지 메커니즘

인증서 스토리지	작성자 및 관리자	RSE 디먼	RSE 서버
키 링	SAF 준수 보안 제품	지원됨	지원됨
키 데이터베이스	z/OS UNIX gskkyman	지원됨	/
키 저장소	Java의 keytool	/	지원됨

참고:

- SAF 준수 키 링은 인증서 관리를 위해 선호하는 방법입니다.
- SAF 준수 키 링은 보안 데이터베이스에서 또는 System z 암호 하드웨어에 대한 인터페이스인 ICSF를 사용하여 인증서의 개인 키를 저장할 수 있습니다. ICSF에 대한 액세스는 CSFSERV 보안 클래스에서 프로파일에 의해 보호됩니다.

RSE 디먼은 시스템 SSL 기능을 사용하여 SSL을 관리합니다. 이는 SYS1.SIEALNKE가 보안 소프트웨어에 의해 제어되는 프로그래머가 하며 `rsed.envvars`에 있는 `LINKLIST` 또는 `STEPLIB` 지시문을 사용할 경우 RSE에서 사용 가능해야 함을 의미합니다.

다음 코드 샘플에서는 샘플 `ssl.properties` 파일을 표시하며, 이는 사용자의 시스템 환경에 맞도록 사용자 정의되어야 합니다. US 코드 페이지를 사용할 때 주석 행은 숫자 부호(#)로 시작합니다. 데이터 라인에는 지시문과 할당값만 있을 수 있습니다. 같은 라인에서 주석은 허용되지 않습니다. 라인 연속은 지원하지 않습니다.

```
# ssl.properties - SSL 구성 파일
enable_ssl=false

# 디먼 특성

#daemon_keydb_file=
#daemon_keydb_password=
#daemon_key_label=

# 서버 특성

#server_keystore_file=
#server_keystore_password=
#server_keystore_label=
#server_keystore_type=JCERACFKS
```

그림 34. `ssl.properties` - SSL 구성 파일

디먼 및 서버 특성은 SSL이 사용 가능한 경우에만 설정해야 합니다. SSL 설정에 대한 자세한 정보는 *Developer for System z 호스트 구성 참조서*에 있는 "SSL 및 X.509 인증 설정"의 내용을 참조하십시오.

enable_ssl

SSL 통신을 사용 또는 사용 안함으로 설정하십시오. 기본값은 `false`입니다. 올바른 옵션은 `true` 및 `false`입니다.

daemon_keydb_file

RACF 또는 유사한 보안 제품 키 링 이름입니다. 키 링을 사용하는 대신 **gskkyman**을 사용하여 키 데이터베이스를 작성하는 경우 키 데이터베이스 이름을 제공하십시오. SSL이 사용되는 경우, 이 지시문을 주석 해제한 후 사용자 정의하십시오.

daemon_keydb_password

키 링을 사용하는 경우 주석 처리하거나 공백으로 남겨 두십시오. 그렇지 않으면 키 데이터베이스 비밀번호를 제공하십시오. SSL이 사용되고 **gskkyman** 키 데이터베이스를 사용 중인 경우, 이 지시문을 주석 해제하고 사용자 정의하십시오.

daemon_key_label

기본값으로 정의되지 않은 경우 키 링 또는 키 데이터베이스에서 사용되는 인증 레이블입니다. 기본값이 사용되는 경우에는 주석 처리되어야 합니다. SSL이 사용되고 기본 보안 인증서를 사용하지 않는 경우, 이 지시문을 주석 해제한 후 사용자 정의하십시오. 키 레이블에서는 대소문자를 구분합니다.

server_keystore_file

Java의 **keytool** 명령에서 작성하는 키 저장소 이름 또는, **server_keystore_type=JCERACFKS**의 경우 RACF 또는 유사한 보안 제품 키 링 이름입니다. SSL이 사용되는 경우, 이 지시문을 주석 해제한 후 사용자 정의하십시오.

server_keystore_password

키 링을 사용하는 경우 주석 처리하거나 공백으로 남겨 두십시오. 그렇지 않으면 키 저장소 비밀번호를 제공하십시오. SSL이 사용되고 **keytool** 키 저장소를 사용 중인 경우 이 지시문을 주석 해제한 후 사용자 정의하십시오.

server_keystore_label

키 링 또는 키 저장소에서 사용되는 인증 레이블입니다. 기본값은 발견된 첫 번째 유효 인증서입니다. SSL이 사용되고 기본 보안 인증서를 사용하지 않는 경우, 이 지시문을 주석 해제한 후 사용자 정의하십시오. 키 레이블에서는 대소문자를 구분합니다.

server_keystore_type

키 저장소 유형입니다. 기본값은 JKS입니다. 올바른 값은 다음과 같습니다.

표 18. 유효한 키 저장소 유형

키워드	키 저장소 유형
JKS	Java 키 저장소
JCERACFKS	SAF 준수 키 링, 여기서 인증서의 개인 키는 보안 데이터베이스에 저장됩니다.
JCECCARACFKS	SAF 준수 키 링, 여기서 인증서의 개인 키는 System z 암호 하드웨어에 대한 인터페이스인 ICSF를 사용하여 저장됩니다.

참고: 공개 시 IBM z/OS Java에는 JCECCARACFKS를 지원하는 /usr/lpp/java/J6.0/lib/security/java.security 파일의 업데이트가 필요합니다. 다음 라인이 추가되어야 합니다.

```
security.provider.1=com.ibm.crypto.hdwrCCA.provider.IBMJCECCA
```

결과 파일은 다음과 유사합니다.

```
security.provider.1=com.ibm.crypto.hdwrCCA.provider.IBMJCECCA
security.provider.2=com.ibm.jsse2.IBMJSSEProvider2
security.provider.3=com.ibm.crypto.provider.IBMJCE
```

```
security.provider.4=com.ibm.security.jgss.IBMJGSSProvider
security.provider.5=com.ibm.security.cert.IBMCertPath
security.provider.6=com.ibm.security.sasl.IBMSASL
```

(선택사항) rsecomm.properties, RSE 추적

이 사용자 정의 태스크는 도움, 특수 자원 또는 특수 사용자 정의 태스크가 필요하지 않습니다.

Developer for System z는 문제점 해결을 위해 다양한 레벨의 내부 프로그램 플로우 추적을 지원합니다. RSE 및 RSE에서 호출하는 일부 서비스는 rsecomm.properties의 설정을 사용하여 출력 로그의 필수 초기 세부사항 레벨을 알 수 있습니다.

경고: 이 설정을 변경하면 성능이 저하될 수 있으므로 IBM 지원 센터의 지시 하에서만 수행해야 합니다.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 rsecomm.properties는 /etc/rdz/에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오. TSO **OEDIT** 명령으로 파일을 편집할 수 있습니다.

다음 코드 샘플은 rsecomm.properties 파일을 보여주며, 사용자의 추적 요구사항에 맞도록 사용자 정의될 수 있습니다. US 코드 페이지를 사용할 때 주석 행은 숫자 부호(#)로 시작합니다. 데이터 라인에는 지시문과 할당값만 있을 수 있습니다. 같은 라인에서 주석은 허용되지 않습니다. 라인 연속은 지원하지 않습니다.

```
# server.version - 수정하지 마십시오!
server.version=5.0.0
```

```
# 로깅 레벨
# 0 - 오류 메시지 기록
# 1 - 오류 및 경고 메시지 기록
# 2 - 오류, 경고 및 정보 메시지 기록
debug_level=1#USER=userid
#USER=(userid,userid,...)
```

그림 35. rsecomm.properties - 로깅 구성 파일

server.version

로깅 서버 버전. 기본값은 5.0.0입니다. 수정하지 마십시오.

debug_level

출력 로그의 세부사항 레벨. 기본값은 1이며, 오류 및 경고 메시지를 기록합니다. debug_level 세부사항은 여러 서비스의 세부사항 레벨 및 여러 출력 과일을 제어합니다. 세부사항 레벨을 높이면 성능이 저하되므로 IBM 지원 센터

의 지시 하에서만 수행해야 합니다. 이 지시문이 제어하는 로그에 대한 자세한 정보는 **호스트 구성 참조서 (SA30-4501)**에 있는 "RSE 추적"의 내용을 참조하십시오.

올바른 값은 다음과 같습니다.

0	오류 메시지만 기록합니다.
1	오류 및 경고 메시지를 기록합니다.
2	오류, 경고 및 정보 메시지를 기록합니다.

참고: 209 페이지의 제 11 장 『운영자 명령』에서 설명한 것처럼, `debug_level` 은 **modify rsecommlog**, **modify rseserverlog**, 및 **modify rsedaemonlog** 운영자 명령과 함께 특정 로그 파일에 대해 동적으로 변경될 수 있습니다.

| **USER** 서버를 시작하는 동안 지정된 사용자 ID에 대해 디버그 레벨 2(오류, 경고 및
 | 정보 메시지)를 설정합니다. 모든 기타 사용자에게 대한 디버그 레벨은
 | `debug_level` 지시문에서 지정된 대로 기본값입니다. **USER** 지시문은 RSE 서
 | 버(`rsecomm.log`) 및 MVS 데이터 세트 서비스(`lock.log` 및 `ffs*.log`)의 추
 | 적 세부사항 레벨을 변경하며 **modify trace user** 운영자 명령을 실행하는 것
 | 과 동일합니다.

(선택사항) include.conf, C/C++ 콘텐츠 지원을 위해 강제로 포함됨

이 사용자 정의 태스크는 도움, 특수 자원 또는 특수 사용자 정의 태스크가 필요하지 않습니다.

C/C++용 콘텐츠 지원은 `include.conf`의 정의를 사용하여 지정된 파일 또는 멤버를 강제로 포함할 수 있습니다. 강제 포함은 해당 파일 또는 멤버가 프리프로세서 지시문을 사용하여 소스 코드에 포함되었는지 여부에 관계없이 콘텐츠 지원 오퍼레이션이 수행될 때 구문 분석할 파일이나 디렉토리, 데이터 세트 또는 데이터 세트 멤버로 구성됩니다.

파일은 사용되기 전에 `include.c` 또는 `include.cpp` 변수가 `rsed.envvars`에서 참조해야 합니다. `rsed.envvars`에서 이렇게 참조한다는 것은 C 및 C++에서 사용할 경우 다른 파일을 지정할 수 있음을 의미합니다. `rsed.envvars`의 변수는 기본적으로 사용 안함으로 설정됩니다.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 샘플 `include.conf`는 `/etc/rdz/`에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오. **TSO OEDIT** 명령으로 파일을 편집할 수 있습니다.

정의는 1열에서 시작되어야 합니다. **US** 코드 페이지를 사용할 때 주석 라인은 파운드 기호(#)로 시작합니다. 데이터 라인에는 디렉토리, 파일, 데이터 세트 또는 멤버의 이름

만 있을 수 있습니다. 주석은 같은 라인에서 허용되지 않습니다. 라인 연속은 지원하지 않습니다.

```
# /usr/include 디렉토리에 stdio.h 파일을 포함시키려면 다음을 입력하십시오.
# /usr/include/stdio.h
#
# /usr/include 디렉토리의 모든 파일과 그 하위 디렉토리의 모든 파일을
# 포함시키려면 다음을 입력하십시오.
# /usr/include
#
# 선택된 파일 유형(대소문자 구분)으로 z/OS UNIX 와일드카드 포함을 제한하려면
# FILETYPES 변수를 주석 해제 후 사용자 정의하십시오.
# 파일 유형은 쉼표로 구분되는 목록(공백 없음)에서 지정됩니다.
# FILETYPES=H,h,hpp,C,c,cpp,cxx

# CBC.SCLBH.H 데이터 세트의 모든 멤버를 포함시키려면 다음을 입력하십시오.
# //CBC.SCLBH.H
#
# CBC.SCLBH.H 데이터 세트의 STDIOSTR 멤버를 포함시키려면 다음을 입력하십시오.
# //CBC.SCLBH.H(STDIOSTR)
# 샘플 목록에는 공통적으로 사용되는 몇 가지 C 표준 라이브러리 파일이 있습니다.
/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
```

그림 36. *include.conf* - C/C++ 콘텐츠 지원을 위한 강제 포함

(선택사항) z/OS UNIX 서브프로젝트

이 사용자 정의 태스크는 도움, 특수 자원 또는 특수 사용자 정의 태스크가 필요하지 않습니다.

REXEC(Remote Execution)은 클라이언트가 호스트 시스템에서 명령을 실행할 수 있도록 하는 TCP/IP 서비스입니다. SSH(Secure Shell)도 유사한 서비스이지만, SSL(Secure Socket Layer)을 사용하여 모든 통신을 암호화합니다. Developer for System z는 z/OS UNIX 서브프로젝트에서 원격(호스트 기반) 조치를 수행하는 데 이러한 서비스 중 하나를 사용합니다.

참고:

- Developer for System z는 TSO 버전이 아니라 REXEC의 z/OS UNIX 버전을 사용합니다.
- REXEC/SSH가 기본 포트를 사용하도록 구성되지 않으면, Developer for System z 클라이언트는 z/OS UNIX 서브프로젝트에서 사용할 올바른 포트를 정의해야 합니다. 이 구성은 창 > 환경 설정 > z/OS 솔루션 > USS 서브프로젝트 > 원격 조치 옵션 환경 설정 페이지를 선택하여 완료할 수 있습니다. 사용할 포트를 알려면 『REXEC 또는 SSH 설정』의 내용을 참조하십시오.

REXEC 또는 SSH 설정

REXEC 및 SSH는 다른 TCP/IP 서비스인 INETD(Internet Daemon)에서 제공하는 서비스를 사용합니다. *Communications Server IP Configuration Guide(SC31-8775)*에서는 INETD, REXEC 및 SSH 설정에 필요한 단계에 대해 설명합니다. 자세한 내용 및 대체 설정 메소드는 Developer for System z 라이브러리(<http://www-01.ibm.com/support/docview.wss?uid=swg27038517>)에 있는 *Rational Developer for System z와 INETD, REXEC 및 SSH 사용(SC14-7301)* 백서를 참조하십시오.

REXEC에서 사용하는 공통 포트는 512입니다. 사용되는 포트를 확인하려면 `/etc/inetd.conf` 및 `/etc/services`를 확인하십시오.

- `/etc/inetd.conf`에서 `rexecd` 서버(7번째 단어)의 서비스 이름(첫 번째 단어, 이 예제에서는 `exec`)을 찾으십시오.

```
exec stream tcp nowait OMVSKERN /usr/sbin/orexecd rexecd -LV
```

- `/etc/services`에서 이 서비스 이름(첫 번째 단어)에 첨부되어 있는 포트(두 번째 단어, 이 예에서 512)를 찾으십시오.

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

똑같은 원리가 SSH에 적용됩니다. 공통 포트는 22이고, 서버 이름은 `sshd`입니다.

(선택사항) 프리프로세서 지원 포함

이 사용자 정의 태스크는 도움, 특수 자원 또는 특수 사용자 정의 태스크가 필요하지 않습니다.

Developer for System z는 select third-party include 문을 포함한 COBOL 및 PL/I include 문을 해석하고 확장하도록 지원합니다. Developer for System z도 샘플 REXX `exec`, `FEKRNPLI`를 제공하며, 이는 Developer for System z 클라이언트가 PL/I 컴파일러를 호출함으로써 PL/I 소스를 확장할 때 사용할 수 있습니다.

`FEK.SFEKSAMP(FEKSETUP)` 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 `FEKRNPLI`는 `FEK.#CUST.CNTL`에 있습니다. 자세한 정보는 17 페이지의 제 2 장 『기본 사용자 정의』의 내용을 참조하십시오.

멤버 내에 설명된 대로 샘플 FEK.#CUST.CNTL(FEKRNPLI) exec를 사용자 정의하십시오. 다음 정보를 제공해야 합니다.

- compiler_hlq: PL/I 컴파일러를 위한 상위 레벨 규정자

Developer for System z 클라이언트는 TSO 명령 서비스를 사용하여 exec를 실행합니다. 이는 TSO 명령 서비스의 경우 FEKRNPLI exec가 SYSPROC 또는 SYSEXEC 접합에 위치하면 사용자가 exec의 정확한 위치를 알 필요가 없음을 의미합니다. 사용자는 이름만 알면 됩니다. 기본적으로 TSO 명령 서비스는 ISPF Client Gateway를 사용하여 TSO 환경을 작성하지만, *Using APPC to provide TSO command services* (SC14-7291) 백서에 설명한 대로 APPC도 지원합니다. ISPF Client Gateway를 사용할 때, SYSPROC 또는 SYSEXEC 접합은 ISPF.conf에서 정의됩니다. 이 파일의 사용자 정의에 대한 자세한 정보는 64 페이지의 『ISPF.conf, ISPF의 TSO/ISPF Client Gateway 구성 파일』의 내용을 참조하십시오.

(선택사항) Enterprise COBOL 및 PL/I용 xUnit 지원

이 사용자 정의 태스크는 도움이 필요하지 않으나, 다음과 같은 자원 또는 특수 사용자 정의 태스크가 필요합니다.

- LINKLIST 업데이트
-

개발자가 반복 가능한 자체 검사 단위 테스트를 수행하는 코드를 작성하도록 지원하는 프레임워크를 종합하여 xUnit이라고 합니다. Developer for System z는 이러한 프레임워크를 Enterprise COBOL 및 PL/I 코드의 단위 테스트에(zUnit이라고 함) 제공합니다.

zUnit 프레임워크를 사용하려면 개발자가 STEPLIB 또는 LINKLIST를 통해 FEK.SFEKLOAD 로드 라이브러리에서 AZU* 및 IAZU* 로드 모듈에 액세스해야 합니다. zUnit 테스트 사용자인 AZUTSTRN은 STEPLIB 또는 LINKLIST를 통해 여러 시스템 라이브러리에 액세스해야 합니다.

- CEE.SCEERUN 및 CEE.SCEERUN2(LE 런타임)
- SYS1.CSSLIB(호출 가능한 시스템 서비스)
- SYS1.SIXMOD1(XML 툴킷)

zUnit 테스트 사용자도 다른 테스트 경우가 있는 로드 라이브러리에 대해 액세스해야 합니다. 이 라이브러리는 개발자의 경우에만 해당될 수 있습니다.

zUnit 테스트 사용자, AZUTSTRN은 일괄처리 모드에서, TSO 명령행에서 그리고 z/OS UNIX 명령행에서 Developer for System z 클라이언트가 호출할 수 있습니다.

- Developer for System z에서는 샘플 프로시저 AZUZUNIT를 제공하여 일괄처리 모드에서 zUnit 테스트 사용자 호출을 단순화시킵니다. FEK.SFEKSAMP(FEKSETUP) 작

업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 AZUZUNIT는 FEK.#CUST.PROCLIB에 있습니다. 자세한 정보는 17 페이지의 제 2 장 『기본 사용자 정의』의 내용을 참조하십시오.

멤버 내에 설명된 대로 샘플 프로시저 FEK.#CUST.PROCLIB(AZUZUNIT)를 사용자 정의한 후 SYS1.PROCLIB로 복사하십시오.

프로시저의 이름과 프로시저에 있는 단계의 이름은 Developer for System z 클라이언트에 포함된 기본 특성과 일치합니다. 프로시저의 이름 또는 프로시저에 있는 단계의 이름이 변경되면, 모든 클라이언트에서 상응하는 특성 파일을 업데이트해야 합니다. 프로시저와 단계 이름을 변경해서는 안 됩니다.

AZUZUNIT 프로시저를 시스템 프로시저 라이브러리로 복사할 수 없으면, Developer for System z 사용자에게 JOB 카드 바로 다음의 JCLLIB 카드를 호출 작업에 추가하도록 요청하십시오.

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

- z/OS UNIX에서 (/usr/lpp/rdz/bin/zunit 스크립트를 사용하여) zUnit 테스트 사용자를 호출하는 경우, rsed.envvars의 STEPLIB 지시문에서 필요한 비LINKLIST 데이터 세트를 지정하면 개발자 설정이 단순해질 수 있습니다.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 rsed.envvars는 /etc/rdz/에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오. TSO **OEDIT** 명령으로 파일을 편집할 수 있습니다.

zunit 스크립트를 사용하면 스크립트에서 사용되는 STEPLIB 지시문에 추가될 데이터 세트를 사용자가 지정할 수 있습니다.

- FEK.SFEKPROC(FEKZUNIT) exec를 사용하여 TSO 명령행에서 zUnit 테스트 사용자를 호출하는 경우, 시스템 라이브러리는 LINKLIST에 있어야 합니다. 그렇지 않으면 개발자가 zUnit 테스트 사용자의 모든 호출 인스턴스에 대해 시스템 데이터 세트 이름을 지정해야 합니다. 데이터 세트의 **TSOLIB** 할당을 수행하는 랩퍼 exec도 작성할 수 있습니다. 이 랩퍼 exec 코딩 방법의 예로서 FEKZUNIT 자체를 사용할 수 있습니다.

zUnit 테스트 사용자는 테스트 보고서의 자동 재형식화를 허용합니다. Developer for System z에서는 기본 /usr/lpp/rdz 위치에 Developer for System z를 설치한 경우 /usr/lpp/rdz/samples/zunit/xsd 및 /usr/lpp/rdz/samples/zunit/xsl에 있는 샘플 변환(예: Ant 또는 junit 형식으로서의 변환)을 제공합니다.

(선택사항) 엔터프라이즈 서비스 도구 지원

이 사용자 정의 태스크는 도움, 특수 자원 또는 특수 사용자 정의 태스크가 필요하지 않습니다.

Developer for System z 클라이언트에는 엔터프라이즈 서비스 도구는 코드 생성 컴포넌트가 있습니다. 생성되는 코드의 유형에 따라, 이 코드는 Developer for System z 호스트 시스템 설치에서 제공하는 기능에 의존합니다. 이러한 호스트 시스템 기능을 이용 가능하게 설정하는 작업에 대해서는 다음 절에서 설명합니다.

- 119 페이지의 제 5 장 『(선택사항) 애플리케이션 배치 관리자(더 이상 사용되지 않음)』
- 『(선택사항) CICS 양방향 언어 지원』
- 145 페이지의 『(선택사항) 생성된 코드의 진단 IRZ 메시지』

참고: 엔터프라이즈 서비스 도구에는 서비스 플로우 모델러(SFM) 및 엔터프라이즈용 XML 서비스와 같은 여러 도구가 포함됩니다.

(선택사항) CICS 양방향 언어 지원

이 사용자 정의 태스크를 완료하려면 CICS 관리자의 도움이 필요하며, 다음 자원 또는 특수 사용자 정의 태스크가 필요합니다.

- CICS 리전 JCL 업데이트
 - CICS로 프로그램 정의
-

Developer for System z 엔터프라이즈 서비스 도구 컴포넌트는 서로 다른 형식의 아랍어와 히브리어 인터페이스 메시지, 양방향 데이터 프리젠테이션 및 모든 편집기와 보기에서의 편집을 지원합니다. 터미널 애플리케이션에서는 왼쪽에서 오른쪽으로 그리고 오른쪽에서 왼쪽으로 화면 모두가 지원되며 숫자 필드와 역상 화면 방향 필드도 지원됩니다.

추가적 양방향 기능에는 다음이 포함됩니다.

- 엔터프라이즈 서비스 도구 서비스 요청자는 인터페이스 메시지의 양방향 속성을 동적으로 지정합니다.
- 서비스 플로우의 양방향 데이터 처리는 양방향 속성(텍스트 유형, 텍스트 방향, 숫자 스와핑 및 동기식 스와핑)을 기초로 합니다. 이러한 속성은 인터페이스 및 터미널 플로우 둘 다의 다양한 플로우 작성 단계에서 지정할 수 있습니다.
- 엔터프라이즈 서비스 도구 생성 런타임 코드에는 서로 다른 양방향 속성을 가진 메시지 필드 사이의 데이터 변환이 포함됩니다.

또한 엔터프라이즈 서비스 도구 생성 코드는 CICS SFR(Service Flow Runtime) 이외의 환경에서 양방향 변환을 지원할 수 있습니다. 한 가지 예가 일괄처리 애플리케이션입니다. 엔터프라이즈 서비스 도구 생성 마법사에서 해당하는 양방향 변환 옵션을 지정하고 생성된 프로그램을 해당하는 양방향 변환 라이브러리, FEK.SFEKLOAD와 링크하여 양방향 변환 루틴에 대한 호출이 포함되도록 엔터프라이즈 서비스 도구 생성기를 작성할 수 있습니다.

CICS 양방향 언어 지원을 활성화하려면 다음 태스크를 수행하십시오.

1. FEK.SFEKLOAD 로드 모듈 FEJBDCMP와 FEJBDTRX를 CICS RPL 접합(DD 명령문 DFHRPL)에 배치하십시오. 적용된 유지보수가 CICS에서 자동으로 사용될 수 있도록 접합에 설치 데이터 세트를 추가하여 배치해야 합니다.

중요사항: 설치 데이터 세트를 연결하지 않지만 새 데이터 세트 또는 기존 데이터 세트로 모듈을 복사하는 경우, 그 모듈이 DLL이고 PDSE 라이브러리에 상주해야 함을 기억하십시오.

2. 다음과 같은 적절한 CEDA 명령을 이용하여 CICS에 프로그램으로서 FEJBDCMP와 FEJBDTRX를 정의하십시오.

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

(선택사항) 생성된 코드의 진단 IRZ 메시지

이 사용자 정의 태스크는 도움이 필요하지 않으나, 다음과 같은 자원 또는 특수 사용자 정의 태스크가 필요합니다.

- LINKLIST 업데이트
 - CICS 리전 JCL 업데이트
-

System z 클라이언트에는 엔터프라이즈 서비스 도구라는 코드 생성 컴포넌트가 있습니다. 진단 오류 메시지를 발행하기 위해 엔터프라이즈 서비스 도구에서 생성하는 코드의 경우, FEK.SFEKLMOD 로드 라이브러리에 있는 모든 IRZM* 및 IIRZ* 모듈이 생성된 코드에서 사용할 수 있어야 합니다. 엔터프라이즈 서비스 도구는 다음과 같은 환경에 대해 코드를 생성할 수 있습니다.

- CICS
- IMS
- MVS 일괄처리

생성된 코드가 CICS 트랜잭션에서 실행되는 경우, FEK.SFEKLMOD의 IRZM* 및 IIRZ* 모듈 모두를 CICS 리전의 DFHRPL DD에 추가하십시오. 적용된 유지보수가 자동으로 사용 가능하도록 접합에 설치 데이터 세트를 추가하여 이를 수행해야 합니다.

기타 모든 상황에서는 FEK.SFEKLMOD의 IRZ* 및 IIRZ* 모듈 모두를 STEPLIB 또는 LINKLIST를 통해 사용 가능하도록 설정하십시오. 적용된 유지보수가 자동으로 사용 가능하도록 접합에 설치 데이터 세트를 추가하여 이를 수행해야 합니다.

STEPLIB를 사용하는 경우, 코드를 실행하는 태스크의 STEPLIB 지시문에서 LINKLIST를 통해 사용할 수 없도록 모듈을 정의하십시오.

로드 모듈을 사용할 수 없고 생성된 코드에서 오류가 발생하면, 다음 메시지가 발행됩니다.

IRZ9999S Failed to retrieve the text of a Language Environment runtime message. Check that the Language Environment runtime message module for facility IRZ is installed in DFHRPL or STEPLIB.

참고:

- FEK.SFEKLMOD(IRZPWSIO) 모듈은 하향식 IMS MPP 코드 생성 동안 정적으로 링크됩니다. 그러므로, 모듈은 생성된 코드의 런타임 도중 사용할 수 없어야 합니다. 컴파일 도중에만 사용할 수 있어야 합니다.
- 버전 8.5에서 IRZ* 및 IIRZ*의 로드 모듈과 진단 메시지는 FEK.SFEKLOAD 로드 라이브러리에서 FEK.SFEKLMOD로 이동되었습니다.
- 버전 9.0.1에서 FEK.SFEKLMOD(IRZPWSIO) 및 연관된 FEK.SFEKSAMP(IRZPWSH) 샘플 PL/I에는 Developer for System z에서 IMS 버전 12로 이동된 멤버가 포함됩니다. 파트는 각각 IMS.SDFSRESL(DFSPWSIO) 및 IMS.SDFSMSPL(DFSPWSH)로 이름이 바뀝니다.

(선택사항) 통합 디버거

이 사용자 정의 태스크를 완료하려면 보안, TCP/IP 및 CICS 관리자의 도움이 필요하며, 다음 자원 또는 특수 사용자 정의 태스크가 필요합니다.

- LINKLIST 업데이트
- APF 권한 부여
- 시작 태스크 정의
- 보안 프로파일 및 액세스 목록 정의
- 클라이언트-호스트 및 호스트-한정 통신에 대한 TCP/IP 예약
- (선택사항) SVC 추가(IPL 필수)
- (선택사항) SVC용 LPA 업데이트
- (선택사항) CICS 리전 JCL 업데이트
- (선택사항) CICS CSD 업데이트

Developer for System z 통합 디버거 호스트 컴포넌트를 사용하면 버전 9.0.1 이상의 클라이언트가 읽기 전용 메모리에 로드된 CICS 트랜잭션 등의 다양한 LE(Language Environment) 기반 애플리케이션을 디버깅할 수 있습니다.

통합 디버거 데이터 플로우의 개요에 대해서는 호스트 구성 참조서(SA30-4501)의 "Developer for System z 이해" 장의 "통합 디버거" 절을 참조하십시오.

사용자 사이트에서 통합 디버거 사용을 시작하려면 다음 태스크를 수행해야 합니다. 특별히 언급되지 않는 한 모든 태스크가 필수입니다.

1. 통합 디버거의 경우, 선택적 DBGMGR 시작 태스크가 (필수 RSED 시작 태스크와 함께) 활성화 상태여야 합니다. DBGMGR 시작 JCL에 대해서는 28 페이지의 『DBGMGR, 디버그 관리자 시작 태스크』의 내용을 참조하십시오.
2. 통합 디버거 구성은 DBGMGR 시작 태스크의 시작 인수에서 관리합니다. 자세한 정보는 148 페이지의 『통합 디버거 구성 매개변수』의 내용을 참조하십시오.
3. DBGMGR 시작 태스크를 수행하려면 FEK.SFEKAUTH 라이브러리가 APF 인증을 받아야 합니다. 자세한 정보는 148 페이지의 『통합 디버거 parmlib 업데이트』의 내용을 참조하십시오.
4. 통합 디버거는 애플리케이션에서 액세스할 수 있어야 하며 STEPLIB 또는 LINKLIST 업데이트가 필요합니다. 자세한 정보는 148 페이지의 『통합 디버거 parmlib 업데이트』의 내용을 참조하십시오.
5. 통합 디버거를 사용하려면 디버깅 중인 애플리케이션의 사용자 ID에 유효한 OMVS 세그먼트가 있어야 합니다. 자세한 정보는 149 페이지의 『통합 디버거 보안 업데이트』의 내용을 참조하십시오.
6. DBGMGR 시작 태스크에는 몇 가지 보안 허용이 필요합니다. 자세한 정보는 149 페이지의 『통합 디버거 보안 업데이트』의 내용을 참조하십시오.

다음 단계는 CICS 트랜잭션을 디버깅하는 경우에만 필요합니다.

1. 통합 디버거는 CICS 트랜잭션을 디버깅할 수 있습니다. 이 기능을 사용하려면 통합 디버거가 CICS로 정의되어야 합니다. 자세한 정보는 149 페이지의 『통합 디버거 CICS 업데이트』의 내용을 참조하십시오.
2. (선택사항) 통합 디버거는 읽기 전용 메모리에 로드된 CICS 트랜잭션을 디버깅할 수 있습니다. 이 경우 Developer for System z 수퍼바이저 호출(SVC)이 시스템에 정의되어 있어야 합니다. 관련된 로드 모듈이 IPL 시에 LPA에 로드되어야 합니다. 자세한 정보는 148 페이지의 『통합 디버거 parmlib 업데이트』의 내용을 참조하십시오.

SVC가 문제-상태(승인되지 않음) 환경에서 사용되는 경우, 보안 프로파일에 대해 사용자가 허용되어야 합니다. 자세한 정보는 149 페이지의 『통합 디버거 보안 업데이트』의 내용을 참조하십시오.

지정된 애플리케이션 또는 CICS 리전에서 통합 디버거와 같은 하나의 LE(언어 환경 기반 디버거만이 활성화될 수 있습니다. 디버거가 LE 기반의 디버거인지 여부를 나타내는 표시는 애플리케이션에서 사용 가능해야 하는 별명 또는 CEEEVDBG 로드 모듈을 제공하는 것입니다.

통합 디버거 구성 매개변수

통합 디버거를 사용하면 DBGMGR 시작 JCL에서 다음 변수를 구성할 수 있습니다. DBGMGR 시작 JCL에 대해서는 28 페이지의 『DBGMGR, 디버그 관리자 시작 태스크』의 내용을 참조하십시오.

- 시간대 오프셋: 기본값(EST5DST)
- 외부(클라이언트-호스트) 통신에 사용하는 포트, 기본값 5335
- 내부(호스트 한정) 통신에 사용하는 포트, 기본값 5336
- 읽기 전용 CICS 트랜잭션을 디버깅하는 데 사용하는 SVC 번호, 기본값 251
- 로드 라이브러리의 상위 레벨 규정자(기본값 FEK)

통합 디버거 parmlib 업데이트

- DBGMGR 시작 태스크를 수행하려면 FEK.SFEKAUTH 라이브러리가 APF 인증을 받아야 합니다. 자세한 정보는 23 페이지의 『PROGxx에 APF 권한 부여』의 내용을 참조하십시오.
- LE(Language Environment)는 통합 디버거를 호출할 수 있어야 합니다. 따라서 FEK.SFEKAUTH 라이브러리는 디버깅할 애플리케이션의 LINKLIST 또는 STEPLIB에 있어야 합니다. 자세한 정보는 24 페이지의 『PROGxx의 LINKLIST 정의』의 내용을 참조하십시오.

참고: LINKLIST를 사용하는 경우에는 FEK.SFEKAUTH가 CEEEVDBG 또는 CEEEV006 로드 모듈을 보유하는 기타 LE 기반 디버거의 라이브러리의 앞에 있도록 해야 합니다. 예를 들어, z/OS용 IBM 디버그 도구는 hlq.SEQA* 라이브러리를 사용합니다.

- 통합 디버거는 z/OS Binder API를 사용합니다. 이 API는 z/OS 1.10부터 /usr/lib/iewbndd.so로 사용 가능하며 z/OS 1.13부터는 SYS1.SIEAMIGE(IEWBNDD)로도 사용 가능합니다. 따라서 z/OS 1.13 이상의 경우 SYS1.SIEAMIGE가 LINKLIST(또는 STEPLIB)에 있어야 합니다. 자세한 정보는 25 페이지의 『필수 LINKLIST 및 LPA 정의』의 내용을 참조하십시오.

참고: SYS1.SIEAMIGE가 z/OS 1.13 이상의 시스템에 있는 LINKLIST 또는 STEPLIB에 없으면 통합 디버거가 다음 메시지를 발행하며 /usr/lib/iewbndd.so를 사용합니다.

CEE3501S The module //IEWBNDD was not found

- 통합 디버거는 읽기 전용 메모리에 로드된 CICS 트랜잭션을 디버깅할 수 있습니다. 이 경우 Developer for System z 수퍼바이저 호출(SVC)이 시스템에 정의되어 있어야 합니다. 기본 SVC 번호는 251입니다. 관련 로드 모듈 FEK.SFEKLPA(AQESVC01)는 IPL 시에 LPA에 로드되어야 합니다. 세부사항은 22 페이지의 『IEASVCxx의 SVC 정의』 및 22 페이지의 『LPALSTxx의 LPA 정의』의 내용을 참조하십시오.

통합 디버거 TCP/IP 업데이트

통합 디버거는 두 개의 TCP/IP 포트를 사용합니다. 세부사항은 호스트 구성 참조 안내서(SA30-4501)의 "TCP/IP 고려사항"을 참조하십시오.

- 클라이언트-호스트 통신용 포트(기본값: 5335). 이 포트에서는 통신이 암호화될 수 있습니다.
- 호스트 한정 통신용 포트(기본값: 5336)

통합 디버거 보안 업데이트

통합 디버거에는 다음과 같은 보안 정의가 필요합니다. 자세한 정보는 34 페이지의 『보안 정의』의 내용을 참조하십시오.

- 애플리케이션을 실행 중인 사용자 ID의 OMVS 세그먼트를 디버깅하는 중입니다.
- DBGMR 시작 태스크
- 시작 태스크 사용자 ID에 대한 BPX.SERVER 허용
- 시작 태스크 로드 라이브러리에 대한 프로그램 제어

통합 디버거는 필요에 따라 다음과 같은 보안 정의를 사용합니다. 세부사항은 호스트 구성 참조 안내서(SA30-4501)의 "보안 고려사항"을 참조하십시오.

- 사용자가 문제-상태(승인되지 않음) 환경에서 읽기 전용 메모리에 로드된 CICS 트랜잭션을 디버깅하기 위한 AQE.AUTHDEBUG.WRITEBUFFER 허용

참고: 통합 디버거 를 사용하지 않고 기존 Developer for System z 설치에서 간단히 마이그레이션하도록 샘플 JCL FEK.SFEKSAMP(AQERACF)가 통합 디버거 관련 보안 정의만 지정하기 위해 RACF 명령과 함께 제공됩니다.

통합 디버거 CICS 업데이트

CICS 트랜잭션을 디버깅하려면 통합 디버거에 다음과 같은 CICS 업데이트가 필요합니다.

- CICS JCL 업데이트:
 - 라이브러리가 LINKLIST에 없는 경우에 리전의 DFHRPL DD문에서 FEK,SFEKAUTH 로드 라이브러리를 정의합니다.
 - z/OS 1.13 이상의 경우, 라이브러리가 LINKLIST에 없으면 리전의 STEPLIB DD문에서 SYS1.SIEAMIGE 로드 라이브러리를 정의합니다. 자세한 내용은 148 페이지의 『통합 디버거 parmlib 업데이트』의 z/OS Binder API 정보를 참조하십시오.
- CICS CSD 업데이트:

AQECSD 샘플 CSD 업데이트 작업에서 설명한 대로 디버거를 CICS 리전에 정의하십시오. FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치

를 지정하지 않은 한, AQECSD는 FEK.#CUST.JCL에 있습니다. 자세한 내용은 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오.

(선택사항) 문제점 판별 도구 지원

이 사용자 정의 태스크는 도움, 특수 자원 또는 특수 사용자 정의 태스크가 필요하지 않습니다.

Developer for System z는 다양한 IBM z/OS 문제점 판별 도구와 통합할 수 있습니다. 다음 절에서는 Developer for System z 클라이언트에서 이러한 도구를 이용 가능하게 하는 방법에 대해 기술합니다.

- z/OS용 IBM 디버그 도구: 『(선택사항) DB2 및 IMS 디버그 지원』의 내용을 참조하십시오. Developer for System z 버전 9.0.1에서는 디버그 도구 대신 사용할 수 있는 통합 디버거를 제공합니다.
- z/OS용 IBM 파일 관리자: 151 페이지의 『(선택사항) 파일 관리자 지원』의 내용을 참조하십시오.
- z/OS용 IBM Fault Analyzer: Developer for System z 호스트 시스템 구성이 필요하지 않습니다. 버전 9.0 이후로, Developer for System z가 더 이상 결함 분석기 통합(FAI)을 지원하지 않는다는 것에 주목하십시오. 이 지원을 계속 받는 이전 클라이언트는 기능을 설치 제거하고 Eclipse용 IBM Fault Analyzer 플러그인을 설치해야 합니다. 이 플러그인은 IBM 문제점 판별 도구 플러그인 웹 페이지 (<http://www-01.ibm.com/software/awdtools/deployment/pdtpplugins/>)에서 사용 가능합니다.

(선택사항) DB2 및 IMS 디버그 지원

이 사용자 정의 태스크에는 Developer for System z 구성을 위한 도움, 특수 자원 또는 특수 사용자 정의 태스크가 필요하지 않습니다. 그러나 z/OS용 IBM 디버그 도구 구성에 대한 요구사항이 있습니다.

z/OS용 IBM 디버그 도구는 사용자 정의된 LE(Language Environment) 사용자 종료(CEEEXITA)를 제공하며 이는 IMS 및 DB2 스토어드 프로시저의 LE 초기화 논리에서 호출되는 경우 TEST 런타임 옵션을 리턴합니다. z/OS용 IBM 디버그 도구는 문제점 판별 도구 공통 컴포넌트 서버에 디버그 도구 확장을 제공하여 z/OS 시스템에서 TEST 런타임 옵션 데이터 세트를 작성 및 관리합니다. Developer for System z에서는 IMS 및 DB2 스토어드 프로시저 런타임의 디버그 프로파일 관리에 z/OS용 IBM 디버그 도구 지원을 사용하고 개선할 수 있습니다.

z/OS용 IBM 디버그 도구 문서에서 필요한 설정에 대해 상세히 설명하므로 여기에서는 간단히 설명합니다.

- 언어 환경 사용자 종료(h1q.SEQA*)를 통해 TEST 런타임 옵션 지정

- DTSP 프로파일 보기를 위한 지원 추가
 - 문제점 판별 도구 공통 컴포넌트 서버(hlq.SIPV*, 작업 IPVGSVRJ) 설치
 - hlq.SEQA* 문제점 판별 도구 공통 컴포넌트의 디버그 도구 확장 설치 및 구성

참고:

- z/OS용 IBM 디버그 도구 제품은 별도로 확보, 설치 및 구성되어야 합니다. 이 매뉴얼에서는 이 제품의 설치 및 사용자 정의에 대해 설명하지 않습니다.
- Developer for System z 클라이언트는 Eclipse용 DTSP 프로파일 보기 플러그인을 사용하지 않습니다.
- Developer for System z 클라이언트는 일반 일괄처리 모드 디버깅에 언어 환경 사용자 종료를 사용하지 않습니다.
- Developer for System z 클라이언트는 직접적으로 문제점 판별 도구 공통 컴포넌트 서버와 통신하며, 이는 사용자가 이 포트 번호를 알고 있어야 하고 이 서버에서 사용되는 포트는 z/OS 호스트 시스템을 보호하는 방화벽에서 열려 있어야 함을 의미합니다.

(선택사항) 파일 관리자 지원

이 사용자 정의 태스크에는 Developer for System z 구성을 위한 도움, 특수 자원 또는 특수 사용자 정의 태스크가 필요하지 않습니다. 그러나 수행해야 하는 z/OS용 IBM File Manager 구성이 있습니다.

z/OS용 IBM 파일 관리자와의 Developer for System z 초기 통합은 Developer for System z 버전 8.0.3에서는 더 이상 사용되지 않는 것으로 표시되고, 버전 8.5에서는 더 이상 지원되지 않습니다. 이 기능에서 제공하는 서비스는 여러 영역으로 이동되었습니다. 형식화되지 않은 QSAM 편집 등과 같은 일부 기능은 현재 Developer for System z에서 처리하는 일반 데이터 세트의 일부입니다. 카피북 또는 포함 파일을 사용한 형식화된 데이터 편집 등의 고급 기능에서는 Eclipse용 IBM 파일 관리자 플러그인이 Developer for System z 클라이언트에 설치되어 있어야 합니다. 이 플러그인은 IBM 문제점 판별 도구 플러그인 웹 페이지(<http://www-01.ibm.com/software/awdtools/deployment/pdtpplugins/>)에서 사용 가능합니다.

Eclipse용 IBM 파일 관리자 플러그인에서는 문제점 판별 도구 서버를 사용하여 파일 관리자 서비스에 액세스합니다. 이 서버는 파일 관리자 ISPF 패널 인터페이스에서 사용되지 않습니다. 그러므로 문제점 판별 도구만의 추가 파일 관리자 설정 태스크가 있습니다. 자세한 내용은 파일 관리자 문서를 참조하십시오.

문제점 판별 도구 서버에서 사용되는 포트 번호는 `rsed.envvars` 지시문 `PD_SERVER_PORT`에서 지정되어야 합니다.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 rsed.envvars는 /etc/rdz에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오. TSO **OEDIT** 명령으로 파일을 편집할 수 있습니다.

참고:

- z/OS용 IBM 파일 관리자 제품은 별도로 확보, 설치 및 구성해야 합니다. 이 매뉴얼에서는 이 제품의 설치 및 사용자 정의에 대해 설명하지 않습니다.
- Developer for System z 클라이언트는 문제점 판별 도구 서버와 직접 통신하며, 이는 이 서버에서 사용되는 포트가 z/OS 호스트 시스템을 보호하는 방화벽에서 열려야 함을 의미합니다.

(선택사항) WORKAREA 및 /tmp 정리

이 사용자 정의 태스크는 도움, 특수 자원 또는 특수 사용자 정의 태스크가 필요하지 않습니다.

ISPF의 TSO/ISPF Client Gateway 및 SCLM 개발자 툴킷 기능은 WORKAREA 및 /tmp 디렉토리를 사용하여 세션이 완료되기 전에 제거되는 임시 작업 파일을 저장합니다. 그러나 임시 출력은 가끔(예를 들어, 처리 중에 통신 오류가 있는 경우) 뒤에 남아 있기도 합니다. 그러므로 때때로 WORKAREA 및 /tmp 디렉토리를 지우십시오.

z/OS UNIX에서는 현재 있는 디렉토리 및 이들의 기간에 따라 파일을 삭제하는 셸 스크립트 skulker를 제공합니다. 지정된 날짜 및 시간에 명령을 실행하는 z/OS UNIX cron 디먼과 함께 결합하여 주기적으로 대상 디렉토리를 정리하는 자동화된 도구를 설정할 수 있습니다. skulker 스크립트 및 cron 디먼에 대한 자세한 정보는 *UNIX System Services Command Reference*(SA22-7802)를 참조하십시오.

참고: FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 WORKAREA는 /var/rdz에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오.

제 8 장 설치 검증

제품 사용자 정의를 완료한 후 이 장에서 설명하는 설치 검증 프로그램(IVP)을 사용하여 키 제품 컴포넌트의 설정을 확인할 수 있습니다.

시작 태스크 확인

JMON, JES 작업 모니터

JMON 시작 태스크 또는 사용자 작업을 시작하십시오. DD SYSOUT의 시작 정보는 다음 메시지로 끝나야 합니다.

FEJ211I 서버가
연결을 허용할 준비가 되었습니다.

작업이 리턴 코드 66으로 종료되면, FEK.SFEKAUTH에는 APF 권한이 부여되지 않습니다.

참고: 다른 IVP 테스트를 계속하기 전에 JES 작업 모니터를 시작하십시오.

RSED, RSE 디먼

IVP=IVP 매개변수를 사용하여 RSED 시작 태스크 또는 사용자 작업을 시작하십시오. 이 매개변수를 사용하면 서버가 몇몇 설치 검증 테스트를 수행한 후 종료됩니다. DD STDOUT에서 이러한 테스트의 출력을 볼 수 있습니다. 특정 오류가 발생할 경우에는 DD STDERR에서 데이터를 사용할 수 있습니다. DD STDOUT에서 다음 IVP가 완료되었음을 나타내는 메시지를 확인하십시오.

- Java 시작
- JES 작업 모니터 연결
- TCP/IP 설정

STDOUT 데이터는 다음 샘플과 유사해야 합니다.

```
-----  
RSE daemon startup script  
-----
```

```
arguments: IVP -C/etc/rdz -P
```

```
RSE daemon IVP test
```

```
CDFMVS08 -- Fri Mar 23 17:50:52 2012 UTC  
uid=8(STCRSE) gid=1(STCGRP)
```

```
started from /usr/lpp/rdz/bin/rsed.sh
```

```

startup script version Aug09,2012

configuration files located in /etc/rdz -- startup argument
daemon port is 4035 -- set in rsed.envvars
debug level is 1 -- set in rsecomm.properties
TMPDIR=/tmp -- default

-----
current environment variables
-----
@="/usr/lpp/rdz/bin/rsed.sh" @[1]="-C/etc/rdz" @[2]="-P"
ANT_HOME="/usr/lpp/Apache/Ant/apache-ant-1.7.1"
CGI_DWORK="/var/rdz"
CGI_ISPCONF="/etc/rdz"
CGI_ISPHOME="/usr/lpp/ispf"
CGI_ISPWORK="/var/rdz"
CGI_TRANTABLE="FEK.#CUST.LSTRANS.FILE"
CLASSPATH=".:usr/lpp/rdz/lib:usr/lpp/rdz/lib/dstore_core.jar:usr/lpp/
ERRNO="0"
HOME="/tmp"
IFS="
"
JAVA_HOME="/usr/lpp/java/J6.0"
JAVA_PROPAGATE="NO"
LANG="C"
LIBPATH=".:usr/lib:usr/lpp/java/J6.0/bin:usr/lpp/java/J6.0/bin/classi
LINENO="66"
LOGNAME="STCRSE"
MAILCHECK="600"
OLDPWD="/tmp"
OPTIND="1"
PATH=".:usr/lpp/java/J6.0/bin:usr/lpp/rdz/bin:usr/lpp/ispf/bin:bin:/
PPID="33554711"
PS1="\$ "
PS2="> "
PS3="#? "
PS4="+ "
PWD="/etc/rdz"
RANDOM="27298"
RSE_CFG="/etc/rdz"
RSE_HOME="/usr/lpp/rdz"
RSE_LIB="/usr/lpp/rdz/lib"
SECONDS="0"
SHELL="/bin/sh"
STEPLIB="NONE"
TMPDIR="/tmp"
TZ="EST5EDT"
X_ARG="-T"
X_C="-- startup argument"
X_KEY="-T"
X_L="-- set in rsecomm.properties"
X_LOG="1"
X_P="-- set in rsed.envvars"
X_PORT="4035"
X_VAL=""
_="-----"
_BPX_SHAREAS="YES"
_BPX_SPAWN_SCRIPT="YES"

```

```

_CEE_DMPTARG="/tmp"
_CEE_RUNOPTS="ALL31(ON) HEAP(32M,32K,ANYWHERE,KEEP,,) TRAP(ON)"
_CMDSERV_BASE_HOME="/usr/lpp/ispf"
_CMDSERV_CONF_HOME="/etc/rdz"
_CMDSERV_WORK_HOME="/var/rdz"
_EDC_ADD_ERRNO2="1"
_RSE_ISPF_OPTS="&SESSION=SPAWN"
_RSE_DAEMON_CLASS="com.ibm.etools.zos.server.RseDaemon"
_RSE_DAEMON_IVP_TEST="1"
_RSE_HOST_CODEPAGE="IBM-1047"
_RSE_JAVAOPTS=" -DISPF_OPTS='&SESSION=SPAWN' -DA_PLUGIN_PATH=
_RSE_JMON_PORT="6715"
_RSE_LOG_LEVEL="1"
_RSE_POOL_SERVER_CLASS="com.ibm.etools.zos.server.ThreadPoolProcess"
_RSE_RSED_PORT="4035"
_RSE_SAF_CLASS="/usr/include/java_classes/IRRRacf.jar"
_RSE_SCRIPT_VERSION="Jan09,2012"
_RSE_SERVER_CLASS="org.eclipse.dstore.core.server.Server"
_RSE_SERVER_TIMEOUT="120000"
_SCLMDT_BASE_HOME="/usr/lpp/rdz"
_SCLMDT_CONF_HOME="/var/rdz/sclmdt"
_SCLMDT_TRANTABLE="FEK.#CUST.LSTRANS.FILE"
_SCLMDT_WORK_HOME="/var/rdz"
debug_level="1"

```

----- Address Space size limits

```

current address space size limit is 1913626624 (1825.0 MB)
maximum address space size limit is 2147483647 (2048.0 MB)

```

----- service history

```

Fri Jun 14 13:47:39 2013 -- COPY -- HHOP900 v9000 created 14 Jun 2013

```

----- java service level

```

java full version "J2RE 1.6.0 IBM z/OS build pmz3160sr13-20130207_01(SR13)

```

----- LE runtime options

```

Options Report for Enclave main 05/23/12 1:50:52 PM
Language Environment V01 R11.00

```

LAST WHERE SET	OPTION
Installation default	ABPERC(NONE)
Programmer default	ABTERMENC(RETCODE)
Installation default	NOAIXBLD
Invocation command	ALL31(ON)
Programmer default	ANYHEAP(32768,16384,ANYWHERE,FREE)
Installation default	NOAUTOTASK
Programmer default	BELOWHEAP(32768,16384,FREE)
Installation default	CBLOPTS(ON)
Installation default	CBLPShPOP(ON)


```

|      Installation default      CBLQDA(OFF)
|      Installation default
|      CEEDUMP(60,SYSOUT=*,FREE=END,SPIN=UNALL
|      Installation default      CHECK(ON)
|      Installation default      COUNTRY(US)
|      Installation default      NODEBUG
|      Installation default      DEPTHCONDLMT(10)
|      Installation default      DYNDUMP(*USERID,NODYNAMIC,TDUMP)
|      Installation default      ENVAR("")
|      Installation default      ERRCOUNT(0)
|      Installation default      ERRUNIT(6)
|      Installation default      FILEHIST
|      Installation default      FILETAG(NOAUTOCVT,NOAUTOTAG)
|      Default setting          NOFLOW
|      Invocation command        HEAP(33554432,32768,ANYWHERE,KEEP,16384
|      Installation default      HEAPCHK(OFF,1,0,0,0)
|      Installation default      HEAPPOLS(OFF,8,10,32,10,128,10,256,10,
|      Installation default      INFOMSGFILTER(OFF,,,))
|      Installation default      INQPCOPN
|      Installation default      INTERRUPT(OFF)
|      Programmer default        LIBSTACK(32768,16384,FREE)
|      Installation default      MSGFILE(SYSOUT,FBA,121,0,NOENQ)
|      Installation default      MSGQ(15)
|      Installation default      NATLANG(ENU)
|      Ignored                   NONONIPSTACK(See THREADSTACK)
|      Installation default      OCSTATUS
|      Installation default      NOPC
|      Installation default      PLITASKCOUNT(20)
|      Programmer default        POSIX(ON)
|      Installation default      PROFILE(OFF,"")
|      Installation default      PRTUNIT(6)
|      Installation default      PUNUNIT(7)
|      Installation default      RDRUNIT(5)
|      Installation default      RECPAD(OFF)
|      Invocation command        RPTOPTS(ON)
|      Installation default      RPTSTG(OFF)
|      Installation default      NORTEREUS
|      Installation default      NOSIMVRD
|      Programmer default
|      STACK(65536,65536,ANYWHERE,KEEP,524288,131072)
|      Installation default      STORAGE(NONE,NONE,NONE,0)
|      Installation default      TERMTHDACT(TRACE,,96)
|      Installation default      NOTEST(ALL,"*","PROMPT","INSPREF")
|      Installation default      THREADHEAP(4096,4096,ANYWHERE,KEEP)
|      Installation default
|      THREADSTACK(OFF,4096,4096,ANYWHERE,KEEP,131072,
|      Installation default      TRACE(OFF,4096,DUMP,LE=0)
|      Invocation command        TRAP(ON,SPIE)
|      Installation default      UPSI(00000000)
|      Installation default      NOUSRHDLR(,)
|      Installation default      VCTRSVE(OFF)
|      Installation default      XPLINK(OFF)
|      Installation default      XUFLOW(AUTO)
|
|      -----
|      java startup test...
|      -----
|      java full version "JRE 1.6.0 IBM z/OS build pmz3160sr13-20130207_01

```

```

(SR13)"
java version "1.6.0"
Java(TM) SE Runtime Environment (build pmz3160sr13-20130207_01(SR13))
IBM J9 VM (build 2.4, JRE 1.6.0 IBM J9 2.4 z/OS s390-31 jvmmz3160sr13-
20130114_1
J9VM - 20130114_134867
JIT - r9_20130108_31100
GC - 20121212_AA)
JCL - 20130204_01

```

```

-----
JES Job Monitor test...
-----

```

```

executed on CDFMVS08 -- Fri Mar 23 17:50:52 EDT 2012
executed by uid=8(STCRSE) gid=1(STCGROUP)
using /etc/rdz/rsed.envvars

```

```

current address space size limit is 1913626624 (1825.0 MB)
maximum address space size limit is 2147483647 (2048.0 MB)

```

```

testing JES Job Monitor on port 6715...
hostName=CDFMVS08
hostAddr=9.42.112.75
IPv4 is supported
Waiting for JES Job Monitor response...
ACKNOWLEDGE01v03
Success

```

```

-----
TCP/IP IVP test...
-----

```

```

executed on CDFMVS08 -- Fri Mar 23 17:50:53 EDT 2012
executed by uid=8(STCRSE) gid=1(STCGROUP)
using /etc/rdz/rsed.envvars

```

```

current address space size limit is 1913626624 (1825.0 MB)
maximum address space size limit is 2147483647 (2048.0 MB)

```

```

-----
TCP/IP resolver configuration (z/OS UNIX search order):
-----

```

```

Resolver Trace Initialization Complete -> 2012/05/23 17:50:54.208378

```

```

res_init Resolver values:
Global Tcp/Ip Dataset = None
Default Tcp/Ip Dataset = None
Local Tcp/Ip Dataset = /etc/resolv.conf
Translation Table = Default
UserId/JobName = STCRSE
Caller API = LE C Sockets
Caller Mode = EBCDIC
(L) DataSetPrefix = TCPIP
(L) HostName = CDFMVS08
(L) TcpIpJobName = TCPIP
(L) DomainOrigin = RALEIGH.IBM.COM
(L) NameServer = 9.42.206.2
               9.42.206.3

```

```

(L) NsPortAddr      = 53                (L) ResolverTimeout    = 10
(L) ResolveVia      = UDP                (L) ResolverUdpRetries = 1
(*) Options NDots   = 1
(*) SockNoTestStor
(*) AlwaysWto       = NO                (L) MessageCase        = MIXED
(*) LookUp          = DNS LOCAL
res_init Succeeded
res_init Started: 2012/05/23 17:50:54.229888
res_init Ended: 2012/05/23 17:50:54.229898
*****
MVS TCP/IP NETSTAT CS V1R11          TCPIP Name: TCPIP          17:50:54
Tcplib started at 11:31:40 on 05/23/2012 with IPv6 enabled

-----
host IP address:
-----
hostName=CDFMVS08
hostAddr=9.42.112.75
bindAddr=9.42.112.75
localAddr=9.42.112.75

Success, addresses match
-----
RSE daemon IVP ended -- return code 0 -- Fri Mar 23 17:50:55 EDT 2012
-----

```

참고: 다른 IVP 테스트를 시작하기 전에 IVP 매개변수 없이 RSE 디몬을 시작하십시오. 시작에 성공하면 RSE 디몬이 다음과 같은 콘솔 메시지를 발행합니다.

FEK002I RseDaemon이 시작되었음(포트=4035)

DBGMGR, 디버그 관리자

선택적 DBGMGR 시작 태스크 또는 사용자 작업을 시작하십시오. 정상적으로 시작되면 서버가 다음 콘솔 메시지를 발생립니다. 여기서 clientport는 외부(클라이언트-호스트) 통신에 사용되는 포트 번호이며 hostport는 내부(호스트 한정) 통신에 사용되는 포트 번호입니다.

AQECM001I Debug Manager startup complete (clientport/hostport)

작업이 리턴 코드 66으로 종료되면, FEK.SFEKAUTH에는 APF 권한이 부여되지 않습니다.

참고: 다른 디버그 관련 IVP 테스트를 계속하기 전에 DBGMGR을 시작하십시오.

IVP 운영자 명령

활동 RSE 디몬은 **IVP** modify 명령을 지원하며, 콘솔에서 선택한 IVP를 수행하는 데 사용할 수 있습니다.

PassTicket 재사용가능성

매 초당 한 사용자에게 대해 하나의 PassTicket만 생성하도록 제한하므로 Developer for System z에서는 생성하는 PassTicket을 재사용할 수 있어야 합니다. 다음 운영자 명령을 실행하여 PassTicket 재사용가능성을 확인하십시오. userid를 올바른 TSO 사용자 ID로 대체하십시오.

```
MODIFY RSED,APPL=IVP PASSTICKET,userid
```

명령은 다음 샘플과 유사한 출력을 리턴해야 합니다.

```
MODIFY RSED,APPL=IVP PASSTICKET,IBMUER
```

```
+FEK900I PASSTICKET IVP: start: serverid=STCRSE userid=IBMUER
+FEK900I PASSTICKET IVP: the default applid=FEKAPPL
+FEK900I PASSTICKET IVP: Success, PassTicket IVP finished normally
+FEK901I PASSTICKET IVP Exit code = 0
```

RSE 디먼 연결

다음 명령을 실행하여 RSE 디먼 연결을 확인하십시오. userid를 올바른 TSO 사용자 ID로 대체하십시오.

```
MODIFY RSED,APPL=IVP DAEMON,userid
```

이 명령은 161 페이지의 『서비스 확인』에서 설명한 fekfivpd IVP와 기능상 동일하지만 비밀번호가 필요하지 않은 장점이 있습니다. RSE는 PassTicket을 생성하고 이것을 비밀번호로 사용합니다. 명령은 다음 샘플과 유사한 출력을 리턴해야 합니다.

```
F RSED,APPL=IVP DAEMON,IBMUER
```

```
+FEK900I DAEMON IVP: SSL is disabled
+FEK900I DAEMON IVP: connected
+FEK900I DAEMON IVP: 1343
+FEK900I DAEMON IVP: 8878350
+FEK900I DAEMON IVP: Success
+FEK901I DAEMON IVP Exit code = 0
```

ISPF Client Gateway

다음 명령을 실행하여 ISPF Client Gateway 연결을 확인하십시오. userid를 올바른 TSO 사용자 ID로 대체하십시오.

```
MODIFY RSED,APPL=IVP ISPF,userid
```

이 명령은 기능상으로 161 페이지의 『서비스 확인』에서 기술된 fekfivpi IVP와 동일합니다. 명령은 다음 샘플과 유사한 출력을 리턴해야 합니다.

```
F RSED,APPL=IVP ISPF,IBMUER
```

```
+FEK900I ISPF IVP: executed on CDFMVS08 -- Tue Sep 13 22:29:28 EDT 2011
+FEK900I ISPF IVP: executed by uid=1(IBMUSER) gid=0(SYS1)
+FEK900I ISPF IVP: using /etc/rdz/rsed.envvars
+FEK900I ISPF IVP: current address space size limit is 2147483647
(2048.0 MB)
```

```

+FEK900I ISPF IVP: maximum address space size limit is 2147483647
(2048.0 MB)
+FEK900I ISPF IVP: -----
-----
+FEK900I ISPF IVP: /etc/rdz/ISPF.conf content:
+FEK900I ISPF IVP: -----
-----
+FEK900I ISPF IVP: ispllib=ISP.SISPLDLOAD
+FEK900I ISPF IVP: ispmllib=ISP.SISPMENU
+FEK900I ISPF IVP: isptlib=ISP.SISPTENU
+FEK900I ISPF IVP: ispllib=ISP.SISPPENU
+FEK900I ISPF IVP: ispslib=ISP.SISPSLIB
+FEK900I ISPF IVP: sysproc=ISP.SISPCLIB,FEK.SFEKPROC
+FEK900I ISPF IVP: -----
-----
+FEK900I ISPF IVP: Host install verification for RSE
+FEK900I ISPF IVP: Review IVP log messages from HOST below :
+FEK900I ISPF IVP: -----
-----
+FEK900I ISPF IVP: Service level 22Feb2011
+FEK900I ISPF IVP: RSE connection and base TSO/ISPF session initializati
on check only
+FEK900I ISPF IVP: *** CHECK : ENVIRONMENT VARIABLES - key variables
displayed below :
+FEK900I ISPF IVP: Server PATH = ./usr/lpp/java/J6.0/bin:/usr/l
pp/rdz/bin:/usr/lpp/ispf/bin:/bin:/usr/sbin
+FEK900I ISPF IVP: STEPLIB = NONE
+FEK900I ISPF IVP: Temporary directory = /tmp
+FEK900I ISPF IVP: CGI_ISPHOME = /usr/lpp/ispf
+FEK900I ISPF IVP: CGI_ISPCONF = /etc/rdz
+FEK900I ISPF IVP: CGI_ISPWORK = /var/rdz
+FEK900I ISPF IVP: -----
-----
+FEK900I ISPF IVP: *** CHECK : USS MODULES
+FEK900I ISPF IVP: Checking ISPF Directory : /usr/lpp/ispf
+FEK900I ISPF IVP: Checking modules in /usr/lpp/ispf/bin directory
+FEK900I ISPF IVP: Checking for ISPF configuration file ISPF.conf
+FEK900I ISPF IVP: RC=0
+FEK900I ISPF IVP: MSG: SUCCESSFUL
+FEK900I ISPF IVP: -----
-----
+FEK900I ISPF IVP: *** CHECK : TSO/ISPF INITIALIZATION
+FEK900I ISPF IVP: ( TSO/ISPF session will be initialized )
+FEK900I ISPF IVP: RC=0
+FEK900I ISPF IVP: MSG: SUCCESSFUL
+FEK900I ISPF IVP: -----
-----
+FEK900I ISPF IVP: *** CHECK: Shutting down TSO/ISPF IVP session
+FEK900I ISPF IVP: RC=0
+FEK900I ISPF IVP: MSG: SUCCESSFUL
+FEK900I ISPF IVP: -----
-----
+FEK900I ISPF IVP: Host installation verification completed successfully
+FEK900I ISPF IVP: -----
-----
+FEK901I ISPF IVP Exit code = 0

```

서비스 확인

Developer for System z 설치의 기본 및 선택적 서비스에 대해 여러 IVP(Installation Verification Programs)를 제공합니다. IVP 스크립트는 설치 디렉토리(기본값은 /usr/lpp/rdz/bin/)에 있습니다.

표 19. 서비스를 위한 IVP

fekfivpc	167 페이지의 『(선택사항) CARMA 연결』
fekfivpd	164 페이지의 『RSE 디먼 연결』
fekfivpi	165 페이지의 『ISPF의 TSO/ISPF Client Gateway 연결』
fekfivpj	165 페이지의 『JES 작업 모니터 연결』
fekfivps	168 페이지의 『(선택사항) SCLMDT 연결』
fekfivpt	163 페이지의 『TCP/IP 설정』

다음 절에서 설명하는 태스크에서는 사용자가 z/OS UNIX에서 활성 상태여야 합니다. OMVS TSO 명령을 실행하면 수행할 수 있습니다. TSO로 리턴하려면 **exit** 명령을 사용하십시오.

많은 양의 메모리가 필요한 Java와 같은 기능이 실행되므로 IVP를 실행하는 사용자 ID에게는 대용량 리전이 필요합니다. 리전 크기를 131072 킬로바이트(128 메가바이트) 이상으로 설정해야 합니다.

다음 샘플 오류는 리전 크기 부족에 대한 분명한 표시이지만, 기타 오류도 발생할 수 있습니다. 예를 들어, Java 시작에 실패할 수 있습니다.

```
CEE5213S The signal SIGPIPE was received.
%z/OS UNIX command%: command was killed by signal number 13
%line-number% ** %REXX command%
+++ RC(137) +++
```

참고: IVP 테스트를 시작하기 전에 Developer for System z 시작 태스크가 활성 상태여야 합니다.

IVP 초기화

이 절의 모든 샘플 명령에서는 특정 환경 변수를 설정해야 합니다. 이렇게 하면 PATH 명령문을 통해 IVP 스크립트를 사용할 수 있고 사용자 정의된 구성 파일의 위치가 알려집니다. **pwd** 및 **cd** 명령을 사용하여 현재 디렉토리를 확인하고 사용자 정의된 구성 파일이 있는 디렉토리로 변경하십시오. 그런 다음 **ivpinit** 쉘 스크립트를 사용하여 다음 샘플에서 처럼 RSE 환경 변수를 설정할 수 있습니다. 여기서 \$는 z/OS UNIX 프롬프트입니다.

```
$ pwd
/u/userid
$ cd /etc/rdz
$ ./ivpinit
RSE configuration files located in /etc/rdz --default
added /usr/lpp/rdz/bin to PATH
```

. ./ivpinit에 있는 첫 번째 점(.)은 현재 환경에서 셸을 실행하는 z/OS UNIX 명령 이므로, 셸의 환경 변수 세트가 셸 종료 후에도 계속 적용됩니다. 두 번째 점(.)은 현재 디렉토리를 말합니다.

참고:

- . ./ivpinit가 fekfivp* 스크립트보다 먼저 실행되지 않는 경우, 이 스크립트에 대한 경로는 다음 샘플에서 처럼 호출 시 지정해야 합니다.

```
/usr/lpp/rdz/bin/fekfivpr 512 USERID
```

또한 . ./ivpinit가 먼저 실행되지 않으면, fekfivp*스크립트 모두는 사용자 정의된 rsed.envvars의 위치를 요청합니다.

- 일부 IVP 테스트에서는 TCP/IP REXX 소켓 API를 사용하며, TCP/IP 로드 라이브러리(기본값: TCPIP.SEZALOAD)가 LINKLIST 또는 STEPLIB에 있어야 합니다. 다음 명령은 이 IVP 테스트를 실행하는 데 필요할 수 있습니다.

```
$ EXPORT STEPLIB=$STEPLIB:TCPIP.SEZALOAD
```

기존 STEPLIB에 APF 무허가 라이브러리를 추가하면 기존 STEPLIB 데이터 세트의 APF 권한 부여가 제거됩니다.

CEE.SCEELKED가 LINKLIST 또는 STEPLIB에 있는 경우, TCPIP.SEZALOAD는 CEE.SCEELKED 앞에 있어야 합니다. 그렇지 않으면 TCP/IP REXX 소켓 호출에 대해 0C1 시스템 이상 종료가 발생합니다.

호스트 시스템 연결 문제 진단에 대한 정보는 호스트 구성 참조서 (SA30-4501)의 "구성 문제점 해결" 및 Developer for System z 웹 사이트의 지원 섹션(<http://www-03.ibm.com/software/products/us/en/developerforsystemz/>)에 있는 기술 문서를 참조하십시오.

포트 사용 가능성

JES 작업 모니터와 RSE 디먼 포트 사용 가능성은 **netstat** 명령을 실행하여 확인할 수 있습니다. 결과는 다음 샘플에서와 같이, 이러한 서비스에서 사용하는 포트를 표시합니다.

IPv4

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

IPv6


```

$ netstat
MVS TCP/IP NETSTAT CS VxRy      TCPIP Name: TCPIP      14:03:35
User Id  Conn      State
-----  ----      -
RSED     0000004B Listen
  Local Socket:  0.0.0.0..4035
  Foreign Socket: 0.0.0.0..0
JMON     00000037 Listen
  Local Socket:  0.0.0.0..6715
  Foreign Socket: 0.0.0.0..0

```

TCP/IP 설정

Developer for System z는 초기화될 때 올바른 호스트 이름을 갖는 TCP/IP에 종속됩니다. 이는 다른 TCP/IP 및 분석기 구성 파일이 올바르게 설정되어 있어야 함을 의미합니다. TCP/IP 및 분석기 설정에 대한 자세한 정보는 *호스트 구성 참조서 (SA30-4501)* 에서 "TCP/IP 설정"의 내용을 참조하십시오. 다음 명령을 실행하여 현재 설정을 확인하십시오.

```
fekfivpt
```

참고: 이 IVP는 TCPIP **netstat -u** 명령을 실행하며, 보안 소프트웨어에 의해 실행되지 않도록 보호될 수 있습니다. SERVAUTH 클래스의 EZB.NETSTAT.mvsname.tcprocname.UP 프로파일을 참조하십시오.

명령은 이 샘플과 유사한 출력을 리턴해야 합니다.

```

$ fekfivpt

executed on CDFMVS08 -- Wed Jul  2 13:11:54 EDT 2008
executed by uid=1(USERID) gid=0(GROUP)
using /etc/rdz/rsed.envvars

current address space size limit is 1914675200 (1826.0 MB)
maximum address space size limit is 2147483647 (2048.0 MB)

-----
TCP/IP resolver configuration (z/OS UNIX search order):
-----
Resolver Trace Initialization Complete -> 2008/07/02 13:11:54.745964

res_init Resolver values:
Global Tcp/Ip Dataset = None
Default Tcp/Ip Dataset = None
Local Tcp/Ip Dataset  = /etc/resolv.conf
Translation Table      = Default
UserId/JobName          = USERID
Caller API              = LE C Sockets
Caller Mode             = EBCDIC
(L) DataSetPrefix      = TCPIP
(L) HostName           = CDFMVS08
(L) TcpIpJobName       = TCPIP
(L) DomainOrigin       = RALEIGH.IBM.COM
(L) NameServer         = 9.42.206.2
                      9.42.206.3

```

```

(L) NsPortAddr      = 53                (L) ResolverTimeout    = 10
(L) ResolveVia      = UDP                (L) ResolverUdpRetries = 1
(*) Options NDots    = 1
(*) SockNoTestStor
(*) AlwaysWto        = NO                (L) MessageCase         = MIXED
(*) LookUp           = DNS LOCAL
res_init Succeeded
res_init Started: 2008/07/02 13:11:54.755363
res_init Ended: 2008/07/02 13:11:54.755371
*****
MVS TCP/IP NETSTAT CS V1R9          TCPIP Name: TCPIP          13:11:54
Tcpiip started at 01:28:36 on 06/23/2008 with IPv6 enabled

-----
host IP address:
-----
hostName=CDFMVS08
hostAddr=9.42.112.75
bindAddr=9.42.112.75
localAddr=9.42.112.75

Success, addresses match

```

RSE 디먼 연결

다음 명령을 실행하여 RSE 디먼 연결을 확인하십시오.

```
fekfivpd
```

비밀번호를 위해 프롬프트를 표시한 후 명령은 다음 샘플과 유사한 출력을 리턴해야 합니다.

```
$ fekfivpd
```

```

executed on CDFMVS08 -- Wed Jul  2 15:00:27 EDT 2008
executed by uid=1(USERID) gid=0(GROUP)
using /etc/rdz/rsed.envvars

```

```

current address space size limit is 1914675200 (1826.0 MB)
maximum address space size limit is 2147483647 (2048.0 MB)

```

```
attempting to connect userid USERID using port 4035 ...
```

```

Password:
SSL is disabled
connected
8108
570655399
Success

```

SSL 사용 연결을 테스트 중인 경우, IVP를 실행하는 사용자 ID가 Developer for System z 인증서 사인에 사용되는 CA 인증서를 포함하여 필요한 인증서 모두에 대한 액세스를 가지고 있어야 합니다. 이 IVP, F RSED,APPL=IVP DAEMON,userid의 운영

자 명령 버전에서는 RSE 호스트 시스템에 수행된 SSL 설정을 사용하므로 오류가 적게 발생하는 경향이 있습니다. 다음 목록에서는 몇 가지 공통 SSL 관련 오류를 제공합니다.

- 다음 오류 메시지가 표시되면 IVP를 실행하는 사용자 ID가 필요한 모든 인증서에 대한 액세스를 가지고 있는지 확인하십시오. gsk_environment_init() failed: Error detected while opening the certificate data base
- 다음 오류 메시지가 표시되면 키링에 대해서도 CA 인증서 사인이 되었는지 확인하십시오. gsk_secure_socket_init() failed: Certificate validation error

JES 작업 모니터 연결

다음 명령을 실행하여 JES 작업 모니터 연결을 확인하십시오.

```
fekfivpj
```

명령은 다음 샘플에 표시되는 것과 유사한 JES 작업 모니터 수신확인 메시지를 리턴해야 합니다(\$는 z/OS UNIX 프롬프트입니다).

```
$ fekfivpj
```

```
executed on CDFMVS08 -- Wed Jul  2 15:00:27 EDT 2008
executed by uid=1(USERID) gid=0(GROUP)
using /etc/rdz/rsed.envvars
```

```
current address space size limit is 1914675200 (1826.0 MB)
maximum address space size limit is 2147483647 (2048.0 MB)
```

```
testing JES Job Monitor on port 6715...
hostName=CDFMVS08
hostAddr=9.42.112.75
IPv4 is supported
Waiting for JES Job Monitor response...
ACKNOWLEDGE01v03
```

```
Success
```

ISPF의 TSO/ISPF Client Gateway 연결

다음 명령을 실행하여 ISPF의 TSO/ISPF Client Gateway 연결을 확인하십시오.

```
fekfivpi
```

명령은 변수, HFS 모듈, TSO/ISPF 세션의 시작 및 중지 및 같은 ISPF의 TSO/ISPF client Gateway 관련 검사 결과를 리턴해야 합니다. 출력은 다음 샘플의 내용과 유사해야 합니다.

```
$ fekfivpi
```

```
executed on CDFMVS08 -- Wed Jul  2 15:00:27 EDT 2008
executed by uid=1(USERID) gid=0(GROUP)
using /etc/rdz/rsed.envvars
```

```
current address space size limit is 1914675200 (1826.0 MB)
```

maximum address space size limit is 2147483647 (2048.0 MB)

/etc/rdz/ISPF.conf content:

```
ispmllib=ISP.SISPMENU
isptlib=ISP.SISPTENU
ispplib=ISP.SISPPENU
ispslib=ISP.SISPSLIB
ispllib=ISP.SISPLOAD
sysproc=ISP.SISPCLIB,FEK.SFEKPROC
```

Host install verification for RSE
Review IVP log messages from HOST below :

RSE connection and base TSO/ISPF session initialization check only

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

```
Server PATH          =
/usr/lpp/java/J6.0/bin:/usr/lpp/rdz/lib:/usr/lpp/ispf/bin:
/bin:/usr/sbin:.
```

```
STEPLIB              = FEK.SFEKAUTH:FEK.SFEKLOAD
```

```
CGI_ISPHOME   = /usr/lpp/ispf
CGI_ISPCONF   = /etc/rdz
CGI_ISPWORK   = /var/rdz
```

*** CHECK : USS MODULES
Checking ISPF Directory : /usr/lpp/ispf
Checking modules in /usr/lpp/ispf/bin directory
Checking for ISPF configuration file ISPF.conf
RC=0
MSG: SUCCESSFUL

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

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

Host installation verification completed successfully

참고: ISPF 검사가 실패하면 좀 더 자세한 정보가 표시됩니다.

fekfivpi에는 다음과 같은 선택적이고 위치가 지정되지 않은 매개변수가 있습니다.

-file fekfivpi는 수 백 라인에 이르는 대용량 출력을 생성할 수 있습니다. -file 매개변수는 이 출력을 \$TMPDIR/fekfivpi.log 파일로 전송합니다. 여기서 \$TMPDIR은 rsed.envvars에 있는 TMPDIR 지시문의 값입니다. 기본값은 /tmp 입니다.

-debug

-debug 매개변수는 상세한 테스트 출력을 작성합니다. IBM 지원 센터에서 지시한 경우가 아니면 이 옵션을 사용하지 마십시오.

(선택사항) CARMA 연결

다음 명령을 실행하여 CARMA에 대한 연결을 확인하십시오.

```
fekfivpc
```

명령은 다음 샘플에 표시된 대로 CARMA 관련 검사 결과를 리턴해야 합니다.

```
$ fekfivpc
```

```
executed on CDFMVS08 -- Fri Aug 20 14:15:46 EDT 2010
executed by uid=1(USERID) gid=0(GROUP)
using /etc/rdz/rsed.envvars
```

```
current address space size limit is 140484608 ( 134.0 MB)
maximum address space size limit is 2147483647 (2048.0 MB)
```

*** /etc/rdz/CRASRV.properties 콘텐츠:

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

*** /tmp/fekfivpc.log 작성

*** CARMA 설치 확인...

1. Creating CARMA connection (timeout after 60 seconds)
2. Initializing CARMA
3. Retrieving RAM list
The following RAMs were found
00 CA Endevor SCM Unique ID: COM.IBM.CARMA.ENDEVORRAM
4. Getting customization data for RAM 00
5. Initializing RAM 00
6. Retrieving Repository Instance List
Found 6 Repository Instance(s)
7. Terminating RAM 00
8. Terminating CARMA

*** IVP 완료!!!!

참고: IVP가 실패하면 /tmp/fekfivpc.log의 콘텐츠를 확인하십시오. 이 로그는 RSE와 CARMA 사이의 통신에 대해 문서화하고 장애의 근본 원인을 찾는 데 도움이 되는 정보가 들어 있을 수 있습니다.

fekfivpc에는 다음의 선택적이고 위치가 지정되지 않은 매개변수가 있습니다.

-noram

기본적으로 fekfivpc는 CRADEF VSAM 데이터 세트에 정의되어 있는 첫 번째 RAM을 시작합니다. RAM을 테스트하지 않으려는 경우 인스턴스가 있을 수 있습니다. 예를 들어, 씨드파티 RAM은 먼저 표시되고 예상치 못한 입력이 필요합니다. 이런 경우 -noram 시작 인수를 사용하여 IVP 테스트의 RAM 특정 단계(단계 4 - 7)를 생략할 수 있습니다.

(선택사항) SCLMDT 연결

다음 명령을 실행하여 SCLM 개발자 툴킷에 대한 연결을 확인하십시오.

```
fekfivps
```

명령은 변수, HDS 모듈, REXX 런타임, TSO/ISPF 세션의 시작 및 중지 및 같은 SCLM 개발자 툴킷 관련 검사의 결과를 리턴하고, 다음 샘플과 유사한 출력을 표시해야 합니다.

```
$ fekfivps
```

```
executed on CDFMVS08 -- Wed Jul  2 15:00:27 EDT 2008
executed by uid=1(USERID) gid=0(GROUP)
using /etc/rdz/rsed.envvars
```

```
current address space size limit is 1914675200 (1826.0 MB)
maximum address space size limit is 2147483647 (2048.0 MB)
```

```
-----
/etc/rdz/ISPF.conf content:
-----
```

```
ispmllib=ISP.SISPMENU
isptlib=ISP.SISPTENU
isplib=ISP.SISPPENU
ispslib=ISP.SISPSLIB
ispllib=ISP.SISpload
sysproc=ISP.SISPCLIB,FEK.SFEKPROC
```

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

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

```
Server PATH          = /usr/lpp/java/J6.0/bin:/usr/lpp/rdz/lib:/usr/lpp/ispf/bin:
/bin:/usr/sbin:.
```

```
STEPLIB              = FEK.SFEKAUTH:FEK.SFEKLOAD
```

```

CGI_ISPHOME = /usr/lpp/ispf
CGI_ISPCONF = /etc/rdz
CGI_ISPWORK = /var/rdz
_SCLMDT_CONF_HOME = /var/rdz/sclmdt
_SCLMDT_WORK_HOME = /var/rdz
_SCLMDT_TRANTABLE = FEK.#CUST.LSTRANS.FILE

```

```

-----
*** CHECK : JAVA PATH SETUP VERIFICATION
RC=0
MSG: SUCCESSFUL

```

```

-----
*** CHECK : USS MODULES
Checking ISPF Directory : /usr/lpp/ispf
Checking modules in /usr/lpp/ispf/bin directory
Checking for ISPF configuration file ISPF.conf
Checking install bin Directory : /usr/lpp/rdz/bin
RC=0
MSG: SUCCESSFUL

```

```

-----
*** CHECK : REXX RUNTIME ENVIRONMENT
RC=0
MSG: SUCCESSFUL

```

```

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

```

```

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

```

```

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

```

참고: SCLMDT 검사가 실패하면 자세한 정보가 표시됩니다.

fekfivps에는 다음과 같은 선택적이고 위치가 지정되지 않은 매개변수가 있습니다.

-file fekfivps는 수 백 라인에 이르는 대용량 출력을 생성할 수 있습니다. -file 매개변수는 이 출력을 \$TMPDIR/fekfivps.log 파일로 전송하며, 여기서 \$TMPDIR은 rsed.envvars에 있는 TMPDIR 지시문 값이고, 기본값은 /tmp입니다.

-debug

-debug 매개변수는 상세한 테스트 출력을 작성합니다. IBM 지원 센터에서 지시한 경우가 아니면 이 옵션을 사용하지 마십시오.

(선택사항) 통합 디버거 연결

다음 명령을 실행하여 통합 디버거에 대한 연결을 확인하십시오.

```
fekfivpe
```

명령은 다음 샘플과 유사한 출력에 표시된 대로 통합 디버거 관련 검사 결과를 리턴해야 합니다.

제 9 장 보안 정의

Developer for System z의 기본 보안 정의를 작성하는 샘플 RACF 및 z/OS UNIX 명령이 있는 샘플 FEKRACF 멤버를 사용자 정의하고 제출하십시오.

FEK.SFEKSAMP(FEKSETUP) 작업을 사용자 정의하고 제출할 때 다른 위치를 지정한 경우가 아니면 FEKRACF는 FEK.#CUST.JCL에 있습니다. 자세한 정보는 18 페이지의 『사용자 정의 설치』의 내용을 참조하십시오.

RACF 명령에 대한 자세한 정보는 *RACF Command Language Reference* (SA22-7687)를 참조하십시오.

참고:

- z/OS용 CA ACF2™을 사용하는 사이트의 경우 CA 지원 사이트(<https://support.ca.com>)의 제품 페이지를 참조하거나 관련 Developer for System z 지식 문서, TEC492389를 확인하십시오. 이 지식 문서에는 Developer for System z를 적절히 구성하는 데 필요한 보안 명령에 대한 세부사항이 들어 있습니다.
- z/OS용 CA Top Secret®을 사용하는 사이트의 경우 CA 지원 사이트(<https://support.ca.com>)의 제품 페이지를 참조하거나 관련 Developer for System z 지식 문서, TEC492091을 확인하십시오. 이 지식 문서에는 Developer for System z를 적절히 구성하는 데 필요한 보안 명령에 대한 세부사항이 들어 있습니다.

다음 절에서는 필수 단계, 선택적 구성 및 가능한 대안에 대해 설명합니다.

요구사항 및 체크리스트

보안 설정을 완료하려면 보안 관리자가 표 20에 나열된 값을 알아야 합니다. 이러한 값은 이전 Developer for System z 설치 및 사용자 정의 단계 중에 정의되었습니다.

표 20. 보안 설정 변수

설명	<ul style="list-style-type: none">기본값답을 찾을 수 있는 위치	값
Developer for System z 제품 상위 레벨 규정자	<ul style="list-style-type: none">FEKSMP/E 설치	
Developer for System z 사용자 정의 상위 레벨 규정자	<ul style="list-style-type: none">FEK.#CUSTFEK.SFEKSAMP(FEKSETUP)(18 페이지의 『사용자 정의 설치』에 설명됨)	

표 20. 보안 설정 변수 (계속)

설명	<ul style="list-style-type: none"> 기본값 답을 찾을 수 있는 위치 	값
통합 디버거 시작 태스크 이름	<ul style="list-style-type: none"> DBGMGR FEK.#CUST.PROCLIB(DBGMGR)(27 페이지의 『PROCLIB 변경사항』에 설명됨) 	
JES 작업 모니터 시작된 태스크 이름	<ul style="list-style-type: none"> JMON FEK.#CUST.PROCLIB(JMON)(27 페이지의 『PROCLIB 변경사항』에 설명됨) 	
RSE 디먼 시작된 태스크 이름	<ul style="list-style-type: none"> RSED FEK.#CUST.PROCLIB(RSED)(27 페이지의 『PROCLIB 변경사항』에 설명됨) 	
애플리케이션 ID	<ul style="list-style-type: none"> FEKAPPL /etc/rdz/rsed.envvars(54 페이지의 『_RSE_JAVAOPTS를 사용하여 여분의 Java 시작 매개변수 정의』에 설명됨) 	

다음 목록은 Developer for System z의 기본 보안 설정을 완료하는 데 필요한 조치 개요입니다. 다음 절에 설명된 대로 필수 보안 레벨에 따라 여러 메소드를 사용하여 이러한 요구사항을 충족시킬 수 있습니다. 선택적 Developer for System z 서비스 보안 설정에 대한 정보는 이전 절을 참조하십시오.

- 173 페이지의 『보안 설정 및 클래스 활성화』
- 174 페이지의 『Developer for System z 사용자에게 대한 OMVS 세그먼트 정의』
- 174 페이지의 『Developer for System z 시작 태스크 정의』
- 176 페이지의 『RSE를 보안 z/OS UNIX 서버로 정의』
- 176 페이지의 『RSE에 대한 MVS 프로그램 제어 라이브러리 정의』
- 177 페이지의 『RSE에 대한 PassTicket 지원 정의』
- 179 페이지의 『RSE에 대한 애플리케이션 보호 정의』
- 179 페이지의 『JES 명령 보안 정의』
- 181 페이지의 『데이터 세트 프로파일 정의』
- 187 페이지의 『RSE에 대한 z/OS UNIX 프로그램 제어 파일 정의』
- 187 페이지의 『보안 설정 확인』

보안 설정 및 클래스 활성화

Developer for System z는 다양한 보안 메커니즘을 사용하여 안전하고 제어된 클라이언트의 호스트 시스템 환경을 확보합니다. 이를 위해 여러 클래스 및 보안 설정을 다음 샘플 RACF 명령에 표시된 대로 활성화해야 합니다.

- 현재 설정 표시
 - SETROPTS LIST
- z/OS UNIX 및 디지털 인증서 프로파일에 대해 facility 클래스 활성화
 - SETROPTS GENERIC(FACILITY)
 - SETROPTS CLASSACT(FACILITY) RACLIST(FACILITY)
- 시작된 태스크 정의 활성화
 - SETROPTS GENERIC(STARTED)
 - RDEFINE STARTED ** STDATA(USER(=MEMBER) GROUP(STCGROUP) TRACE(YES))
 - SETROPTS CLASSACT(STARTED) RACLIST(STARTED)
- JES 작업 모니터에 대한 콘솔 보안 활성화
 - SETROPTS GENERIC(CONSOLE)
 - SETROPTS CLASSACT(CONSOLE) RACLIST(CONSOLE)
- JES 작업 모니터에 대한 운영자 명령 보호 활성화
 - SETROPTS GENERIC(OPERCMDS)
 - SETROPTS CLASSACT(OPERCMDS) RACLIST(OPERCMDS)
- RSE에 대한 애플리케이션 보호 활성화
 - SETROPTS GENERIC(APPL)
 - SETROPTS CLASSACT(APPL) RACLIST(APPL)
- RSE에 대해 PassTicket을 사용하여 보안 사인온 활성화
 - SETROPTS GENERIC(PTKTDATA)
 - SETROPTS CLASSACT(PTKTDATA) RACLIST(PTKTDATA)
- 프로그램 제어를 활성화하여 RSE가 신뢰할 수 있는 코드만 로드하도록 보장
 - RDEFINE PROGRAM ** ADDMEM('SYS1.CMDLIB'//NOPADCHK) UACC(READ)
 - SETROPTS WHEN(PROGRAM)

참고: PROGRAM 클래스에 * 프로파일이 이미 있으면 ** 프로파일을 작성하지 마십시오. 이 프로파일을 사용하면 보안 소프트웨어에서 사용하는 검색 경로가 모호하고 복잡해집니다. 이 경우, 기존 * 및 새 ** 정의를 병합해야 합니다. *Security Server RACF Security Administrator's Guide(SA22-7683)*에 설명된 대로 ** 프로파일을 사용하십시오.

주의: FTP와 같은 일부 제품의 경우 "WHEN PROGRAM"이 활성화되어 있으면 프로그램 제어가 필요합니다. 이 프로그램 제어를 테스트한 후에 프로덕션 시스템에서 활성화하십시오.

- (선택사항) X.509 HostIdMappings 및 확장 POE(Port Of Entry) 지원 활성화
 - SETROPTS GENERIC(SERVAUTH)
 - SETROPTS CLASSACT(SERVAUTH) RACLIST(SERVAUTH)

Developer for System z 사용자에게 대한 OMVS 세그먼트 정의

Developer for System z 사용자마다 0이 아닌 올바른 z/OS UNIX 사용자 ID(UID), 홈 디렉토리 및 셸 명령을 지정하는 RACF OMVS 세그먼트(또는 동등 기능)를 정의해야 합니다. 기본 그룹에는 그룹 ID가 있는 OMVS 세그먼트도 필요합니다.

선택적 통합 디버거를 사용할 때 애플리케이션을 디버깅하는 중인 사용자 ID가 활성화되며 해당 기본 그룹 또한 유효한 RACF OMVS 세그먼트 또는 그와 동등한 기능이 필요합니다.

다음 샘플 RACF 명령에서 #userid, #user-identifier, #group-name 및 #group-identifier 플레이스홀더를 실제 값으로 대체하십시오.

- ALTUSER #userid
OMVS(UID(#user-identifier) HOME(/u/#userid) PROGRAM(/bin/sh) NOASSIZEMAX)
- ALTGROUP #group-name OMVS(GID(#group-identifier))

Developer for System z 시작 태스크 정의

다음 샘플 RACF 명령은 보호된 사용자 ID(STCDBGM, STCJMON, STCRSE)와 지정된 FEKD, DBGMGR, JMON 및 RSED 시작 태스크를 작성합니다. #group-id 및 #user-id-* 플레이스홀더를 올바른 OMVS ID로 대체하십시오.

- ADDGROUP STCGROUP OMVS(GID(#group-id))
DATA('GROUP WITH OMVS SEGMENT FOR STARTED TASKS')
- ADDUSER STCDBM DFLTGRP(STCGROUP) NOPASSWORD
NAME('RDZ - DEBUG MANAGER')
OMVS(UID(#user-id-debug) HOME(/tmp) PROGRAM(/bin/sh))
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
- ADDUSER STCJMON DFLTGRP(STCGROUP) NOPASSWORD NAME('RDZ - JES JOBMONITOR')
OMVS(UID(#user-id-jmon) HOME(/tmp) PROGRAM(/bin/sh))
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
-

```

| ADDUSER STCRSE DFLTGRP(STCGROUP) NOPASSWORD NAME('RDZ - RSE DAEMON')
| OMVS(UID(#user-id-rse) HOME(/tmp) PROGRAM(/bin/sh) ASSIZEMAX(2147483647)
| )
| DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
|
| • RDEFINE STARTED DBGMR.* DATA('RDZ - DEBUG MANAGER')
|   STDATA(USER(STCDBM) GROUP(STCGROUP) TRUSTED(NO))
|
| •
|
| RDEFINE STARTED JMON.* DATA('RDZ - JES JOBMONITOR')
| STDATA(USER(STCJMON) GROUP(STCGROUP) TRUSTED(NO))
|
| •
|
| RDEFINE STARTED RSED.* DATA('RDZ - RSE DAEMON')
| STDATA(USER(STCRSE) GROUP(STCGROUP) TRUSTED(NO))
|
| •
|
| SETROPTS RACLIST(STARTED) REFRESH

```

참고:

- NOPASSWORD 키워드를 지정하여 시작된 태스크 사용자 ID가 보호되는지 확인하십시오.
- 이 uid에 부여된 z/OS UNIX 관련 권한 때문에 RSE 서버에 고유 OMVS uid가 있는지 확인하십시오.
- RSE 디먼이 적절히 작동하려면 주소 공간 크기가 커야 합니다(2GB). 사용자 ID STCRSE에 대해 OMVS 세그먼트의 ASSIZEMAX 변수에 이 값을 설정하십시오. 이 값을 설정하면 SYS1.PARMLIB(BPXPRMxx)의 MAXASSIZE 변경에 관계없이 RSE 디먼이 필요한 리전 크기를 갖습니다.
- RSE가 올바르게 작동하려면 스레드 수도 많아야 합니다. 사용자 ID STCRSE에 대해 OMVS 세그먼트의 THREADSMAX 변수에 한계를 설정할 수 있습니다. 이 한계를 설정하면 SYS1.PARMLIB(BPXPRMxx)의 MAXTHREADS 또는 MAXTHREADTASKS 변경에 관계없이 RSE가 필요한 스레드 한계를 갖습니다. 올바른 스레드 한계 값을 결정하려면 호스트 구성 참조서 (SA30-4501)의 "튜닝 고려사항"의 내용을 참조하십시오.
- JES 작업 모니터는 클라이언트 연결당 하나의 스레드를 사용하기 때문에 사용자 ID STCJMON은 OMVS 세그먼트에 THREADSMAX를 설정하기에 적합한 또 다른 후보입니다.
- 통합 디버거 시작 태스크(DBGMR)는 선택적 통합 디버거 기능에 의해서만 사용됩니다.

STCRSE 사용자 ID를 제한할 것을 고려하십시오. RESTRICTED 속성을 가진 사용자는 명확하게 액세스 권한이 부여되지 않은 보호(MVS) 자원에 액세스할 수 없습니다.

```
ALTUSER STCRSE RESTRICTED
```

제한된 사용자가 "기타" 권한 비트를 통해 z/OS UNIX 파일 시스템 자원에 대한 액세스를 확보하지 않도록 하려면 UACC(NONE)을 사용하여 UNIXPRIV 클래스에

RESTRICTED.FILESYS.ACCESS 프로파일을 정의하십시오. 사용자 ID 제한에 대한 자세한 정보는 *Security Server RACF Security Administrator's Guide*(SA22-7683)를 참조하십시오.

경고: 제한된 사용자 ID를 사용하는 경우, TSO **PERMIT** 또는 z/OS UNIX **setfac1** 명령을 사용하여 자원 액세스 권한을 명시적으로 추가하십시오. 이 자원에는 Developer for System z 문서가 UACC를 사용하는 자원(예: PROGRAM 클래스의 ** 프로파일) 또는 공통 z/OS UNIX 규약을 사용하는 자원(예: 모든 사람이 Java 라이브러리에 대한 읽기 및 실행 권한을 가짐)이 포함됩니다. 액세스를 테스트한 후에 프로덕션 시스템에서 활성화하십시오.

RSE를 보안 z/OS UNIX 서버로 정의

클라이언트의 스프레드에 대한 보안 환경을 작성 또는 삭제하려면 RSE에 BPX.SERVER 프로파일에 대한 UPDATE 액세스 권한이 필요합니다. 이 프로파일이 정의되지 않은 경우에는 RSE에 UID(0)이 필요합니다. 클라이언트가 연결할 수 있으려면 이 단계가 필요합니다.

디버그 스프레드에 대한 보안 환경을 작성 또는 삭제하려면 통합 디버거에 BPX.SERVER 프로파일에 대한 UPDATE 액세스 권한이 필요합니다. 이 프로파일이 정의되지 않은 경우에는 STCDBM 시작 태스크 사용자 ID에 UID(0)이 필요합니다. 이 허용은 선택적 통합 디버거 기능이 사용되는 경우에만 필요합니다.

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

경고: BPX.SERVER 프로파일을 정의하면 z/OS UNIX가 UNIX 레벨 보안에서 보다 안전한 z/OS UNIX 레벨 보안으로 전체 전환됩니다. 이러한 전환으로 다른 z/OS UNIX 애플리케이션 및 조작에 영향을 줄 수 있습니다. 보안을 테스트한 후에 프로덕션 시스템에서 활성화하십시오. 다른 보안 레벨에 대한 자세한 정보는 *UNIX System Services Planning*(GA22-7800)을 참조하십시오.

RSE에 대한 MVS 프로그램 제어 라이브러리 정의

BPX.SERVER에 대한 권한을 가진 서버는 프로그램으로 제어되는 정리된 환경에서 실행해야 합니다. 이 요구사항은 RSE로 호출되는 모든 프로그램도 프로그램으로 제어되어야 함을 의미합니다. MVS 로드 라이브러리의 경우, 보안 소프트웨어가 프로그램 제어를 관리합니다. 클라이언트가 연결할 수 있으려면 이 단계가 필요합니다.

RSE는 시스템(SYS1.LINKLIB), Language Environment의 런타임(CEE.SCEERUN*), ISPF의 TSO/ISPF Client Gateway(ISP.SISPLoad) 로드 라이브러리를 사용합니다.

- RALTER PROGRAM ** UACC(READ) ADDMEM('SYS1.LINKLIB'//NOPADCHK)
- RALTER PROGRAM ** UACC(READ) ADDMEM('CEE.SCEERUN'//NOPADCHK)
- RALTER PROGRAM ** UACC(READ) ADDMEM('CEE.SCEERUN2'//NOPADCHK)
- RALTER PROGRAM ** UACC(READ) ADDMEM('ISP.SISPLOAD'//NOPADCHK)
- SETROPTS WHEN(PROGRAM) REFRESH

참고: PROGRAM 클래스에 * 프로파일이 이미 있으면 ** 프로파일을 사용하지 마십시오. 이 프로파일을 사용하면 보안 소프트웨어에서 사용하는 검색 경로가 모호하고 복잡해 집니다. 이 경우, 기존 * 및 새 ** 정의를 병합해야 합니다. *Security Server RACF Security Administrator's Guide*(SA22-7683)에 설명된 대로 ** 프로파일을 사용하십시오.

선택적 서비스 사용을 지원하려면 다음 추가 필수 소프트웨어 라이브러리가 프로그램으로 제어되어야 합니다. 이 목록에는 IBM 파일 관리자와 같이 Developer for System z가 상호작용하는 제품에 특정한 데이터 세트가 포함되지 않습니다.

- 대체 REXX 런타임 라이브러리, SCLM 개발자 툴킷용
 - REXX.*.SEAGALT
- 시스템 로드 라이브러리, SSL 암호화용
 - SYS1.SIEALNKE
- Developer for System z 라이브러리, 통합 디버거용
 - FEK.SFEKAUTH

참고: LPA 배치를 위해 디자인된 라이브러리의 경우도 LINKLIST 또는 STEPLIB를 통해 액세스할 경우 프로그램 제어 권한이 필요합니다. 이 책에서는 다음 LPA 라이브러리 사용을 설명합니다.

- ISPF, TSO/ISPF Client Gateway용
 - ISP.SISPLPA
- REXX 런타임 라이브러리, SCLM 개발자 툴킷용
 - REXX.*.SEAGLPA
- Developer for System z, CARMA용
 - FEK.SFEKLPA

RSE에 대한 PassTicket 지원 정의

클라이언트의 비밀번호 또는 X.509 인증서 같은 다른 식별 수단은 연결 시 ID를 확인하는 데만 사용됩니다. 스레드 보안을 유지보수하는 데에는 PassTicket이 사용됩니다. 클라이언트가 연결할 수 있으려면 이 단계가 필요합니다.

PassTicket은 수명이 10분 정도인 시스템 생성 비밀번호입니다. 생성된 PassTicket은 비밀 키를 기반으로 합니다. 이 키는 64비트 숫자입니다(16개의 16진 문자). 다음 샘플 RACF 명령에서 key16 플레이스홀더를 사용자가 제공하는 16자 16진 문자열(문자 0-9 및 A-F)로 대체하십시오.

- ```
RDEFINE PTKTDATA FEKAPPL UACC(NONE) SSIGNON(KEYMASKED(key16))
APPLDATA('NO REPLAY PROTECTION - DO NOT CHANGE')
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
```
- ```
RDEFINE PTKTDATA IRRPTAUTH.FEKAPPL.* UACC(NONE)
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
```
- ```
PERMIT IRRPTAUTH.FEKAPPL.* CLASS(PTKTDATA) ACCESS(UPDATE) ID(STCRSE)
```
- ```
SETRPTS RACLIST(PTKTDATA) REFRESH
```

RSE는 FEKAPPL이 아닌 애플리케이션 ID 사용을 지원합니다. 이를 활성화하려면 54 페이지의 『_RSE_JAVAOPTS를 사용하여 여분의 Java 시작 매개변수 정의』에 설명된 대로 rsed.envvars의 "APPLID=FEKAPPL" 옵션을 주석 해제하고 사용자 정의하십시오. PTKTDATA 클래스 정의는 RSE에서 사용하는 실제 애플리케이션 ID와 일치해야 합니다.

대부분의 z/OS UNIX 애플리케이션 비밀 키를 열기 때문에 OMVSAPPL을 애플리케이션 ID로 사용해서는 안됩니다. 사용자 일괄처리 작업을 포함하여 대부분의 MVS 애플리케이션 비밀 키를 열기 때문에 기본 MVS 애플리케이션 ID(MVS 다음에 시스템의 SMF ID가 음)도 사용해서는 안됩니다.

참고:

- PTKTDATA 클래스가 정의되어 있으면 위에 나열된 프로파일을 작성하기 전에 이 클래스가 일반 클래스로 정의되어 있는지 확인하십시오. PTKTDATA 클래스에서 일반 문자에 대한 지원은 PassTicket에 Java 인터페이스가 도입되면서 z/OS 릴리스 1.7부터 새로운 기능입니다.
- IRRPTAUTH.FEKAPPL.* 정의의 와일드카드(*)를 올바른 사용자 ID 마스크로 대체하여 RSE가 PassTicket을 생성할 수 있는 사용자 ID를 제한하십시오.
- RACF 설정에 따라 프로파일을 정의하는 사용자도 프로파일의 액세스 목록에 있을 수 있습니다. PTKTDATA 프로파일의 경우 이 권한을 제거하십시오.
- JES 작업 모니터가 RSE가 제공한 PassTicket을 평가할 수 있으려면 JES 작업 모니터 및 RSE는 애플리케이션 ID가 동일해야 합니다. JES 작업 모니터의 경우, FEJJCENFG 구성 파일에서 설정된 애플리케이션 ID는 APPLID 지시문과 일치해야 합니다.

- 시스템에 암호화 제품이 설치되어 사용할 수 있는 경우, 추가된 보호에 대해 보안 사인은 애플리케이션 키를 암호화할 수 있습니다. 이를 수행하려면 KEYMASKED 대신 KEYENCRYPTED 키워드를 사용하십시오. 자세한 정보는 *Security Server RACF Security Administrator's Guide*(SA22-7683)를 참조하십시오.

경고: PassTicket이 올바르게 설정되지 않으면 클라이언트 연결 요청이 실패합니다.

RSE에 대한 애플리케이션 보호 정의

클라이언트 로그인 중에 RSE 디먼은 사용자가 애플리케이션을 사용할 수 있는지 확인합니다.

- RDEFINE APPL FEKAPPL UACC(READ) DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
- SETROPTS RACLIST(APPL) REFRESH

참고:

- 177 페이지의 『RSE에 대한 PassTicket 지원 정의』에 자세히 설명된 대로 RSE는 FEKAPPL이 아닌 애플리케이션 ID 사용을 지원합니다. APPL 클래스 정의는 RSE에서 사용하는 실제 애플리케이션 ID와 일치해야 합니다.
- 애플리케이션 ID가 APPL 클래스에 정의되어 있지 않으면 클라이언트 연결 요청이 성공합니다.
- 애플리케이션 ID가 정의되어 있으나 사용자에게 프로파일에 대한 READ 액세스 권한이 없는 경우에만 클라이언트 연결 요청이 실패합니다.

JES 명령 보안 정의

JES 작업 모니터는 35 페이지의 『FEJJCNFG, JES 작업 모니터 구성 파일』에 설명된 대로 CONSOLE_NAME 지시문을 사용하여 이름이 제어되는 확장 MCS(EMCS) 콘솔을 통해 사용자가 요청한 모든 JES 운영자 명령을 실행합니다.

다음 샘플 RACF 명령은 Developer for System z 사용자에게 제한된 JES 명령 세트(보류, 해제, 취소 및 제거)에 대한 조건부 액세스 권한을 제공합니다. 사용자가 JES 작업 모니터를 통해 명령을 실행할 경우 사용자는 실행 권한만 가집니다. #console 플레이스홀더를 실제 콘솔 이름으로 대체하십시오.

- RDEFINE OPERCMDS MVS.MCSOPER.#console UACC(READ)
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
-

```
RDEFINE OPERCMDS JES%.** UACC(NONE)
```

•

```
PERMIT JES%.** CLASS(OPERCMDS) ACCESS(UPDATE) WHEN(CONSOLE(JMON)) ID(*)
```

•

```
SETROPTS RACLIST(OPERCMDS) REFRESH
```

참고:

- MVS.MCSOPER.#console 프로파일을 정의하지 않으면 콘솔 사용이 허용됩니다.
- WHEN(CONSOLE(JMON))이 작동하려면 CONSOLE 클래스가 활성화되어 있어야 하지만 EMCS 콘솔의 경우 CONSOLE 클래스에서 실제 프로파일 검사는 없습니다.
- WHEN(CONSOLE(JMON)) 절에서 JMON을 실제 콘솔 이름으로 대체하지 마십시오. JMON 키워드는 콘솔 이름이 아닌 진입점 애플리케이션을 나타냅니다.

경고: 보안 소프트웨어에 유니버설 액세스 NONE을 사용하여 JES 명령을 정의하면 다른 애플리케이션과 조작에 영향을 줄 수 있습니다. 보안을 테스트한 후에 프로덕션 시스템에서 활성화하십시오.

표 21 및 표 22은 JES2 및 JES3에 대해 실행되는 운영자 명령과 이를 보호하는 데 사용할 수 있는 개별 보안 프로파일을 표시합니다.

표 21. JES2 작업 모니터 운영자 명령

조치	명령	OPERCMDS 프로파일	필수 액세스 권한
보류	\$Hx(jobid) x = {J, S 또는 T}	jesname.MODIFYHOLD.BAT jesname.MODIFYHOLD.STC jesname.MODIFYHOLD.TSU	UPDATE
해제	\$Ax(jobid) x = {J, S 또는 T}	jesname.MODIFYRELEASE.BAT jesname.MODIFYRELEASE.STC jesname.MODIFYRELEASE.TSU	UPDATE
취소	\$Cx(jobid) x = {J, S 또는 T}	jesname.CANCEL.BAT jesname.CANCEL.STC jesname.CANCEL.TSU	UPDATE
제거	\$Cx(jobid),P x = {J, S 또는 T}	jesname.CANCEL.BAT jesname.CANCEL.STC jesname.CANCEL.TSU	UPDATE

표 22. JES3 작업 모니터 운영자 명령

조치	명령	OPERCMDS 프로파일	필수 액세스 권한
보류	*F,J=jobid,H	jesname.MODIFY.JOB	UPDATE
해제	*F,J=jobid,R	jesname.MODIFY.JOB	UPDATE
취소	*F,J=jobid,C	jesname.MODIFY.JOB	UPDATE
제거	*F,J=jobid,C	jesname.MODIFY.JOB	UPDATE

참고:

- 값이 LIMITED 또는 NOLIMIT인 LIMIT_COMMANDS=가 JES 작업 모니터 구성 파일에 지정되어 있지 않으면 클라이언트 사용자 ID가 소유한 스푼 파일에 대해서만 보류, 해제, 취소, 제거 JES 운영자 명령과 JCL 표시 명령을 실행할 수 있습니다. 자세한 정보는 호스트 구성 참조서 (SA30-4501)의 "작업에 대한 조치 - 대상 제한사항"의 내용을 참조하십시오.
- LIMIT_VIEW=USERID가 JES 작업 모니터 구성 파일에 정의되어 있지 않으면 사용자가 스푼 파일을 찾아볼 수 있습니다. 자세한 정보는 호스트 구성 참조서 (SA30-4501)의 "스푼 파일에 대한 액세스"의 내용을 참조하십시오.
- 사용자에게 이러한 운영자 명령에 대한 권한이 없더라도 이러한 자원(예: JESINPUT, JESJOBS 및 JESSPOOL 클래스의 자원)을 보호하는 가능한 프로파일에 대한 충분한 권한이 있으면 JES 작업 모니터를 통해 여전히 작업을 제출하고 작업 출력을 읽을 수 있습니다.

TSO 세션에서 JMON 콘솔을 작성하여 JES 작업 모니터 서버의 ID를 가정하는 것은 보안 소프트웨어에서 금지됩니다. 콘솔을 작성할 수 있지만 진입점이 다릅니다(예: JES 작업 모니터 대 TSO). 이 책에 설명된 대로 보안이 설정되고 사용자에게 다른 방법을 통해 JES 명령에 대한 권한이 없으면 이 콘솔에서 실행된 JES 명령은 보안 검사에 실패합니다.

데이터 세트 프로파일 정의

대부분의 Developer for System z 데이터 세트에서는 사용자에게 대한 READ 액세스와 시스템 프로그래머에 대한 ALTER이면 충분합니다. #sysprog 플레이스홀더를 올바른 사용자 ID 또는 RACF 그룹 이름으로 대체하십시오. 또한 제품을 설치하여 구성된 시스템 프로그래머에게 올바른 데이터 세트 이름을 요청하십시오. FEK는 설치 중에 사용되는 기본 상위 레벨 규정자이고 FEK.#CUST는 사용자 정의 프로세스 중에 작성된 데이터 세트의 기본 상위 레벨 규정자입니다.

- ADDGROUP (FEK) OWNER(IBMUSER) SUPGROUP(SYS1)
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - HLQ STUB')
- ADDSD 'FEK.*.**' UACC(READ)
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
- PERMIT 'FEK.*.**' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
- SETOPTS GENERIC(DATASET) REFRESH

참고:

- 이 데이터 세트는 APF의 인증을 받았으므로 FEK.SFEKAUTH가 업데이트되지 않도록 보호합니다. FEK.SFEKLOAD 및 FEK.SFEKLPA의 경우도 마찬가지지만 이러한 데이터 세트는 프로그램 제어되기 때문에 여기서는 언급하지 않습니다.
- 이 책과 FEKRACF 작업에서의 샘플 명령은 EGN(Enhanced Generic Naming)이 활성화되어 있다고 가정합니다. EGN이 활성화되면 ** 규정자를 사용하여 DATASET 클래스의 규정자를 얼마든지 나타낼 수 있습니다. 시스템에서 EGN이 활성화되어 있지 않으면 **를 *로 대체하십시오. EGN에 대한 자세한 정보는 *Security Server RACF Security Administrator's Guide(SA22-7683)*를 참조하십시오.

일부 선택적 Developer for System z 컴포넌트에는 보안 데이터 세트 프로파일이 추가로 필요합니다. #sysprog, #ram-developer, #cicsadmin 플레이스홀더를 올바른 사용자 ID 또는 RACF 그룹 이름으로 대체하십시오.

- SCLM 개발자 툴킷의 긴/짧은 이름 변환이 사용되는 경우, 사용자에게는 맵핑 VSAM(FEK.#CUST.LSTRANS.FILE)에 대한 UPDATE 액세스 권한이 필요합니다.

```

-
ADDSD 'FEK.#CUST.LSTRANS.*.**' UACC(UPDATE)
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - SCLMDT')
-
PERMIT 'FEK.#CUST.LSTRANS.*.**' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
-
SETROPTS GENERIC(DATASET) REFRESH

```

- CARMA 저장소 액세스 관리자(RAM) 개발자에게는 CARMA VSAM(FEK.#CUST.CRA*)에 대한 UPDATE 액세스 권한이 필요합니다.

```

-
ADDSD 'FEK.#CUST.CRA*.*' UACC(READ)
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - CARMA')
-
PERMIT 'FEK.#CUST.CRA*.*' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
-
PERMIT 'FEK.#CUST.CRA*.*' CLASS(DATASET) ACCESS(UPDATE) ID(#ram-developer)
-
SETROPTS GENERIC(DATASET) REFRESH

```

- 애플리케이션 배치 관리자의 CRD(CICS Resource Definition) 서버가 사용되는 경우, CICS 관리자에게는 CRD 저장소 VSAM에 대한 UPDATE 액세스 권한이 필요합니다.

```

-
ADDSD 'FEK.#CUST.ADNREP*.*' UACC(READ)
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - ADN')
-

```

```
PERMIT 'FEK.#CUST.ADNREP*.*' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
```

```
PERMIT 'FEK.#CUST.ADNREP*.*' CLASS(DATASET) ACCESS(UPDATE) ID(#cicsadmin)
```

```
SETROPTS GENERIC(DATASET) REFRESH
```

- 애플리케이션 배치 관리자의 Manifest 저장소가 정의된 경우, 모든 CICS 트랜잭션 서버 사용자에게는 Manifest 저장소 VSAM에 대한 UPDATE 액세스 권한이 필요합니다.

```
ADDSD 'FEK.#CUST.ADNMAN*.*' UACC(UPDATE)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - ADN')
```

```
PERMIT 'FEK.#CUST.ADNMAN*.*' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
```

```
SETROPTS GENERIC(DATASET) REFRESH
```

READ 액세스도 제어되는 경우 보다 안전한 설정을 위해 다음 샘플 RACF 명령을 사용하십시오.

- uacc(none) 데이터 세트 보호

```
ADDGROUP (FEK)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - HLQ STUB')  
OWNER(IBMUSER) SUPGROUP(SYS1)"
```

```
ADDSD 'FEK.*.*' UACC(NONE)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
```

```
ADDSD 'FEK.SFEKAUTH' UACC(NONE)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
```

```
ADDSD 'FEK.SFEKLOAD' UACC(NONE)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
```

```
ADDSD 'FEK.SFEKLMOD' UACC(NONE)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
```

```
ADDSD 'FEK.SFEKPROC' UACC(NONE)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
```

```
ADDSD 'FEK.#CUST.PARMLIB' UACC(NONE)  
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
```

```

—
ADDSD 'FEK.#CUST.CNTL' UACC(NONE)
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')

—
ADDSD 'FEK.#CUST.SQL' UACC(NONE)
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')

—
ADDSD 'FEK.#CUST.LSTRANS.*.**' UACC(NONE)
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - SCLMDT')

—
ADDSD 'FEK.#CUST.CRA*.**' UACC(NONE)
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - CARMA')

—
ADDSD 'FEK.#CUST.ADNREP*.**' UACC(NONE)
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - ADN')

—
ADDSD 'FEK.#CUST.ADNMAN*.**' UACC(NONE)
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - ADN')
• 시스템 프로그래머가 모든 라이브러리를 관리하도록 허용

—
PERMIT 'FEK.*.** CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)

—
PERMIT 'FEK.SFEKAUTH CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)

—
PERMIT 'FEK.SFEKLOAD CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)

—
PERMIT 'FEK.SFEKLMOD CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)

—
PERMIT 'FEK.SFEKPROC CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)

—
PERMIT 'FEK.#CUST.PARMLIB CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)

—
PERMIT 'FEK.#CUST.CNTL CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
— PERMIT 'FEK.#CUST.SQL CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)

—
PERMIT 'FEK.#CUST.LSTRANS.*.**' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)

—
PERMIT 'FEK.#CUST.CRA*.**' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)

—

```

```
PERMIT 'FEK.#CUST.ADNREP*.*' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
```

—

```
PERMIT 'FEK.#CUST.ADNMAN*.*' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
```

- 클라이언트가 로드 및 exec 라이브러리에 액세스하도록 허용

—

```
PERMIT 'FEK.SFEKAUTH' CLASS(DATASET) ACCESS(READ) ID(*)
```

—

```
PERMIT 'FEK.SFEKLOAD' CLASS(DATASET) ACCESS(READ) ID(*)
```

—

```
PERMIT 'FEK.SFEKPROC' CLASS(DATASET) ACCESS(READ) ID(*)
```

—

```
PERMIT 'FEK.#CUST.CNTL' CLASS(DATASET) ACCESS(READ) ID(*)
```

— PERMIT 'FEK.#CUST.SQL' CLASS(DATASET) ACCESS(READ) ID(*)

참고: 모든 사람이 LPA에 상주하는 모든 코드에 액세스할 수 있으므로 FEK.SFEKLPA에 대해서는 허용이 필요하지 않습니다.

|

- 통합 디버거가 로드 라이브러리에 액세스하도록 허용

|

— PERMIT 'FEK.SFEKAUTH' CLASS(DATASET) ACCESS(READ) ID(STCDBM)

- JES 작업 모니터가 로드 및 매개변수 라이브러리에 액세스하도록 허용

—

```
PERMIT 'FEK.SFEKAUTH' CLASS(DATASET) ACCESS(READ) ID(STCJMON)
```

—

```
PERMIT 'FEK.#CUST.PARMLIB' CLASS(DATASET) ACCESS(READ) ID(STCJMON)
```

- (선택사항) 클라이언트가 SCLMDT용 긴/짧은 이름 변환 VSAM을 업데이트하도록 허용

—

```
PERMIT 'FEK.#CUST.LSTRANS*.*' CLASS(DATASET) ACCESS(UPDATE) ID(*)
```

- (선택사항) RAM 개발자가 CARMA용 CARMA VSAM을 업데이트하도록 허용

—

```
PERMIT 'FEK.#CUST.CRA*.*' CLASS(DATASET) ACCESS(UPDATE) ID(#ram-developer)
```

- (선택사항) CICS 사용자가 애플리케이션 배치 관리자용 CRD 저장소 VSAM을 읽을 수 있도록 허용

—

```
PERMIT 'FEK.#CUST.ADNREP*.*' CLASS(DATASET) ACCESS(READ) ID(*)
```

- (선택사항) CICS 관리자가 애플리케이션 배치 관리자용 CRD 저장소 VSAM을 업데이트하도록 허용

—

```
PERMIT 'FEK.#CUST.ADNREP*.*' CLASS(DATASET) ACCESS(UPDATE) ID(#cicsadmin)
```

- (선택사항) CICS 사용자가 애플리케이션 배치 관리자용 Manifest 저장소 VSAM을 업데이트하도록 허용

—

```
PERMIT 'FEK.#CUST.ADNMAN*.*' CLASS(DATASET) ACCESS(UPDATE) ID(*)
```

- (선택사항) CICS TS 서버가 양방향 및 애플리케이션 배치 관리자용 로드 라이브러리에 액세스하도록 허용

—

```
PERMIT 'FEK.SFEKLOAD' CLASS(DATASET) ACCESS(READ) ID(#cicsts)
```

- (선택사항) CICS TS 서버, IMS 리전 및 MVS 일괄처리 작업이 IRZ 메시지용 로드 라이브러리에 액세스하도록 허용

—

```
PERMIT 'FEK.SFEKLMOD' CLASS(DATASET) ACCESS(READ) ID(#cicsts)
```

```
PERMIT 'FEK.SFEKLMOD' CLASS(DATASET) ACCESS(READ) ID(#ims)
```

```
PERMIT 'FEK.SFEKLMOD' CLASS(DATASET) ACCESS(READ) ID(#batch)
```

- 보안 프로파일 활성화

—

```
SETROPTS GENERIC(DATASET) REFRESH
```

시스템 데이터 세트에 대한 READ 액세스를 제어하는 경우, Developer for System z 서버와 사용자에게 다음 데이터 세트에 대한 READ 권한을 제공해야 합니다.

- CEE.SCEERUN
- CEE.SCEERUN2
- CBC.SCLBDLL
- ISP.SISPLoad
- ISP.SISPLPA
- SYS1.LINKLIB
- SYS1.SIEALNKE
- SYS1.SIEAMIGE
- REXX.V1R4M0.SEAGLPA

참고: REXX 제품 패키지에 대한 대체 라이브러리를 사용하는 경우, 기본 REXX 런타임 라이브러리 이름은 이전 샘플에서 사용된 대로 REXX.*.SEAGLPA 대신 REXX.*.SEAGALT입니다.

I

RSE에 대한 z/OS UNIX 프로그램 제어 파일 정의

BPX.SERVER에 대한 권한을 가진 서버는 프로그램으로 제어되는 정리된 환경에서 실행해야 합니다. 이 요구사항은 RSE로 호출되는 모든 프로그램도 프로그램으로 제어되어야 함을 의미합니다. z/OS UNIX 파일의 경우, **extattr** 명령이 프로그램 제어를 관리합니다. 이 명령을 실행하려면 FACILITY 클래스의 BPX.FILEATTR.PROGCTL에 대한 READ 액세스 권한 또는 UID(0)이 필요합니다.

RSE 서버는 RACF의 Java 공유 라이브러리 (/usr/lib/libIRRRacf*.so)를 사용합니다.

- `extattr +p /usr/lib/libIRRRacf*.so`

참고:

- z/OS 1.9부터 /usr/lib/libIRRRacf*.so는 SMP/E RACF 설치 중에 프로그램 제어 모드로 설치됩니다.
- z/OS 1.10부터 /usr/lib/libIRRRacf*.so는 SAF의 일부로, 기본 z/OS와 함께 제공되므로 비RACF 고객도 사용할 수 있습니다.
- RACF 이외의 다른 보안 제품을 사용하는 경우 설정이 다를 수 있습니다. 자세한 정보는 보안 제품 문서를 참조하십시오.
- Developer for System z의 SMP/E 설치에는 내부 RSE 프로그램에 대한 프로그램 제어 비트를 설정합니다.
- **ls -Eog** z/OS UNIX 명령을 사용하여 프로그램 제어 비트의 현재 상태를 표시하십시오. 문자 **p**가 두 번째 문자열에 표시되면 파일은 프로그램에서 제어됩니다.

```
$ ls -Eog /usr/lib/libIRRRacf*.so
-rwxr-xr-x  aps-  2      69632 Oct  5  2007 /usr/lib/libIRRRacf.so
-rwxr-xr-x  aps-  2      69632 Oct  5  2007 /usr/lib/libIRRRacf64.so
```

보안 설정 확인

보안 관련 사용자 정의 결과를 표시하려면 다음 샘플 명령을 사용하십시오.

- 보안 설정 및 클래스
 - SETROPTS LIST
- 사용자에게 대한 OMVS 세그먼트
 - LISTUSER #userid NORACF OMVS
 - LISTGRP #group-name NORACF OMVS
- 시작된 태스크
 - LISTGRP STCGROUP OMVS
 - LISTUSER STCDBM OMVS
 - LISTUSER STCJMON OMVS

- LISTUSER STCRSE OMVS
- RLIST STARTED DBGMGR.* ALL STDATA
- RLIST STARTED JMON.* ALL STDATA
- RLIST STARTED RSED.* ALL STDATA
- RSE를 보안 z/OS UNIX 서버로 정의
 - RLIST FACILITY BPX.SERVER ALL
- RSE 대한 MVS 프로그램 제어 라이브러리
 - RLIST PROGRAM ** ALL
- RSE에 대한 PassTicket 지원
 - RLIST PTKTDATA FEKAPPL ALL SSIGNON
 - RLIST PTKTDATA IRRPTAUTH.FEKAPPL.* ALL
- RSE에 대한 애플리케이션 보호
 - RLIST APPL FEKAPPL ALL
- JES 명령 보안
 - RLIST CONSOLE JMON ALL
 - RLIST OPERCMDS MVS.MCSOPER.JMON ALL
 - RLIST OPERCMDS JES%.** ALL
- 데이터 세트 프로파일
 - LISTGRP FEK
 - LISTDSD PREFIX(FEK) ALL
- RSE 대한 z/OS UNIX 프로그램 제어 파일
 - ls -E /usr/lib/libIRRacf*.so

선택적으로 특정 사용자에게 대한 Developer for System z 동작을 지시하는 프로파일이 있을 수 있습니다. 이 프로파일은 FEK.** 필터와 일치하며 기본적으로 FACILITY 클래스에 있습니다. rsed.envvars의 _RSE_FEK_SAF_CLASS 지시문을 참조하십시오. **SEARCH** 명령을 사용하여 프로파일 이름을 나열할 수 있습니다. 프로파일에 대한 세부사항을 표시하려면 **RLIST** 명령을 사용하십시오.

- SEARCH CLASS(FACILITY) FILTER(FEK.**)
- RLIST FACILITY #profile-name ALL

제 10 장 마이그레이션 안내서

마이그레이션 고려사항

이 절에서는 제품의 이전 릴리스와 비교하여 설치 및 구성 변경사항을 강조표시합니다. 이 릴리스로 마이그레이션하기 위한 몇 가지 가이드라인도 제공합니다. 자세한 정보는 이 매뉴얼의 관련 절을 참조하십시오.

- IBM Rational Developer for System z, IBM WebSphere Developer for System z, zSeries용 IBM WebSphere Developer 또는 IBM WebSphere® Studio Enterprise Developer의 이전 사용자인 경우, 이 IBM Rational Developer for System z 버전 9.0 버전으로 업그레이드하기 전에 관련 사용자 정의된 파일을 저장하십시오.
- Developer for System z의 다중 인스턴스를 실행하려는 경우, **호스트 구성 참조서 (SA30-4501)**에 있는 "여러 인스턴스 실행"의 내용을 읽어 보십시오.
- 마이그레이션 시나리오가 2 이상의 릴리스에 해당하는 경우, 이전 릴리스가 없는 것처럼 사용자 정의를 다시 수행해야 합니다.

이전에 구성된 파일 백업

Developer for System z의 이전 사용자인 경우, 이 IBM Developer for System z 버전을 설치하기 전에 관련 사용자 정의된 파일을 저장하십시오.

사용자 정의할 수 있는 Developer for System z 파일은 다음 위치에 있습니다:

- 버전 8.0.1과 8.5
 - FEK.#CUST.RDZ*.*, 구성 유틸리티 작업 파일
 - FEK.SFEKSAMP, 일부 멤버는 FEKSETUP 샘플 작업이 FEK.#CUST.*로 복사하며, 여기서 *는 PARMLIB, PROCLIB, JCL, CNTL, ASM 및 COBOL과 같습니다.
 - FEK.SFEKSAMV
 - /usr/lpp/rdz/samples/, 일부 파일은 FEKSETUP 샘플 작업에 의해 /etc/rdz/와 /var/rdz/sc1mdt/*로 복사되며, 여기서 *는 CONFIG/, CONFIG/PROJECT/ 및 CONFIG/script/와 같습니다.

이전 Developer for System z 설정에서도 기타 제품에서 소유하는 구성 파일의 변경 사항에 대해 설명했습니다.

- 버전 8.0.1 및 8.5
 - SYS1.PARMLIB(BPXPRMxx)
z/OS UNIX 시스템 기본값을 설정합니다.
 - SYS1.PARMLIB(COMMNDxx)

- IPL 시 서버를 시작합니다.
- SYS1.PARMLIB(LPALSTxx)
- LPA에 FEK.SFEKLPA를 추가합니다.
- SYS1.PARMLIB(PROGxx)
- LINKLIST에 FEK.SFEKAUTH 및 FEK.SFEKLOAD를 추가합니다.
- (WLM)
- DB2 스토어드 프로시저의 애플리케이션 환경을 지정합니다.

버전 9.0 마이그레이션 참고

다음 마이그레이션 참고는 IBM Rational Developer for System z 버전 9.0에만 해당됩니다. 이러한 참고는 IBM Rational Developer for System z 버전 9.0.0에서 버전 9.0.1로의 마이그레이션에 유효하며, 기존 버전 9.0.0 마이그레이션 참고에 추가된 사항입니다.

나열된 모든 변경사항은 버전 9.0.1 이후로 유효합니다.

IBM Rational Developer for System z, FMID HHOP900

- CARMA: CA Endeavor® SCM RAM의 CRADEF VSAM 파일이 업데이트되었습니다.
- CARMA: LPA에 상주하는 CRASTART 로드 모듈이 업데이트되었습니다. LPA 업데이트가 필요합니다.
- CARMA: CARMA 시작 중에 사용자 엑시트를 실행하기 위한 지원이 추가되었습니다.
- CARMA: 시작 인수 처리 중 RAM에 대한 지원이 추가되었습니다.
- CARMA: 사용자 정의할 수 있는 새 멤버가 추가되었습니다.
 - CRAEXIT: 샘플 CARMA 사용자 엑시트입니다.
 - CRAALLOC: 사용자 정의 RAM CARMA 호출에 대한 할당 exec입니다.
 - CRACFG: CA Endeavor® SCM RAM 사용 구성 파일입니다.
- CARMA: 사용자 정의할 수 있는 다음 멤버가 변경되었습니다.
 - CRASRV.properties
 - crastart.conf
 - crastart.endevor.conf
 - CRASUBMT
 - CRASUBCA

- CRANDVRA
- CARMA: crastart.endevor.conf 및 CRASUBCA에서 CA Endeavor® SCM RAM에 대한 추가 DD 문이 추가되었습니다.
 - CRAPARM(CRANDVRA에서 할당됨)
 - CRACFG
- CARMA: 비 "CA Endeavor® SCM RAM"에 대한 추가 DD 문이 추가되었습니다.
 - CRAPARM(CRAALLOC에서 할당됨)
- 사용자 정의: 이제 FEKSETUP JCL이 새 멤버를 처리합니다.
 - CRACFG: FEK.#CUST.PARMLIB(CRACFG)로 복사되었습니다.
- RSE: 새 선택적 지시문이 rsed.envvars에 추가되었습니다.
 - (_RSE_JAVAOPTS) -Dsearch.server.limit.timeout
 - (_RSE_JAVAOPTS) -Dkeep.all.logs
 - (_RSE_JAVAOPTS) -Daudit.users
 - RSE_UBLD_DD
 - RSE_UBLD_STEPLIB
- RSE: rsed.envvars에서 사용자 정의할 수 없는 지시문의 기본값이 변경되었습니다.
 - (_RSE_JAVAOPTS) -DDSTORE_SPIRIT_ON
- zUnit: 새 선택적 시작 인수가 추가되었습니다.
 - CLOCALE / -l

IBM Rational Developer for System z 호스트 유틸리티, FMID HAKG900

이 제품에 대한 버전 9.0 특정의 마이그레이션 참고가 없습니다.

버전 8.5에서 버전 9.0으로 마이그레이션

이러한 참고사항은 기본 버전 8.5에서 버전 9.0으로의 마이그레이션에 적용됩니다. 여기에는 버전 8.5 유지보수의 일부로서 이미 문서화된 변경사항이 포함됩니다. 유지보수 스트림의 일부여서 이미 구현된 변경사항은 그러한 변경사항이 소개된 릴리스와 함께 표시됩니다.

IBM Rational Developer for System z, FMID HHOP900

- MVS 및 z/OS UNIX 컴포넌트의 기본 SMP/E 설치 위치가 바뀌지 않았으므로 FEK.* 및 /usr/lpp/rdz/*가 그대로 사용됩니다.

- CARMA: 사용자 정의할 수 있는 CA Endevor® SCM 일괄처리 작업의 새 지원을 사용하려면 CA Endevor® SCM RAM의 CRADEF 및 CRASTRS VSAM 파일을 업데이트해야 합니다(버전 8.5.1부터).
- CARMA: CRADEF VSAM을 작성하는 동안 RAM을 사용 안함으로 설정하기 위해 추가된 지원입니다(버전 8.5.1부터).
- CARMA: CRASRV.properties의 상대 파일 참조를 위해 추가된 지원입니다(버전 8.5.1부터).
- CARMA: 새 샘플 멤버가 추가되었습니다.
 - CRABJOBC: CA Endevor® SCM 일괄처리 작업용 기본 JOB 카드입니다(버전 8.5.1부터).
- CARMA: 사용자 정의할 수 있는 다음 멤버가 변경되었습니다.
 - CRASRV.properties(버전 8.5.1부터)
 - carma.startup.rex(버전 8.5.1부터)
 - CRA\$VCAD(버전 8.5.1부터)
 - CRA\$VDEF(버전 8.5.1부터)
 - CRABATCA(버전 8.5.1부터)
 - CRABCFG(버전 8.5.1부터)
 - CRANDVRA(버전 8.5.1부터)
- CARMA: crastart.endevor.conf 및 CRASUBCA에서 CA Endevor® SCM RAM에 대한 DD 문이 추가되었습니다.
 - CRABJCLO: CRANDVRA가 할당함(버전 8.5.1부터)
 - ENHCEDIT: CRANDVRA가 할당함(버전 8.5.1부터)
- 사용자 정의: 이제 FEKSETUP JCL이 새 멤버를 처리합니다.
 - CRABJOBC: FEK.#CUST.CNTL(CRABJOBC)에 복사되었습니다(버전 8.5.1부터).
 - ELAXFSP: FEK.#CUST.PROCLIB(ELAXFSP)에 복사되었습니다(버전 9.0부터).
 - ELAXFSQL: FEK.#CUST.PROCLIB(ELAXFSQL)에 복사되었습니다(버전 9.0부터).
 - FEKTEP2: FEK.#CUST.SQL(FEKTEP2)에 복사되었습니다(버전 9.0부터).
 - FEKTIAD: FEK.#CUST.SQL(FEKTEP2)에 복사되었습니다(버전 9.0부터).
- Fault Analyzer 통합: FAI에 대한 지원은 중지되었습니다. 이 변경은 여전히 FAI를 사용하는 기존 클라이언트와 호환될 수 없습니다.
- JES 작업 모니터 - 새 운영자 명령이 JMON 시작 태스크에 추가되었습니다.
 - MODIFY USERS(버전 8.5.1부터)
 - MODIFY -T{N | E | I | V}(버전 8.5.1부터)

- MODIFY -M{N | E | W | I | V}(버전 8.5.1부터)
- MODIFY TRACE {N | E | I | V}(버전 9.0부터)
- MODIFY MESSAGE {N | E | W | I | V}(버전 9.0부터)
- JES 작업 모니터 - 새 선택적 지시문이 FEJJCNFG에 추가되었습니다.
 - LOOPBACK_ONLY(버전 9.0부터)
- JES 작업 모니터 - 선택적 지시문이 FEJJCNFG에서 제거되었습니다.
 - _BPXK_SETIBMOPT_TRANSPORT(버전 9.0부터)
- 문제점 판별: 현재 FEKLOGS JCL이 사용자 로그를 수집하기 위해 여러 사용자 ID를 지정하도록 지원합니다(버전 8.5.1부터).
- 문제점 판별: FEKLOGS JCL은 이제 DD REFORMAT을 사용하여 더 빠른 문제점 판별을 위해 다시 형식화된 로그를 수집합니다(버전 8.5.1부터).
- 문제점 판별: 사용자 정의할 수 있는 다음 멤버가 변경되었습니다.
 - FEKLOGS(버전 8.5.1부터)
- RSE - 새 운영자 명령이 RSED 시작 태스크에 추가되었습니다.
 - MODIFY DISPLAY OWNER,DATASET=dataset(버전 9.0부터)
 - MODIFY DEBUG GC,PID=pid(버전 9.0부터)
- RSE: 사용자 정의가 불가능한 새 지시문이 rsed.envvars에 추가되었습니다.
 - _CMDSERV_BASE_HOME(버전 8.5.1부터)
 - _CMDSERV_CONF_HOME(버전 8.5.1부터)
 - _CMDSERV_WORK_HOME(버전 8.5.1부터)
 - RSE_DSN_SFELKLOAD(버전 9.0부터)
 - (_RSE_JAVAOPTS) -Dlock.info.timeout(버전 9.0부터)
 - (_RSE_JAVAOPTS) -DDSTORE_INITIAL_SIZE(버전 9.0부터)
 - (_RSE_JAVAOPTS) -DDSTORE_MAX_FREE(버전 9.0부터)
- RSE: 새 필수 지시문이 rsed.envvars에 추가되었습니다.
 - RSE_HLQ(버전 9.0부터)
- RSE: 새 선택적 지시문이 rsed.envvars에 추가되었습니다.
 - (_RSE_JAVAOPTS) -DRSE_DSICALL(버전 8.5.1부터)
 - (_RSE_JAVAOPTS) -DDISABLE_REMOTE_INDEX_SEARCH(버전 8.5.1부터)
 - (_RSE_JAVAOPTS) -DDISABLE_TEXT_SEARCH(버전 9.0부터)
 - (_RSE_JAVAOPTS) -Dsearch.server.limit.hits(버전 9.0부터)
 - (_RSE_JAVAOPTS) -Dsearch.server.limit.datasets(버전 9.0부터)
 - (_RSE_JAVAOPTS) -Dsearch.server.limit.lines(버전 9.0부터)

- (_RSE_JAVAOPTS) -DDSTORE_SSL_ALGORITHM(버전 9.0부터)
- RSE: rsed.envvars에 있는 사용자 정의할 수 없는 지시문의 기본값이 변경되었습니다.
 - (_RSE_JAVAOPTS) -DSPIRIT_EXPIRY_TIME(버전 9.0부터)
- RSE: rsed.envvars에 있는 선택적 지시문의 기본값이 변경되었습니다.
 - (_RSE_JAVAOPTS) -Xms(버전 8.5.1부터)
 - (_RSE_JAVAOPTS) -Xmx(버전 8.5.1부터)
 - (_RSE_JAVAOPTS) -Dmaximum.clients(버전 8.5.1부터)
 - (_RSE_JAVAOPTS) -Dmaximum.threads(버전 8.5.1부터)
 - CGI_ISPPREF(버전 9.0부터)
- 보안: 새 보안 프로파일에 대한 지원이 추가되었습니다.
 - FEK.USR.** (버전 8.5.1부터)

구성 가능한 파일

표 23에서는 버전 9.0에서 사용자 정의된 파일의 개요를 표시합니다. Developer for System z 샘플 라이브러리(FEK.SFEKSAMP, FEK.SFEKSAMV 및 /usr/lpp/rdz/samples/)에는 컴파일하기 위한 샘플 CARMA 소스 코드 및 작업과 같이 여기에 나열된 멤버 이상의 사용자 정의 가능한 멤버가 포함됩니다. Developer for System z 호스트 유틸리티 샘플 라이브러리(AKG.SAKGSAMP 및 /usr/lpp/rdzutil/samples)에는 샘플 코드 검토 사후 처리 스크립트와 같이 여기서 나열된 것보다 더 많은 사용자 정의 가능한 멤버가 포함됩니다.

다음 멤버 및 파일은 더 이상 사용자 정의할 수 없고 더 이상 사용되지 않습니다.

- LOCKD 시작 태스크
- ELAXMSAM 샘플 DB2 스토어드 프로시저
- DB2 스토어드 프로시저에 대한 ELAXMJCL 샘플 JCL

참고: 샘플 작업 FEKSETUP은 표시된 모든 멤버를 다른 데이터 세트 및 디렉토리로 복사하고 기본값은 FEK.#CUST.* 및 /etc/rdz/*입니다.

표 23. 버전 9.0 사용자 정의

멤버/파일	기본 위치	목적	마이그레이션 참고
FEKSETUP	FEK.SFEKSAMP [FEK.#CUST.JCL]	데이터 세트 및 디렉토리를 작성하고 이를 사용자 정의 가능한 파일로 채우는 JCL	더 이상 사용되지 않는 파일의 조치를 제거하고 새로운 파일의 조치를 추가하도록 업데이트 됨
JMON	FEK.SFEKSAMP(FEJJJCL) [FEK.#CUST.PROCLIB]	JES 작업 모니터의 JCL	없음

표 23. 버전 9.0 사용자 정의 (계속)

멤버/파일	기본 위치	목적	마이그레이션 참고
FEJJJCL	FEK.SFEKSAMP [FEK.#CUST.PROCLIB(JMON)]	JMON 멤버의 이름	JMON 멤버 참조
RSED	FEK.SFEKSAMP(FEKRSED) [FEK.#CUST.PROCLIB]	RSE 디먼의 JCL	없음
FEKRSED	FEK.SFEKSAMP [FEK.#CUST.PROCLIB(RSED)]	RSED 멤버의 이름	RSED 멤버 참조
ELAXF*	FEK.SFEKSAMP [FEK.#CUST.PROCLIB]	원격 프로젝트 빌드 등을 위한 JCL	ELAXFSP 및 ELAXFSQL은 새로 추가, ELAXFCOC 및 ELAXFCP1은 Cobol 버전 5 지원을 위해 업데이트
FEKRACF	FEK.SFEKSAMP [FEK.#CUST.JCL]	보안 정의를 위한 JCL	없음
FEJJCNGF	FEK.SFEKSAMP [FEK.#CUST.PARMLIB]	JES 작업 모니터 구성 파일	새로운 선택적 지시문이 추가되었습니다. 기존의 선택적 지시문은 제거됩니다.
FEJT50	FEK.SFEKSAMP [FEK.#CUST.CNTL]	TSO 제출을 위한 JCL	없음
CRA\$VMSG	FEK.SFEKSAMP [FEK.#CUST.JCL]	CARMA 메시지 VSAM을 작성하기 위한 JCL	없음
CRA\$VDEF	FEK.SFEKSAMP [FEK.#CUST.JCL]	CARMA 구성 VSAM을 작성하기 위한 JCL	RAM을 제외하도록 지원 추가
CRA\$VSTR	FEK.SFEKSAMP [FEK.#CUST.JCL]	CARMA 사용자 정의 정보 VSAM을 작성하기 위한 JCL	없음
CRA\$VCAD	FEK.SFEKSAMP [FEK.#CUST.JCL]	CA Endeavor® SCM RAM에 대한 CARMA 구성 VSAM 작성하는 JCL	RAM 및 VSAM 입력을 제외하도록 추가된 지원이 변경되었습니다.
CRA\$VCAS	FEK.SFEKSAMP [FEK.#CUST.JCL]	CA Endeavor® SCM RAM에 대한 CARMA 사용자 정의 정보 VSAM을 작성하는 JCL	VSAM 입력이 변경됨

표 23. 버전 9.0 사용자 정의 (계속)

멤버/파일	기본 위치	목적	마이그레이션 참고
CRASUBMT	FEK.SFEKSAMP [FEK.#CUST.CNTL]	CARMA 일괄 처리 시작 CLIST	없음
CRASUBCA	FEK.SFEKSAMP [FEK.#CUST.CNTL]	CA Endeavor® SCM RAM을 위한 CARMA 일괄처리 시작 CLIST	없음
CRABCFG	FEK.SFEKSAMP [FEK.#CUST.PARMLIB]	CA Endeavor® SCM RAM의 CARMA 일괄 처리 조치 구성	새 지시문 추가
CRABATCA	FEK.SFEKSAMP [FEK.#CUST.CNTL]	CA Endeavor® SCM RAM의 CARMA 일괄 처리 조치 JCL	변수 JOB 카드 지원 추가
CRASHOW	FEK.SFEKSAMP [FEK.#CUST.PARMLIB]	CA Endeavor® SCM RAM의 CARMA 구성	없음
CRATMAP	FEK.SFEKSAMP [FEK.#CUST.PARMLIB]	CA Endeavor® SCM RAM의 CARMA 구성	없음
CRANDVRA	FEK.SFEKPROC	CA Endeavor® SCM RAM의 CARMA 할당 REXX	새 DD 할당 추가
CRA#VSLM	FEK.SFEKSAMP [FEK.#CUST.JCL]	SCLM RAM 메시지 VSAM 을 작성하는 JCL	없음
CRA#ASLM	FEK.SFEKSAMP [FEK.#CUST.JCL]	SCLM RAM 데이터 세트를 작성하기 위한 JCL	없음
CRA#VPDS	FEK.SFEKSAMP [FEK.#CUST.JCL]	PDS RAM 메 시지 VSAM을 작성하는 JCL	없음
CRA#UADD	FEK.SFEKSAMP [FEK.#CUST.JCL]	RAM 정의를 병합하는 JCL	없음
CRA#UQRY	FEK.SFEKSAMP [FEK.#CUST.JCL]	RAM 정의를 추출하는 JCL	없음
CRAXJCL	FEK.SFEKSAMP [FEK.#CUST.ASM]	IRXJCL 대체 를 위한 샘플 소스 코드	없음

표 23. 버전 9.0 사용자 정의 (계속)

멤버/파일	기본 위치	목적	마이그레이션 참고
CRA#CIRX	FEK.SFEKSAMP [FEK.#CUST.JCL]	CRAXJCL을 컴파일하는 JCL	없음
ADNCSDRS	FEK.SFEKSAMP [FEK.#CUST.JCL]	기본 CICS 리전에 RESTdul CRD 서버를 정의하는 JCL	없음
ADNCSDTX	FEK.SFEKSAMP [FEK.#CUST.JCL]	CICS 리전에 대체 트랜잭션 ID를 정의하는 JCL	없음
ADNTXNC	FEK.SFEKSAMP [FEK.#CUST.JCL]	대체 트랜잭션 ID를 작성하는 JCL	없음
ADNMSGHC	FEK.SFEKSAMP [FEK.#CUST.JCL]	ADNMSGHS를 컴파일하는 JCL	없음
ADNMSGHS	FEK.SFEKSAMP [FEK.#CUST.COBOL]	파이프라인 메시징 핸들러의 샘플 소스 코드	없음
ADNVCRD	FEK.SFEKSAMP [FEK.#CUST.JCL]	CRD 저장소를 작성하는 JCL	없음
ADNCSDWS	FEK.SFEKSAMP [FEK.#CUST.JCL]	기본 CICS 리전에 웹 서비스 CRD 서버를 정의하는 JCL	없음
ADNCSDAR	FEK.SFEKSAMP [FEK.#CUST.JCL]	비기본 CICS 리전에 CRD 서버를 정의하는 JCL	없음
ADNJSPAU	FEK.SFEKSAMP [FEK.#CUST.JCL]	CRD 기본값을 업데이트하는 JCL	없음
ADNVMFST	FEK.SFEKSAMP [FEK.#CUST.JCL]	Manifest 저장소를 작성 및 정의하는 JCL	없음
FEKTEP2	FEK.SFEKSAMP [FEK.#CUST.SQL]	ELAXF*가 사용하는 SQL 명령 입력	신규, 사용자 정의는 선택사항임
FEKTIAD	FEK.SFEKSAMP [FEK.#CUST.JCL]	ELAXF*가 사용하는 SQL 명령 입력	신규, 사용자 정의는 선택사항임
AZUZUNIT	FEK.SFEKSAMP [FEK.#CUST.PROCLIB]	zUnit에 대한 JCL 프로시저	없음

표 23. 버전 9.0 사용자 정의 (계속)

멤버/파일	기본 위치	목적	마이그레이션 참고
FEKRNPLI	FEK.SFEKSAMP [FEK.#CUST.CNTL]	프리프로세서 프레임워크 내에서 PL/I 컴파일러를 호출하기 위한 REXX	없음
FEKLOGS	FEK.SFEKSAMP [FEK.#CUST.JCL]	로그 파일을 수집하기 위한 JCL	추가 검사가 추가되었습니다. 이전 파일에 대한 사용자 정의도 다시 수행되어야 합니다.
rsed.envvars	/usr/lpp/rdz/samples/ [/etc/rdz/]	RSE 환경 변수	이전 사본은 이 사본으로 대체되고 사용자 정의가 다시 수행되어야 합니다.
ISPF.conf	/usr/lpp/rdz/samples/ [/etc/rdz/]	TSO/ISPF Client Gateway 구성 파일	없음
CRASRV.properties	/usr/lpp/rdz/samples/ [/etc/rdz/]	CARMA 구성 파일	기본값에 대해 지원이 추가됨
crastart.conf	/usr/lpp/rdz/samples/ [/etc/rdz/]	CRASTART 사용법을 위한 CARMA 구성 파일	없음
crastart.endevor.conf	/usr/lpp/rdz/samples/ [/etc/rdz/]	CA Endevor® SCM RAM에 대한 CRASTART 사용의 CARMA 구성 파일	없음
include.conf	/usr/lpp/rdz/samples/ [/etc/rdz/]	C/C++ 콘텐츠 지원을 위해 강제로 포함됨	없음
ssl.properties	/usr/lpp/rdz/samples/ [/etc/rdz/]	RSE SSL 구성 파일	없음
rsecomm.properties	/usr/lpp/rdz/samples/ [/etc/rdz/]	RSE 추적 구성 파일	없음
pushtoclient.properties	/usr/lpp/rdz/samples/ [/etc/rdz/]	클라이언트 구성 파일에 정보를 푸시함	없음

IBM Rational Developer for System z 호스트 유틸리티, FMID HAKG900

버전 8.5에는 동등한 기능이 없으므로 마이그레이션 노트가 없습니다.

구성 파일

표 24에서는 버전 9.0에서 사용자 정의된 파일의 개요를 표시합니다. Developer for System z 호스트 유틸리티 샘플 라이브러리(AKG.SAKGSAMP 및 /usr/lpp/rdzutil/samples)에는 샘플 코드 검토 사후 처리 스크립트와 같이 여기서 나열된 것보다 더 많은 사용자 정의 가능한 멤버가 포함됩니다.

참고: 샘플 작업 AKGSETUP은 표시된 모든 멤버를 다른 데이터 세트에 복사하고 기본값은 AKG.#CUST.*입니다.

표 24. 호스트 유틸리티 버전 9.0 사용자 정의. 호스트 유틸리티 버전 9.0 사용자 정의

멤버 또는 파일	기본 위치	목적	마이그레이션 참고
AKGSETUP	AKG.SAKGSAMP [AKG.#CUST.JCL]	데이터 세트를 작성하고 이를 사용자 정의 가능한 파일로 채우는 JCL	새로 작성
AKGCC	AKG.SAKGSAMP [AKG.#CUST.PROCLIB]	코드 적용을 위한 JCL	새로 작성
AKGCR	AKG.SAKGSAMP [AKG.#CUST.PROCLIB]	코드 검토를 위한 JCL	새로 작성
AKGCRADD	AKG.SAKGSAMP [AKG.#CUST.JCL]	씨드파티 코드를 코드 검토에 추가하기 위한 JCL	새로 작성

버전 8.5 마이그레이션 참고

다음 마이그레이션 참고는 버전 8.5에만 해당됩니다. 이러한 참고는 IBM Rational Developer for System z 버전 8.5.0에서 버전 8.5.1로의 마이그레이션에 유효하며, 기존 버전 8.5.0 마이그레이션 참고에 추가된 사항입니다.

여기 나열된 변경사항 모두는 버전 8.5.1 이후로 유효합니다.

- CARMA: 사용자 정의할 수 있는 CA Endeavor® SCM 일괄처리 조치에 대해 새 지원을 사용하려면 CA Endeavor® SCM RAM의 CRADEF 및 CRASTRS VSAM 파일을 업데이트해야 합니다.
- CARMA: CRADEF VSAM 작성 도중 RAM을 사용 안함으로 설정하기 위해 추가된 지원입니다.
- CARMA: CRASRV.properties에서 상대 파일 참조를 위해 추가된 지원입니다.
- CARMA: 새 샘플 멤버가 추가되었습니다.
 - CRABJOBC: CA Endeavor® SCM 일괄처리 조치의 기본 JOB 카드입니다.
- CARMA: 사용자 정의할 수 있는 다음 멤버가 변경되었습니다.

- CRASRV.properties
- carma.startup.rex
- CRA\$VCAD
- CRA\$VDEF
- CRABATCA
- CRABCFG
- CRANDVRA
- CARMA: crastart.endevor.conf 및 CRASUBCA에서 CA Endevor® SCM RAM에 대한 DD 문이 추가되었습니다.
 - CRABJCLO - CRANDVRA에서 할당됨
 - ENHCEDIT - CRANDVRA에서 할당됨
- 사용자 정의: 이제 FEKSETUP JCL이 새 멤버를 처리합니다.
 - CRABJOBC: FEK.#CUST.CNTL(CRABJOBC)로 복사되었습니다.
- JES 작업 모니터 - 새 운영자 명령이 JMON 시작 태스크에 추가되었습니다(버전 8.0.3.2부터).
 - MODIFY USERS
 - MODIFY -T{N | E | I | V}
 - MODIFY -M{N | E | W | I | V}
- 문제점 판별: FEKLOGS JCL에서는 이제 사용자 로그를 수집하기 위해 여러 사용자 ID를 지정하도록 지원합니다.
- 문제점 판별: FEKLOGS JCL에서는 이제 DD REFORMAT을 사용하여 신속한 문제점 판별을 위해 다시 형식화된 로그를 수집합니다.
- 문제점 판별: 사용자 정의할 수 있는 다음 멤버가 변경되었습니다.
 - FEKLOGS
- RSE: 사용자 정의가 불가능한 새 지시문이 rsed.envvars에 추가되었습니다.
 - _CMDSERV_BASE_HOME
 - _CMDSERV_CONF_HOME
 - _CMDSERV_WORK_HOME
- RSE: 새 선택적 지시문이 rsed.envvars에 추가되었습니다.
 - (_RSE_JAVAOPTS) -DRSE_DSICALL
 - (_RSE_JAVAOPTS) -DDISABLE_REMOTE_INDEX_SEARCH
- RSE: rsed.envvars에 있는 선택적 지시문의 기본값이 변경되었습니다.
 - (_RSE_JAVAOPTS) -Xms
 - (_RSE_JAVAOPTS) -Xmx

- (_RSE_JAVAOPTS) -Dmaximum.clients
- (_RSE_JAVAOPTS) -Dmaximum.threads
- RSE: rsed.envvars에 있는 사용자 정의할 수 없는 지시문의 기본값이 변경되었습니다.
 - (_RSE_JAVAOPTS) -DDSTORE_SPIRIT_ON
- 보안: 새 보안 프로파일에 대한 지원이 추가되었습니다.
 - FEK.USR.**

버전 8.0.1에서 버전 8.5로 마이그레이션

이러한 주의사항은 기본 버전 8.0.1에서 8.5으로의 마이그레이션에 적용됩니다. 여기에는 버전 8.0.1 유지보수의 일부로서 이미 문서화된 변경사항이 포함됩니다. 유지보수 스트림의 일부여서 이미 구현된 변경사항은 그러한 변경사항이 소개된 릴리스와 함께 표시됩니다.

IBM Rational Developer for System z, FMID HHOP850

- MVS 및 z/OS UNIX 컴포넌트의 기본 SMP/E 설치 위치가 바뀌지 않았으므로 FEK.* 및 /usr/lpp/rdz/*가 그대로 사용됩니다.
- CARMA - LPA에 상주하는 CRASTART 로드 모듈이며 업데이트되었습니다. LPA 업데이트가 필요합니다(단일 버전 8.0.3.2).
- CARMA - CRAMSG VSAM을 업데이트해야 합니다(버전 8.0.3 및 8.5부터).
- CARMA - CA Endeavor® SCM 일괄처리 작업(버전 8.0.3부터) 및 CA Endeavor® SCM 패키지(버전 8.0.3부터)에 대한 새 지원을 사용하려면 CA Endeavor® SCM RAM용 CRADEF 및 CRASTRS VSAM 파일을 업데이트해야 합니다.
- CARMA - 신규 CRADEF 및 CRASTRS VSAM 입력이 추가되어 CA Endeavor® SCM 요소 메뉴에서 CA Endeavor® SCM 패키지 조치를 복원할 수 있게 되었습니다.
 - CRA0VPKD - CRADEF에 병합됩니다.
 - CRA0VPKS - CRASTRS에 병합됩니다.
- CARMA - 새 샘플 멤버가 추가되었습니다(버전 8.0.3부터).
 - CRABCFG - CA Endeavor® SCM 일괄처리 작업용 구성 파일입니다.
 - CRABATCA - CA Endeavor® SCM 일괄처리 작업용 샘플 작업입니다.
- CARMA - 사용자 정의할 수 있는 다음 멤버가 변경되었습니다(버전 8.0.3, 8.0.3.1 및 8.5부터).
 - CRANDVRA
 - CRASHOW
 - CRASRV.properties

- CRABCFG
- CARMA - crastart.endevor.conf 및 CRASUBCA에서 CA Endeavor® SCM RAM에 DD 문이 추가되었습니다(버전 8.0.3부터).
 - CRABCFG
 - CRABSKEL
 - PKGSCLS(CRANDVRA에서 할당)
- 엔터프라이즈 서비스 도구 - IRZ 로드 모듈 및 메시지 모듈을 새 라이브러리로 이동했습니다(버전 8.5부터).
 - FEK.SFEKLMOD(IRZ* IIRZ*)
- 파일 관리자 통합은 제거되었습니다(버전 8.5부터). 형식화되지 않은 QSAM 편집 등과 같은 일부 기능은 현재 Developer for System z에서 처리하는 일반 데이터 세트의 일부입니다. 카피북을 사용한 형식화된 데이터 편집 등과 같은 고급 기능 또는 포함 파일에는 Eclipse용 IBM File Manager 플러그인이 필요합니다.
- 프리프로세서 포함 - 새 샘플 멤버가 추가되었습니다(버전 8.0.3.1부터).
 - FEKRNPLI
- Host Configuration Utility - 마이그레이션 옵션이 추가되었습니다(버전 8.0.2부터).
- JES 작업 모니터 - 새 운영자 명령이 JMON 시작 태스크에 추가되었습니다(버전 8.0.3.2부터).
 - MODIFY STORAGE
- JES 작업 모니터 - 새 선택적 지시문이 FEJJCNFG에 추가되었습니다(버전 8.0.3.1 및 8.0.3.2부터).
 - LIMIT_CONSOLE
 - SEARCHALL
 - TRACE_STORAGE
- PROCLIB - 다음 PROCLIB 멤버가 변경되었습니다(버전 8.0.3부터).
 - ELAXFUOP
- RSE - TMPDIR을 RSED 및 LOCKD 시작 태스크의 시작 인수로 지정하는 옵션이 제거되었습니다. 이 옵션은 /tmp를 쓰기 조치에 사용할 수 없는 경우 시작 태스크 사용자 ID의 홈 디렉토리를 TMPDIR으로 정의하는 사용자 정의가 불가능한 기능으로 대체됩니다(버전 8.0.3.1부터).
- RSE - 새 운영자 명령이 LOCKD 시작 태스크에 추가되었습니다(버전 8.0.2부터).
 - MODIFY DISPLAY TABLE
- RSE - 새 운영자 명령이 RSED 시작 태스크에 추가되었습니다(버전 8.0.2, 8.0.3 및 8.0.3.2부터).
 - MODIFY IVP ISPF,userid

- MODIFY IVP PASSTICKET,userid
- MODIFY DEBUG HEAPDUMP,PID=pid
- MODIFY DEBUG JAVACORE,PID=pid
- RSE - RSED 시작 태스크 운영자 명령이 개선되었습니다(버전 8.0.2 및 8.0.3.1부터).
 - MODIFY DISPLAY CLIENT[[,LOGON | ,ID | ,USER]]
 - MODIFY DISPLAY PROCESS,CPU[,PID=pid]
- RSE - 다음 콘솔 메시지가 새로 작성되었습니다(버전 8.0.3 및 8.0.3.1부터).
 - FEK910I = {0} IVP Exit code = {1}
 - FEK211W User, {0}, not logged on
- RSE - 사용자 정의가 불가능한 새 지시문이 rsed.envvars에 추가되었습니다(버전 8.0.3부터).
 - (_RSE_JAVAOPTS) -Dldap.server.address
 - (_RSE_JAVAOPTS) -Dldap.server.port
 - (_RSE_JAVAOPTS) -Dldap.ptc.group.name.suffix
 - _RSE_PTC
- RSE - 새 선택적 지시문이 rsed.envvars에 추가되었습니다(버전 8.0.3, 8.0.3.1 및 8.5부터).
 - (_RSE_JAVAOPTS) -Daudit.action
 - (_RSE_JAVAOPTS) -Daudit.action.id
 - (_RSE_JAVAOPTS) -Dlogon.action
 - (_RSE_JAVAOPTS) -Dlogon.action.id
 - (_RSE_JAVAOPTS) -Dreject.logon.threshold
 - (_RSE_JAVAOPTS) -Dinclude.c
 - (_RSE_JAVAOPTS) -Dinclude.cpp
 - (_RSE_JAVAOPTS) -DCPP_CLEANUP_INTERVAL
 - (_RSE_JAVAOPTS) -DRIS_BUFFER
 - (_RSE_JAVAOPTS) -DDSTORE_TCP_NO_DELAY
 - _RSE_FEK_SAF_CLASS
 - _RSE_LDAP_SERVER
 - _RSE_LDAP_PORT
 - _RSE_LDAP_PTC_GROUP_SUFFIX
 - CGI_ISPPREF
- RSE - 기존 필수 지시문의 이름이 변경되었습니다(버전 8.5부터).

- _CMDSESV_BASE_HOME -> CGI_ISPHOME
- _CMDSESV_CONF_HOME -> CGI_ISPCONF
- _CMDSESV_WORK_HOME -> CGI_ISPWORK
- _RSE_CMDSESV_OPTS -> _RSE_ISPF_OPTS
- RSE - 기존의 선택적 지시문이 추가 값으로 확장되었습니다(버전 8.5부터).
 - STEPLIB
- RSE - rsed.envvars의 다음 선택적 지시문의 해석이 변경되었습니다(버전 8.0.3부터).
 - (_RSE_JAVAOPTS) -Dprocess.cleanup.interval
- RSE - 다음 선택적 구성 파일이 새로 작성되었습니다(버전 8.5부터).
 - include.conf
- RSE - 새 선택적 지시문이 pushtoclient.properties에 추가되었습니다(버전 8.0.3부터).
 - accept.product.license
- RSE - pushtoclient.properties의 다음 선택적 지시문의 해석이 변경되었습니다(버전 8.0.3부터).
 - config.enabled
 - product.enabled
 - reject.config.updates
 - reject.product.updates
- RSE - 새 z/OS UNIX 샘플이 추가되었습니다(버전 8.0.3 및 8.0.3.1부터).
 - process_audit.rex
 - process_logon.sh
- 보안 - 새 보안 프로파일에 대한 지원이 추가되었습니다(버전 8.0.3부터).
 - FEK.PTC.**
- zUnit - 새 선택적 PROCLIB 멤버가 추가되었습니다(버전 8.5부터).
 - AZUZUNIT
- 새 서적, *Rational Developer for System z 메시지 및 코드*(SA30-4577).
- 새 서적, *Rational Developer for System z Answers to common host configuration and maintenance issues*(SA30-4502).

구성 가능한 파일

205 페이지의 표 25에서는 버전 8.5에서 사용자 정의되는 파일 개요를 표시합니다. Developer for System z 샘플 라이브러리(FEK.SFEKSAMP, FEK.SFEKSAMV 및 /usr/lpp/rdz/samples/)에는 컴파일하기 위한 샘플 CARMA 소스 코드 및 작업과 같이 여기에 나열된 멤버 이상의 사용자 정의 가능한 멤버가 포함됩니다.

다음 멤버와 파일은 더 이상 사용자 정의하거나 사용될 수 없습니다.

- FMIEXT.properties는 더 이상 사용되지 않음

참고: 샘플 작업 FEKSETUP은 표시된 모든 멤버를 다른 데이터 세트 및 디렉토리로 복사하고 기본값은 FEK.#CUST.* 및 /etc/rdz/*입니다.

표 25. 버전 8.5 사용자 정의

멤버/파일	기본 위치	목적	마이그레이션 참고
FEKSETUP	FEK.SFEKSAMP [FEK.#CUST.JCL]	데이터 세트 및 디렉토리를 작성하고 이를 사용자 정의 가능한 파일로 채우는 JCL	사용자 정의 가능한 새 멤버를 포함하고, 새 디렉토리 구조를 작성하고, 더 이상 사용하지 않는 파일에 대한 조치는 제거하도록 업데이트합니다.
JMON	FEK.SFEKSAMP(FEJJJCL) [FEK.#CUST.PROCLIB]	JES 작업 모니터의 JCL	없음
FEJJJCL	FEK.SFEKSAMP [FEK.#CUST.PROCLIB(JMON)]	JMON 멤버의 이름	JMON 멤버 참조
RSED	FEK.SFEKSAMP(FEKRSED) [FEK.#CUST.PROCLIB]	RSE 디먼의 JCL	변경된 TMPDIR 지원
FEKRSED	FEK.SFEKSAMP [FEK.#CUST.PROCLIB(RSED)]	RSED 멤버의 이름	RSED 멤버 참조
LOCKD	FEK.SFEKSAMP(FEKLOCKD) [FEK.#CUST.PROCLIB]	잠금 디먼의 JCL	변경된 TMPDIR 지원
FEKLOCKD	FEK.SFEKSAMP [FEK.#CUST.PROCLIB(LOCKD)]	LOCKD 멤버의 이름	LOCKD 멤버 참조
ELAXF*	FEK.SFEKSAMP [FEK.#CUST.PROCLIB]	원격 프로젝트 빌드 등을 위한 JCL	ELAXFUOP 멤버가 변경됨
FEKRACF	FEK.SFEKSAMP [FEK.#CUST.JCL]	보안 정의를 위한 JCL	없음
FEJJCNFG	FEK.SFEKSAMP [FEK.#CUST.PARMLIB]	JES 작업 모니터 구성 파일	선택적 새 지시문이 추가됨
FEJTSO	FEK.SFEKSAMP [FEK.#CUST.CNTL]	TSO 제출을 위한 JCL	없음
CRA\$VMSG	FEK.SFEKSAMP [FEK.#CUST.JCL]	CARMA 메시지 VSAM을 작성하기 위한 JCL	VSAM 입력이 변경됨
CRA\$VDEF	FEK.SFEKSAMP [FEK.#CUST.JCL]	CARMA 구성 VSAM을 작성하기 위한 JCL	없음

표 25. 버전 8.5 사용자 정의 (계속)

멤버/파일	기본 위치	목적	마이그레이션 참고
CRA\$VSTR	FEK.SFEKSAMP [FEK.#CUST.JCL]	CARMA 사용자 정의 정보 VSAM을 작성하기 위한 JCL	없음
CRA\$VCAD	FEK.SFEKSAMP [FEK.#CUST.JCL]	CA Endeavor® SCM RAM에 대한 CARMA 구성 VSAM 작성하는 JCL	VSAM 입력이 변경됨
CRA\$VCAS	FEK.SFEKSAMP [FEK.#CUST.JCL]	CA Endeavor® SCM RAM에 대한 CARMA 사용자 정의 정보 VSAM을 작성하는 JCL	VSAM 입력이 변경됨
CRASUBMT	FEK.SFEKSAMP [FEK.#CUST.CNTL]	CARMA 일괄처리 시작 CLIST	없음
CRASUBCA	FEK.SFEKSAMP [FEK.#CUST.CNTL]	CA Endeavor® SCM RAM을 위한 CARMA 일괄처리 시작 CLIST	추가로 DD 문이 추가됨
CRABCFG	FEK.SFEKSAMP [FEK.#CUST.PARMLIB]	CA Endeavor® SCM RAM의 CARMA 일괄처리 조치 구성	신규, 사용자 정의는 선택사항임
CRABATCA	FEK.SFEKSAMP [FEK.#CUST.CNTL]	CA Endeavor® SCM RAM의 CARMA 일괄처리 조치 JCL	신규, 사용자 정의는 선택사항임
CRASHOW	FEK.SFEKSAMP [FEK.#CUST.PARMLIB]	CA Endeavor® SCM RAM의 CARMA 구성	새 필터가 추가됨
CRATMAP	FEK.SFEKSAMP [FEK.#CUST.PARMLIB]	CA Endeavor® SCM RAM의 CARMA 구성	없음
CRANDVRA	FEK.SFEKPROC	CA Endeavor® SCM RAM의 CARMA 할당 REXX	추가로 DD 문이 추가됨
CRA#VSLM	FEK.SFEKSAMP [FEK.#CUST.JCL]	SCLM RAM 메시지 VSAM을 작성하는 JCL	없음
CRA#ASLM	FEK.SFEKSAMP [FEK.#CUST.JCL]	SCLM RAM 데이터 세트를 작성하기 위한 JCL	없음
CRA#VPDS	FEK.SFEKSAMP [FEK.#CUST.JCL]	PDS RAM 메시지 VSAM을 작성하는 JCL	없음
CRA#UADD	FEK.SFEKSAMP [FEK.#CUST.JCL]	RAM 정의를 병합하는 JCL	없음
CRA#UQRY	FEK.SFEKSAMP [FEK.#CUST.JCL]	RAM 정의를 추출하는 JCL	없음
CRAXJCL	FEK.SFEKSAMP [FEK.#CUST.ASM]	IRXJCL 대체를 위한 샘플 소스 코드	없음

표 25. 버전 8.5 사용자 정의 (계속)

멤버/파일	기본 위치	목적	마이그레이션 참고
CRA#CIRX	FEK.SFEKSAMP [FEK.#CUST.JCL]	CRAXJCL을 컴파일하는 JCL	없음
ADNCSDRS	FEK.SFEKSAMP [FEK.#CUST.JCL]	기본 CICS 리전에 RESTdul CRD 서버를 정의하는 JCL	없음
ADNCSDTX	FEK.SFEKSAMP [FEK.#CUST.JCL]	CICS 리전에 대체 트랜잭션 ID 를 정의하는 JCL	없음
ADNTXNC	FEK.SFEKSAMP [FEK.#CUST.JCL]	대체 트랜잭션 ID를 작성하는 JCL	없음
ADNMSGHC	FEK.SFEKSAMP [FEK.#CUST.JCL]	ADNMSGHS를 컴파일하는 JCL	없음
ADNMSGHS	FEK.SFEKSAMP [FEK.#CUST.COBOL]	파이프라인 메시지 핸들러의 샘플 소스 코드	없음
ADNVCRD	FEK.SFEKSAMP [FEK.#CUST.JCL]	CRD 저장소를 작성하는 JCL	없음
ADNCSDWS	FEK.SFEKSAMP [FEK.#CUST.JCL]	기본 CICS 리전에 웹 서비스 CRD 서버를 정의하는 JCL	없음
ADNCSDAR	FEK.SFEKSAMP [FEK.#CUST.JCL]	비기본 CICS 리전에 CRD 서버 를 정의하는 JCL	없음
ADNJSPAU	FEK.SFEKSAMP [FEK.#CUST.JCL]	CRD 기본값을 업데이트하는 JCL	없음
ADNMFST	FEK.SFEKSAMP [FEK.#CUST.JCL]	Manifest 저장소를 작성 및 정의 하는 JCL	없음
ELAXMSAM	FEK.SFEKSAMP [FEK.#CUST.PROCLIB]	WLM 주소 공간의 JCL 프로시저	없음
ELAXMJCL	FEK.SFEKSAMP [FEK.#CUST.JCL]	DB2에 PL/I 및 COBOL 스토어 드 프로시저 빌더를 정의하는 JCL	없음
AZUZUNIT	FEK.SFEKSAMP [FEK.#CUST.PROCLIB]	zUnit에 대한 JCL 프로시저	신규, 사용자 정의는 선택사항임
FEKRNPLI	FEK.SFEKSAMP [FEK.#CUST.CNTL]	프리프로세서 프레임워크 내에서 PL/I 컴파일러를 호출하기 위한 Rexx	신규, 사용자 정의는 선택사항임

표 25. 버전 8.5 사용자 정의 (계속)

멤버/파일	기본 위치	목적	마이그레이션 참고
FEKLOGS	FEK.SFEKSAMP [FEK.#CUST.JCL]	로그 파일을 수집하기 위한 JCL	추가 검사가 추가되었습니다. 이전 파일에 대한 사용자 정의도 다시 수행되어야 합니다.
rse.envvars	/usr/lpp/rdz/samples/ [/etc/rdz/]	RSE 환경 변수	이전 사본은 이 사본으로 대체되고 사용자 정의가 다시 수행되어야 합니다.
ISPF.conf	/usr/lpp/rdz/samples/ [/etc/rdz/]	TSO/ISPF Client Gateway 구성 파일	없음
CRASRV.properties	/usr/lpp/rdz/samples/ [/etc/rdz/]	CARMA 구성 파일	임시 포트를 위해 추가된 지원
crastart.conf	/usr/lpp/rdz/samples/ [/etc/rdz/]	CRASTART 사용법을 위한 CARMA 구성 파일	없음
crastart.endevor.conf	/usr/lpp/rdz/samples/ [/etc/rdz/]	CA Endevor® SCM RAM에 대한 CRASTART 사용의 CARMA 구성 파일	추가로 DD 문이 추가됨
include.conf	/usr/lpp/rdz/samples/ [/etc/rdz/]	C/C++ 콘텐츠 지원을 위해 강제로 포함됨	신규, 사용자 정의는 선택사항임
ssl.properties	/usr/lpp/rdz/samples/ [/etc/rdz/]	RSE SSL 구성 파일	없음
rsecomm.properties	/usr/lpp/rdz/samples/ [/etc/rdz/]	RSE 추적 구성 파일	일부 지시문은 선택사항이 됨
pushtoclient.properties	/usr/lpp/rdz/samples/ [/etc/rdz/]	클라이언트 구성 파일에 정보를 푸시함	추가로 지시문이 추가되고 기존 지시문은 개선됨

제 11 장 운영자 명령

이 장에서는 Developer for System z에서 사용 가능한 운영자(또는 콘솔) 명령에 대한 개요를 제공합니다.

Start(S)

시작 태스크(STC)를 동적으로 시작하려면 **START** 명령을 사용하십시오. 명령의 축약 버전은 문자 S입니다.

통합 디버거

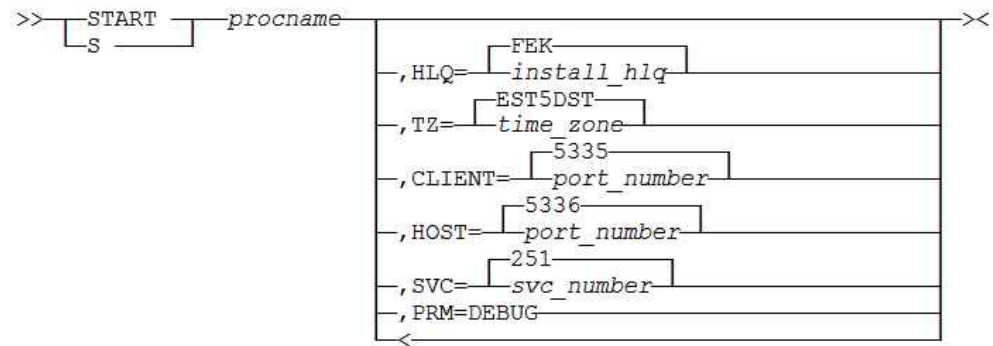


그림 37. START DBGMGR 운영자 명령

procname

프로시저 라이브러리에서 서버를 시작하는 데 사용되는 멤버의 이름입니다. 호스트 시스템 구성 중에 사용되는 기본 이름은 DBGMGR입니다.

HLQ=install_hlq

Developer for System z를 설치하는 데 사용되는 상위 레벨 규정자입니다. 기본값은 FEK입니다.

TZ=time_zone

시간대 오프셋. 기본값은 EST5DST입니다.

CLIENT=port_number

외부(클라이언트-호스트) 통신에 사용하는 포트(기본값 5335)입니다.

HOST=port_number

내부(호스트 한정) 통신에 사용하는 포트(기본값 5336)입니다.

| **SVC=svc_number**
 | 읽기 전용 CICS 트랜잭션을 디버깅하는 데 사용하는 SVC 번호(기본값: 251)
 | 입니다.
 | **PRM=DEBUG**
 | verbose(추적) 모드를 사용합니다. 추적은 성능 저하를 초래하므로 IBM 지원
 | 센터의 지시 하에서만 수행해야 합니다.

JES 작업 모니터

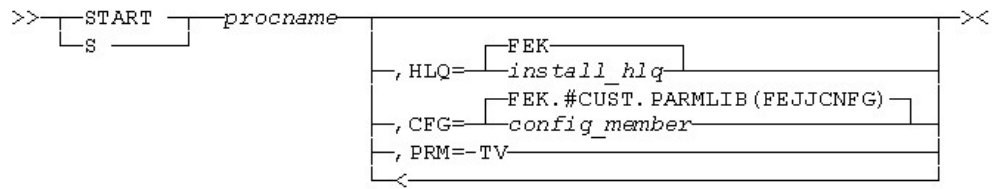


그림 38. *START JMON* 운영자 명령

procname

프로시저 라이브러리에서 서버를 시작하는 데 사용되는 멤버의 이름입니다. 호스트 시스템 구성 중에 사용되는 기본 이름은 JMON입니다.

HLQ=install_hlq

Developer for System z를 설치하는 데 사용되는 상위 레벨 규정자입니다. 기본값은 FEK입니다.

CFG=config_member

JES 작업 모니터 구성 파일의 절대 데이터 세트와 멤버명입니다. 기본값은 FEK.#CUST.PARMLIB(FEJJCNFG)입니다. 이 변수가 NULLFILE로 설정되면, JES 작업 모니터는 기본 구성 값을 사용합니다.

PRM=-TV

verbose(추적) 모드를 사용합니다. 추적은 성능 저하를 초래하므로 IBM 지원 센터의 지시 하에서만 수행해야 합니다.

RSE 디먼

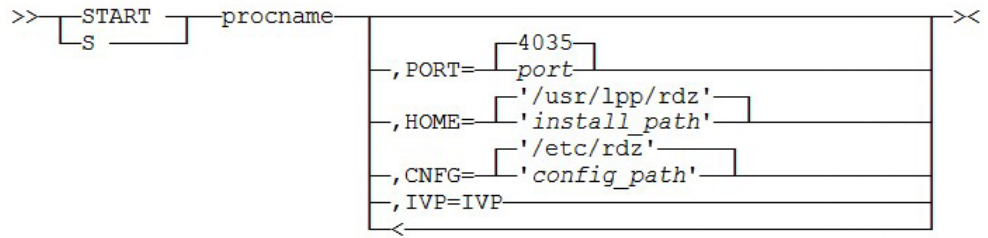


그림 39. START RSED 운영자 명령

procname

프로시저 라이브러리에서 서버를 시작하는 데 사용되는 멤버의 이름입니다. 호스트 시스템 구성 중에 사용되는 기본 이름은 RSED입니다.

PORT=port

클라이언트에 연결하기 위해 RSE 디먼에서 사용하는 포트입니다. 지정되지 않은 경우, 변수 `_RSE_RSED_PORT`에서 `/etc/rdz/rsed.envvars`에서 정의된 포트가 사용됩니다. 기본값은 4035입니다.

IVP=IVP

서버를 시작하지 말고 RSE 디먼 설치 검증 프로그램(IVP)를 운영하십시오.

CNFG='config_path'

z/OS UNIX에 저장된 구성 파일의 절대 위치입니다. 기본값은 `'/etc/rdz'`입니다. z/OS UNIX 경로는 대소문자를 구분하며, 소문자를 그대로 유지하려면 작은따옴표(')로 묶어야 합니다.

HOME='install_path'

Developer for System z를 설치하는 데 사용되는 경로 접두부 및 필수 `/usr/lpp/rdz`입니다. 기본값은 `'/usr/lpp/rdz'`입니다. z/OS UNIX 경로는 대소문자를 구분하며, 소문자를 그대로 유지하려면 작은따옴표(')로 묶어야 합니다.

Modify(F)

MODIFY 명령을 사용하면 활성 태스크의 특성을 동적으로 조회하고 변경할 수 있습니다. 이 명령의 축약 버전은 문자 F입니다.

통합 디버거

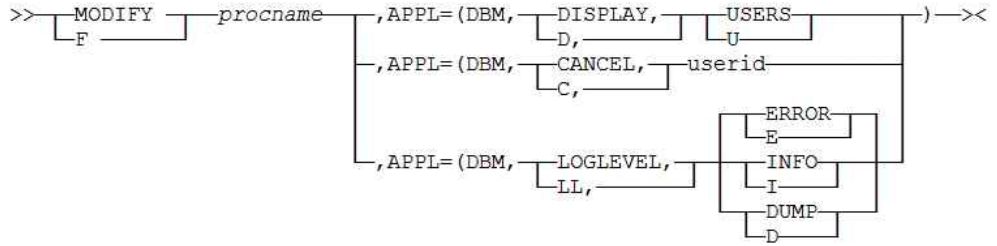


그림 40. MODIFY DBGMGR 운영자 명령

procname

프로시저 라이브러리에서 서버를 시작하는 데 사용되는 멤버의 이름입니다. 호스트 시스템 구성 중에 사용되는 기본 이름은 DBGMGR입니다.

DISPLAY,USERS

AQECM104I 콘솔 메시지와 함께 활성 사용자를 표시합니다. 메시지 AQECM103I는 활성 사용자가 없는 경우에 발행됩니다. 사용자 목록에는 서버에 있는 해당 사용자의 상태가 표시됩니다. 통합 디버거 데이터 플로우의 개요에 대해서는 호스트 구성 참조서(SA30-4501)의 "Developer for System z 이해" 장의 "통합 디버거" 절을 참조하십시오.

```
AQECM104I User:IBMUSER RegisterSocket(2) waits for probe
connection
AQECM104I User:IBMUSR2 ProbeSocket(3) waits for register
connection
AQECM104I User:IBMUSR3 EngineSocket(4) connected to
ProbeSocket(8)
AQECM104I User:IBMUSR4 ProbeSocket(5) waits for engine
connection
AQECM103I There is no active user
```

CANCEL,userid

지정된 사용자 ID에 대한 모든 디버그 세션을 취소합니다. 결과는 AQECM110I 또는 AQECM111I 콘솔 메시지와 함께 표시됩니다.

```
AQECM110I user(IBMUSER) canceled
AQECM111I user(IBMUSER) not connected
```

LOGLEVEL,{ERROR | INFO | DUMP}

디버거 관리자 메시지 로그(DD SYSPRINT)의 상세 레벨을 제어합니다. 기본 값은 E(오류)입니다. 메시지 "LOGLEVEL command processed normally"는 메시지 ID가 AQECM101I인 콘솔에 작성됩니다.

E 또는 ERROR	오류 메시지 전용(기본값)
I 또는 INFO	오류 및 정보 메시지
D 또는 DUMP	오류, 정보 및 디버거/덤프 메시지

자세한 추적은 성능 저하를 초래하므로 IBM 지원 센터의 지시 하에서만 수행해야 합니다.

JES 작업 모니터

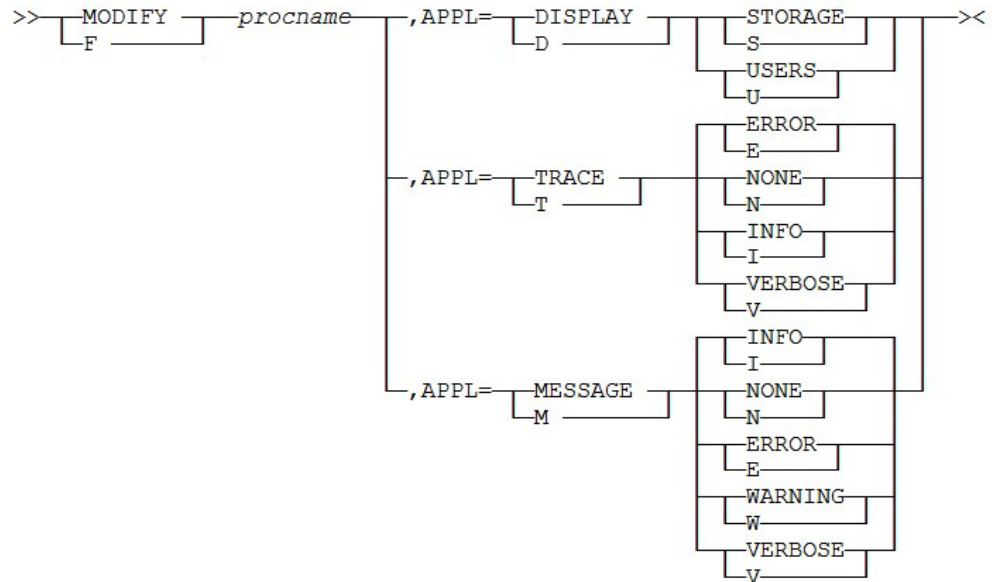


그림 41. MODIFY JMON 운영자 명령

procname

서버를 시작하는 데 사용된 프로시저 라이브러리의 멤버 이름. 호스트 시스템 구성 중에 사용되는 기본 이름은 JMON입니다.

DISPLAY STORAGE

DD SYSOUT에 스토리지 사용량 보고서를 작성합니다. 메시지 "JMON storage information written to SYSOUT"은 메시지 ID BPXM023I를 사용하여 콘솔에 작성됩니다. 스토리지 사용량 보고서는 바이트, KB 및 MB 단위로 여러 스토리지 관련 필드를 표시합니다.

```

>>>STORAGE TRACE (console request)<<<
LDAREGRQ  00000000000 00000000K 00000M requested region size
          below 16M line
LDASIZA   00006266880 00006120K 00005M maximum region size
LDALIMIT  00006266880 00006120K 00005M limit
LDAVVRG   00006266880 00006120K 00005M getmain limit
LDALOAL   00000061440 00000060K 00000M in use
LDAHIAL   00000266240 00000260K 00000M LSQA/SWA/private subpools
_GAP      00000000000 00000000K 00000M gaps in allocation
_Avail    00005939200 00005800K 00005M available (including gaps)
_MAX      00006000640 00005860K 00005M current limit
          above 16M line
LDAESIZA  01905262592 01860608K 01817M maximum region size
LDAELIM   01905262592 01860608K 01817M limit
LDAEVVRG  01905262592 01860608K 01817M getmain limit
LDAELOAL  00000937984 00000916K 00000M in use
LDAEHIAL  00012754944 00012456K 00012M ELSQA/ESWA/private subpools
  
```

```

_EGAP      00000000000 000000000K 00000M gaps in allocation
_EAVAIL    01891569664 01847236K 01803M available (including gaps)
_EMAX      01892507648 01848152K 01804M current limit

```

DISPLAY USERS

DD SYSOUT에 활성 사용자 목록을 작성합니다. 메시지 "JMON user list written to SYSOUT"은 메시지 ID BPXM023I를 사용하여 콘솔에 작성됩니다. 사용자 리스트는 CPU 사용량을 포함하여, 다양한 사용자 관련 데이터를 보여줍니다.

```

S0  userid    USER      4:04(elapsed)    4:04(idle)
Users: 1

```

TRACE {NONE | ERROR | INFO | VERBOSE}

JES 작업 모니터 추적 로그(DD SYSOUT)의 상세 레벨을 제어합니다. 기본 값은 E(오류)입니다. 메시지 "JMON TRACE LEVEL:{NONE | ERROR | INFO | VERBOSE}"는 메시지 ID BPXM023I를 사용하여 콘솔에 작성됩니다.

N 또는 NONE	시작 메시지만
E 또는 ERROR	시작 또는 오류 메시지만(기본)
I 또는 INFO	시작, 오류 및 정보 메시지
V 또는 VERBOSE	시작, 오류, 정보 및 자동 메시지

자세한 추적은 성능 저하를 초래하므로 IBM 지원 센터의 지시 하에서만 수행해야 합니다.

MESSAGE {NONE | ERROR | WARNING | INFO | VERBOSE}

JES 작업 모니터 메시지 로그(DD SYSPRINT)의 상세 레벨을 제어합니다. 기본 값은 I(정보)입니다. 메시지 "JMON MESSAGE LEVEL:{NONE | ERROR | WARNING | INFO | VERBOSE}"는 메시지 ID BPXM023I를 사용하여 콘솔에 작성됩니다.

N 또는 NONE	메시지가 없습니다.
E 또는 ERROR	오류 메시지만
W 또는 WARNING	오류 및 경고 메시지
I 또는 INFO	오류, 경고 및 정보 메시지(기본)
V 또는 VERBOSE	오류 경고, 정보 및 자동 메시지

자세한 추적은 성능 저하를 초래하므로 IBM 지원 센터의 지시 하에서만 수행해야 합니다.

RSE 디맨

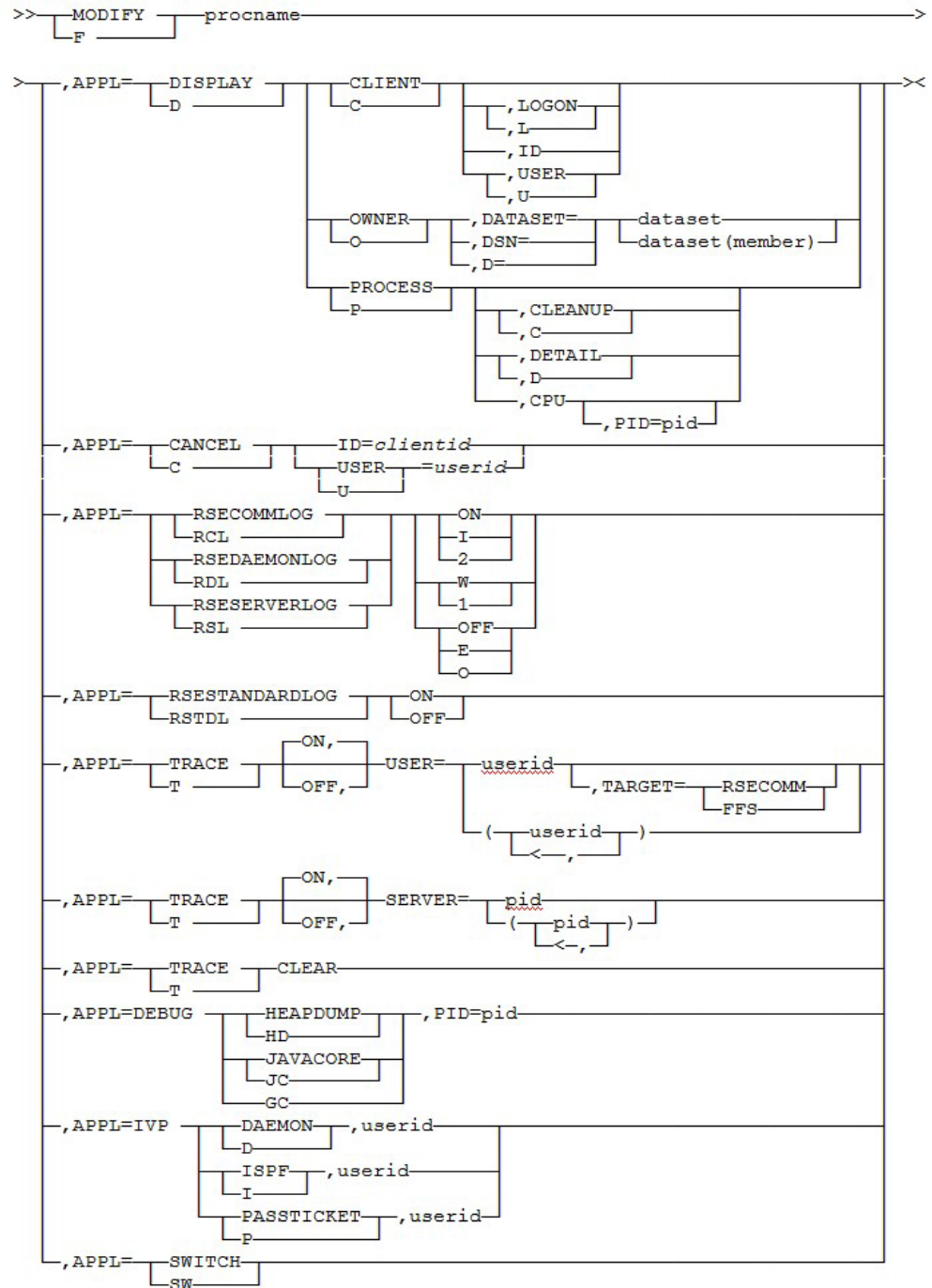


그림 42. MODIFY RSED 운영자 명령

procname

서버를 시작하는 데 사용된 프로시저 라이브러리의 멤버 이름. 호스트 시스템 구성 중에 사용되는 기본 이름은 RSED입니다.

DISPLAY CLIENT[{LOGON | ,ID | ,USER}]

단일 BPXM023I 메시지에 활성 클라이언트를 표시합니다. 결과 레이아웃은 사용한 명령 옵션에 따라 다릅니다. 선택적 명령 인수로 정렬 순서를 변경할 수 있습니다.

- No 명령 옵션: 사용되는 스레드 풀 프로세스별로 클라이언트가 그룹화됩니다.

```
ProcessId(<processid>) ASId(<asid>) JobName(<jobname>)
Clients(<local>/<total>) Order(<startup order>)
<clientid><userid><connected since>
```

- LOGON 명령 옵션: 로그인 시간별로 클라이언트가 정렬됩니다.

```
LOGON TIME----- ID----- USERID--
<connected since>      <clientid> <userid>
```

- ID 명령 옵션: 클라이언트 ID별로 클라이언트가 정렬됩니다.

```
ID----- USERID-- LOGON TIME-----
<clientid> <userid> <connected since>
```

- USER 명령 옵션: 사용자 ID별로 클라이언트별로 정렬됩니다.

```
USERID-- ID----- LOGON TIME-----
<userid> <clientid> <connected since>
```

DISPLAY OWNER,DATASET={dataset | dataset(member)}

한 개의 BPXM023I 메시지에서 데이터 세트 인큐 소유자를 디스플레이합니다.

```
FEK217I <dataset[(member)]> is locked by <userid>
FEK218I <dataset[(member)]> is not locked
FEK219E Failed to determine lock owner for <dataset[(member)]>
```

- 서버는 또한 ISPF 같은 다른 제품이 보유한 잠금을 보고합니다.
- **D GRS,RES=(*,dataset)** 운영자 명령은 Developer for System z 사용자가 실제 인큐 홀더임을 표시할 수 없습니다. 표시할 수 있는 것은 사용자가 활성화되어 있는 스레드 풀입니다.

DISPLAY PROCESS[{CLEANUP | ,CPU [,PID=pid] | ,DETAIL}]

하나 이상의 BPXM023I 메시지에 RSE 스레드 풀 프로세스를 표시합니다. 연결된 사용자의 로드 밸런싱에 사용되는 여러 프로세스가 있을 수 있습니다.

```
ProcessId(<processid>) Memory Usage(<java heap usage>%)
Clients(<number of clients>) Order(<startup order>) <error status>
```

참고:

- 프로세스 고유 z/OS UNIX 운영자 명령에서 <processid>를 사용할 수 있습니다.
- 각 프로세스는 자체적으로 Java 힙을 가지며, rsed.envvars에서 힙 크기를 설정할 수 있습니다. 보고된 Java 힙 사용량에는 Developer for System z에서 릴리스하였으나 Java의 가비지 콜렉션 프로세스에서 차지하고 있는 공간을 바꾸지 않은 스토리지가 포함됩니다.

- <startup order>는 스레드 풀이 시작된 순서를 표시하는 순차 번호입니다. 숫자는 stderr.*.log 및 stdout.*.log 파일의 파일 이름에 사용된 숫자와 일치합니다.

일반적인 경우, <error status>는 공백입니다. 표 26에서 <error status>에 사용 가능한 (공백이 아닌) 값을 설명합니다.

표 26. 스레드 풀 오류 상태

상태	설명
severe error	스레드 풀 프로세스가 복구 불가능한 오류를 발견하여 연산을 중지했습니다. 나머지 상태 필드에서는 마지막으로 알려진 값을 표시합니다. 테이블에서 이 항목을 제거하려면 DISPLAY PROCESS modify 명령의 CLEANUP 옵션을 사용하십시오.
killed process	스레드 풀 프로세스가 Java, z/OS UNIX에 의해 또는 운영자 명령으로 강제 종료되었습니다. 나머지 상태 필드에서는 마지막으로 알려진 값을 표시합니다. 테이블에서 이 항목을 제거하려면 DISPLAY PROCESS modify 명령의 CLEANUP 옵션을 사용하십시오.
timeout	클라이언트 연결 요청 시 스레드 풀 프로세스가 적절한 시간 내에 RSE 디먼에 응답하지 않았습니다. 나머지 상태 필드에서는 현재 값을 표시합니다. 나중의 클라이언트 연결 요청을 위해 스레드 풀이 재외됩니다. 이 스레드 풀이 지원하는 클라이언트가 로그오프하면 *timeout* 상태가 다시 설정됩니다.

DISPLAY PROCESS modify 명령의 **DETAIL** 옵션을 사용하면 자세한 정보가 제공됩니다.

```
ProcessId(33555087) ASId(002E) JobName(RSED8) Order(1)
PROCESS LIMITS:    CURRENT  HIGHWATER    LIMIT
  JAVA HEAP USAGE(%)    10         56         100
    CLIENTS              0          25          30
  MAXFILEPROC          83        103       64000
  MAXPROCUSER          97         99         200
  MAXTHREADS           9         14        1500
  MAXTHREADTASKS       9         14        1500
```

ASId 필드는 16진 표기법의 주소 공간 ID입니다. 프로세스 한계값 테이블에서는 현재 자원 사용량, 자원 사용량 최고치 및 자원 한계값을 표시합니다. 다른 제한 요소로 인해 정의된 한계값에 결코 도달하지 않을 수도 있습니다.

DISPLAY PROCESS modify 명령의 **CPU** 옵션은 스레드 풀에 있는 각 스레드의 누적 CPU 사용량(밀리초)을 보여줍니다. 각 스레드 풀에는 하나의 BPXM023I 메시지가 있습니다. 기본적으로 모든 스레드 풀이 CPU 사용량을 보고하지만 운영자 명령에서 PID=pid를 지정하여 단일 스레드 풀에 대한 범위를 제한할 수 있습니다. 여기서 pid는 대상 스레드 풀의 프로세스 ID입니다.


```

ProcessId(421      ) ASId(007D) JobName(RSED8) Order(1)
USERID  THREAD-ID      TCB@      ACC_TIME TAG
STCRSE  0EDE54000000000 005E6B60      822 1/ThreadPoolProcess
STCRSE  0EDE870000000001 005E69C8        001
STCRSE  0EDE980000000002 005E6518      1814
STCRSE  0EDEBA0000000003 005E66B0      2305
STCRSE  0EDECB0000000004 005E62F8        001
STCRSE  0EDED00000000005 005E60D8        001
STCRSE  0EDF860000000006 005C2BF8        628 6/ThreadPoolMonitor$Memory
UsageMonitor
STCRSE  0EDF970000000007 005C2D90        003 7/ThreadPoolMonitor
STCRSE  0EDFDB0000000008 005C29D8        001
STCRSE  0EE22E000000000E 005C1BE0        070
IBMUSER 0EE0EB00000000011 005C22B8      276 20/ServerReceiver
IBMUSER 0EE25000000000012 005C19C0      137 16/ServerUpdateHandler
IBMUSER 0EE26100000000013 005C17A0      509 15/ServerCommandHandler
IBMUSER 0EE18400000000014 005C1E00      065 21/ZosSystemMiner
STCRSE  0EE15100000000016 005C2098        078
STCRSE  0EE19500000000017 005C1580        001
IBMUSER 0EE23F00000000018 005C1360      021 26/UniversalFileSystemMiner
r
IBMUSER 0EE2A50000000001C 005C0CF0        003 27/EnvironmentMiner
IBMUSER 0EE2830000000001D 005C1140        002 31/CommandMiner
IBMUSER 0EE2720000000001E 005C0E88      081 32/MVSFileSystemMiner
IBMUSER 0EE2940000000001F 005C0AD0      002 33/MVSByteStreamHandler$Op
enCloseThread
STCRSE  0EE2E900000000023 005C0470        001
IBMUSER 0EE2C700000000024 005C08B0      050 38/JESMiner
IBMUSER 0EE2B600000000026 005C0690      004 40/FAMiner
IBMUSER 0EE30B00000000027 005C0250      002 41/LuceneMiner
IBMUSER 0EE31C00000000028 005C0030      002 42/CDTParserMiner
IBMUSER 0EE32D00000000029 005BDE00      002 43/MVSLuceneMiner
IBMUSER 0EE33E0000000002A 005BDBE0      002 44/CDTMVSParserMiner

```

출력 크기가 콘솔 메시지당 허용되는 최대 행 수를 초과하면 출력이 여러 개의 BPXM023I 메시지에 분산됩니다. 이러한 추가 메시지에는 첫 번째 메시지와 동일한 머리글이 표시되며 첫 번째 행에 CONTINUATION 키워드가 추가됩니다.

```

ProcessId(421      ) ASId(007D) JobName(RSED8) Order(1) CONTINUATION
USERID  THREAD-ID      TCB@      ACC_TIME TAG

```

각 스레드 풀당 처음 4000개의 스레드로 결과물이 제한됩니다.

CANCEL ID=clientid

DISPLAY CLIENT modify 명령에 표시되는 클라이언트 ID에 기초하여 클라이언트 연결을 취소합니다.

클라이언트 연결이 취소되면 호스트 시스템 스레드가 일반 종료 처리를 거쳐 사용한 자원을 정리합니다. 이 조치는 일부 스레드가 종료하기 까지 약간의 시간이 소요될 수 있음을 의미합니다. 예를 들어, 제한시간까지 연결 유지 메커니즘을 유지하기 때문입니다.

CANCEL USER=userid

DISPLAY CLIENT modify 명령에 표시된 클라이언트의 사용자 ID에 따라 클라이언트 연결을 취소합니다.

클라이언트 연결이 취소되면 호스트 시스템 스레드가 일반 종료 처리를 거쳐 사용한 자원을 정리합니다. 이 조치는 일부 스레드가 종료하기 까지 약간의 시간이 소요될 수 있음을 의미합니다. 예를 들어, 제한시간까지 연결 유지 메커니즘을 유지하기 때문입니다.

RSECOMMLOG {ON | OFF | I | W | E | 2 | 1 | 0}

RSE 서버(rsecomm.log) 및 MVS 데이터 세트 서비스(lock.log 및 ffs*.log)의 추적 세부사항 레벨을 제어합니다. rsecomm.properties에 시작 기본값이 정의되어 있습니다. 세 가지 세부사항 레벨을 사용할 수 있습니다.

E 또는 0 또는 OFF	오류 메시지 전용.
W 또는 1	오류 및 경고 메시지. rsecomm.properties에서는 이것이 기본 설정입니다.
I 또는 2 또는 ON	오류, 경고 및 정보 메시지.

자세한 추적은 성능 저하를 초래하므로 IBM 지원 센터의 지시 하에서만 수행해야 합니다.

RSEDAEMONLOG {ON | OFF | I | W | E | 2 | 1 | 0}

RSE 디몬(rsedaemon.log)의 추적 세부사항 레벨을 제어합니다. rsecomm.properties에 시작 기본값이 정의되어 있습니다. 세 가지 세부사항 레벨을 사용할 수 있습니다.

E 또는 0 또는 OFF	오류 메시지 전용.
W 또는 1	오류 및 경고 메시지. rsecomm.properties에서는 이것이 기본 설정입니다.
I 또는 2 또는 ON	오류, 경고 및 정보 메시지.

자세한 추적은 성능 저하를 초래하므로 IBM 지원 센터의 지시 하에서만 수행해야 합니다.

RSESERVERLOG {ON | OFF | I | W | E | 2 | 1 | 0}

RSE 스레드 풀(rseserver.log)의 추적 세부사항 레벨을 제어합니다. rsecomm.properties에 시작 기본값이 정의되어 있습니다. 세 가지 세부사항 레벨을 사용할 수 있습니다.

E 또는 0 또는 OFF	오류 메시지 전용.
W 또는 1	오류 및 경고 메시지. rsecomm.properties에서는 이것이 기본 설정입니다.
I 또는 2 또는 ON	오류, 경고 및 정보 메시지.

자세한 추적은 성능 저하를 초래하므로 IBM 지원 센터의 지시 하에서만 수행해야 합니다.

RSESTANDARDLOG {ON | OFF}

stdout*.log 및 stderr*.log 스레드 풀의 stdout 및 stderr 스트림이 있

는 로그 파일 업데이트를 사용하거나(ON) 사용하지 않습니다(OFF). 시작 기본 값은 rsed.envvars에 enable.standard.log 지시문으로 정의됩니다.

자세한 추적은 성능 저하를 초래하므로 IBM 지원 센터의 지시 하에서만 수행 해야 합니다.

TRACE [{ON, | OFF,}]USER=userid[,TARGET={FFS | RSECOMM}]

지정된 사용자 ID에 대한 추적을 사용하거나(ON) 사용하지 않습니다(OFF). 기본값은 ON입니다. 이 설정은 **MODIFY RSECOMMLOG** 운영자 명령에 의해 제어되는 기본 설정을 무효화합니다. 두 가지 세부사항 레벨을 사용할 수 있습니다.

OFF	오류 메시지만
ON(기본값)	오류, 경고 및 정보 메시지.

명령이 RSE 서버(rsecomm.log) 및 MVS 데이터 세트 서비스(lock.log 및 ffs*.log)의 추적 세부사항 레벨을 변경합니다. 이는 두 값을 허용하는 TARGET 키워드를 사용하여 제한할 수 있습니다.

FFS	MVS 데이터 세트 서비스에 대해서만 지정된 로그 레벨 설정 (lock.log 및 ffs*.log)
RSECOMM	RSE 서버에 대해서만 지정된 로그 레벨 설정(rsecomm.log)

현재 로그인하지 않은 사용자에게 대해서 명령을 실행할 수 있습니다. 사용자가 로그오프할 때 설정은 활성 상태로 유지되고 사용자가 로그인할 때 다시 사용 됩니다.

서버가 시작될 때 **MODIFY TRACE USER** 명령을 실행하는 것을 시뮬레이션하려면 rsecomm.properties의 USER 지시문을 사용하십시오. 이전 **MODIFY TRACE USER** 또는 **MODIFY TRACE SERVER** 운영자 명령 또는 rsecomm.properties의 USER 지시문의 기존 설정이 이 명령의 설정으로 대체됩니다.

자세한 추적은 성능 저하를 초래하므로 IBM 지원 센터의 지시 하에서만 수행 해야 합니다.

TRACE [{ON, | OFF,}]USER=(userid,userid,...)

지정된 사용자 ID에 대한 추적을 사용하거나(ON) 사용하지 않습니다(OFF). 기본값은 ON입니다. 이 설정은 **MODIFY RSECOMMLOG** 운영자 명령에 의해 제어되는 기본 설정을 무효화합니다. 두 가지 세부사항 레벨을 사용할 수 있습니다.

OFF	오류 메시지 전용.
ON(기본값)	오류, 경고 및 정보 메시지.

명령이 RSE 서버(rsecomm.log) 및 MVS 데이터 세트 서비스(lock.log 및 ffs*.log)의 추적 세부사항 레벨을 변경합니다. 현재 로그인하지 않은 사용자에 대해서 명령을 실행할 수 있습니다. 사용자가 로그오프할 때 설정은 활성 상태로 유지되고 사용자가 로그인할 때 다시 사용됩니다. 서버가 시작될 때 **MODIFY TRACE USER** 명령을 실행하는 것을 시뮬레이션하려면 rsecomm.properties의 USER 지시문을 사용하십시오. 이전 **MODIFY TRACE USER** 또는 **MODIFY TRACE SERVER** 운영자 명령 또는 rsecomm.properties의 USER 지시문의 기존 설정이 이 명령의 설정으로 대체됩니다.

자세한 추적은 성능 저하를 초래하므로 IBM 지원 센터의 지시 하에서만 수행해야 합니다.

TRACE [{ON, | OFF,}] SERVER={pid | (pid,pid,...)}

지정된 스레드 풀의 모든 사용자에게 대해 추적을 사용하거나(ON) 사용하지 않습니다(OFF). 여기서, pid는 RSE 스레드 풀의 프로세스 ID입니다. 기본값은 ON입니다. 이 설정은 **MODIFY RSECOMMLOG** 운영자 명령에 의해 제어되는 기본 설정을 무효화합니다. 두 가지 세부사항 레벨을 사용할 수 있습니다.

OFF	오류 메시지 전용.
ON(기본값)	오류, 경고 및 정보 메시지.

명령이 RSE 서버(rsecomm.log) 및 MVS 데이터 세트 서비스(lock.log 및 ffs*.log)의 추적 세부사항 레벨을 변경합니다. 이전 **MODIFY TRACE USER** 또는 **MODIFY TRACE SERVER** 운영자 명령 또는 rsecomm.properties의 USER 지시문의 기존 설정이 이 명령의 설정으로 대체됩니다.

자세한 추적은 성능 저하를 초래하므로 IBM 지원 센터의 지시 하에서만 수행해야 합니다.

TRACE CLEAR

MODIFY TRACE USER 및 **MODIFY TRACE SERVER** 운영자 명령 및 rsecomm.properties의 USER 지시문에 의해 설정된 추적 대체를 모두 제거합니다.

DEBUG HEAPDUMP,PID=pid

지정된 스레드 풀의 Java 힙 덤프를 요청합니다. 여기서 pid는 RSE 스레드 풀의 프로세스 ID입니다. 덤프는 rsed.envvars의 _CEE_DUMPTARG에서 지정하는 디렉토리에 작성됩니다. 기본값은 /tmp입니다. 결과는 한 개의 BPXM023I 콘솔 메시지에 표시됩니다.

```
JVMDUMP034I User requested Heap dump using '/tmp/heapdump.20120223.211'
430.16777590.0001.phd' through JVMRI
```

DEBUG JAVACORE,PID=pid

지정된 스레드 풀의 Java 코어 덤프를 요청합니다. 여기서 pid는 RSE 스레드 풀의 프로세스 ID입니다. 덤프는 rsed.envvars의 _CEE_DUMPTARG에서 지정하는 디렉토리에 작성됩니다. 기본값은 /tmp입니다. 결과는 한 개의 BPXM023I 콘솔 메시지에 표시됩니다.

```
JVMDUMP034I User requested Java dump using '/tmp/javacore.20120223.214244.16777590.0002.phd' through JVMRI
```

DEBUG GC,PID=pid

지정된 스레드 풀의 Java 가비지 콜렉션을 요청합니다. 여기서 pid는 RSE 스레드 풀의 프로세스 ID입니다.

IVP DAEMON,userid

RSE 디먼에 사용자 ID userid를 로그인하여 연결 테스트를 수행하십시오. 하나 이상의 FEK900I 콘솔 메시지와 함께 결과가 표시됩니다. FEK901I 콘솔 메시지와 함께 리턴 코드가 표시됩니다.

```
+FEK900I DAEMON IVP: SSL is disabled
+FEK900I DAEMON IVP: connected
+FEK900I DAEMON IVP: 1977
+FEK900I DAEMON IVP: 6902918
+FEK900I DAEMON IVP: Success
+FEK901I DAEMON IVP Exit code = 0
```

참고:

- 이 함수는 fekfivpd IVP(Installation Verification Program)가 수행하는 함수와 유사합니다.
- RSE 디먼이 IVP의 비밀번호로 사용되는 PassTicket을 생성하므로 비밀번호를 요청하는 WTOR(Write To Operator with Reply)이 없습니다.

IVP ISPF,userid

사용자 ID userid로 ISPF의 Client Gateway를 호출하십시오. 하나 이상의 FEK900I 콘솔 메시지와 함께 결과가 표시됩니다. FEK901I 콘솔 메시지와 함께 리턴 코드가 표시됩니다.

```
+FEK900I ISPF IVP: executed on CDFMVS08 -- Tue Sep 13 22:29:28 EDT 2011
+FEK900I ISPF IVP: executed by uid=1(IBMUSER) gid=0(SYS1)
+FEK900I ISPF IVP: using /etc/rdz/rsed.envvars
+FEK900I ISPF IVP: current address space size limit is 2147483647 (2048.0 MB)
+FEK900I ISPF IVP: maximum address space size limit is 2147483647 (2048.0 MB)
+FEK900I ISPF IVP: -----
+FEK900I ISPF IVP: /etc/rdz/ISPF.conf content:
+FEK900I ISPF IVP: -----
+FEK900I ISPF IVP: ispllib=ISP.SISPLoad
+FEK900I ISPF IVP: ispmllib=ISP.SISPMENU
+FEK900I ISPF IVP: isptlib=ISP.SISPTENU
```

```

+FEK900I ISPF IVP: isplib=ISP.SISPPENU
+FEK900I ISPF IVP: ispslib=ISP.SISPSLIB
+FEK900I ISPF IVP: sysproc=ISP.SISPCLIB,FEK.SFEKPROC
+FEK900I ISPF IVP: -----
-----
+FEK900I ISPF IVP: Host install verification for RSE
+FEK900I ISPF IVP: Review IVP log messages from HOST below :
+FEK900I ISPF IVP: -----
-----
+FEK900I ISPF IVP: Service level 22Feb2011
+FEK900I ISPF IVP: RSE connection and base TSO/ISPF session initializati
on check only
+FEK900I ISPF IVP: *** CHECK : ENVIRONMENT VARIABLES - key variables
displayed below :
+FEK900I ISPF IVP: Server PATH           = ./usr/lpp/java/J6.0/bin:/usr/l
pp/rdz/bin:/usr/lpp/ispf/bin:/bin:/usr/sbin
+FEK900I ISPF IVP: STEPLIB               = NONE
+FEK900I ISPF IVP: Temporary directory = /tmp
+FEK900I ISPF IVP: CGI_ISPHOME           = /usr/lpp/ispf
+FEK900I ISPF IVP: CGI_ISPCONF           = /etc/rdz
+FEK900I ISPF IVP: CGI_ISPWORK           = /var/rdz
+FEK900I ISPF IVP: -----
-----
+FEK900I ISPF IVP: *** CHECK : USS MODULES
+FEK900I ISPF IVP: Checking ISPF Directory : /usr/lpp/ispf
+FEK900I ISPF IVP: Checking modules in /usr/lpp/ispf/bin directory
+FEK900I ISPF IVP: Checking for ISPF configuration file ISPF.conf
+FEK900I ISPF IVP: RC=0
+FEK900I ISPF IVP: MSG: SUCCESSFUL
+FEK900I ISPF IVP: -----
-----
+FEK900I ISPF IVP: *** CHECK : TSO/ISPF INITIALIZATION
+FEK900I ISPF IVP: ( TSO/ISPF session will be initialized )
+FEK900I ISPF IVP: RC=0
+FEK900I ISPF IVP: MSG: SUCCESSFUL
+FEK900I ISPF IVP: -----
-----
+FEK900I ISPF IVP: *** CHECK: Shutting down TSO/ISPF IVP session
+FEK900I ISPF IVP: RC=0
+FEK900I ISPF IVP: MSG: SUCCESSFUL
+FEK900I ISPF IVP: -----
-----
+FEK900I ISPF IVP: Host installation verification completed successfully
+FEK900I ISPF IVP: -----
-----
+FEK901I ISPF IVP Exit code = 0

```

참고:

- 이 함수는 fekfivpi IVP(Installation Verification Program)가 수행하는 함수와 유사합니다.
- RSE 디먼이 IVP의 비밀번호로 사용되는 PassTicket을 생성하므로 비밀번호를 요청하는 WTOR(Write To Operator with Reply)이 없습니다.

IVP PASSTICKET_userid

사용자 ID userid에 생성된 PassTicket의 재사용가능성을 테스트합니다. 하나 이상의 FEK900I 콘솔 메시지와 함께 결과가 표시됩니다. FEK901I 콘솔 메시지와 함께 리턴 코드가 표시됩니다.

```
+FEK900I PASSTICKET IVP: the default applid=FEKAPPL
+FEK900I PASSTICKET IVP: Success, PassTicket IVP finished normally
+FEK901I PASSTICKET IVP Exit code = 0
```

참고:

- RACF를 보안 제품으로 사용 시 재사용가능한 PassTicket의 보안 정의에는 "NO REPLAY PROTECTION"이 필요합니다.
- 이 테스트에는 동등한 IVP(Installation Verification Program)가 없습니다. RSE 디먼을 IVP=IVP 인수로 시작하면 PassTicket 생성을 테스트하는 PassTicket IVP는 호출되지만 PassTicket 재사용가능성을 테스트할 수 없습니다.
- RSE 디먼이 IVP의 비밀번호로 사용되는 PassTicket을 생성하므로 비밀번호를 요청하는 WTOR(Write To Operator with Reply)이 없습니다.

SWITCH

새 감사 로그 파일로 전환합니다.

Stop(P)

활성 태스크를 중지하려면 **STOP** 명령을 사용하십시오. 이 명령의 축약 버전은 문자 P입니다.

```
>> STOP proname <<
```

그림 43. *STOP* 운영자 명령

procname

서버를 시작하는 데 사용된 프로시저 라이브러리의 멤버 이름. 통합 디버거, JES 작업 모니터, RSE 디맨의 경우, 각각 호스트 시스템 구성 중 사용되는 기본 이름은 DBGMGR, JMON 및 RSED입니다.

구문 다이어그램 읽는 방법

구문 도표는 사용자가 입력한 내용을 운영 체제에서 올바르게 해석할 수 있도록 명령을 지정하는 방법을 보여줍니다. 기본 경로인 수평선을 따라 왼쪽에서 오른쪽으로, 위에서 아래로 구문 다이어그램을 읽으십시오.

기호

구문 다이어그램에는 다음 기호가 사용됩니다.

기호	설명
>>	구문 다이어그램의 시작을 표시합니다.
>	구문 다이어그램이 계속됨을 표시합니다.
	단편의 시작과 끝 또는 구문 다이어그램의 일부를 표시합니다.
><	구문 다이어그램의 끝을 표시합니다.

피연산자

구문 다이어그램에서는 다음 유형의 피연산자가 사용됩니다.

- 필수 피연산자는 기본 경로 라인에 표시됩니다.

>>—REQUIRED_OPERAND—><

- 선택적 피연산자는 기본 경로 라인 아래에 표시됩니다.

>>—OPTIONAL_OPERAND—><

- 기본 피연산자는 기본 경로 라인 위에 표시됩니다.

>>—DEFAULT_OPERAND—><

피연산자는 키워드 또는 변수로 분류됩니다.

- 키워드는 반드시 제공되어야 하는 상수입니다. 구문 다이어그램에서 키워드가 대문자 또는 소문자로 표시되면, 대문자 부분이 키워드의 약어입니다(예: KEYword). 키워드는 대소문자를 구분하지 않습니다.
- 변수는 이탤릭체로 되고, 소문자로 표시되며 사용자가 입력하는 이름 또는 값을 나타냅니다. 예를 들어, 데이터 세트 이름은 변수입니다. 변수는 대소문자를 구분할 수 있습니다.

구문 예제

다음 예제에서 USER 명령은 키워드입니다. 필수 변수 매개변수는 user_id이고 선택적 변수 매개변수는 password입니다. 변수 매개변수를 사용자 값으로 바꾸십시오.

>>—USER—user_id—password—><

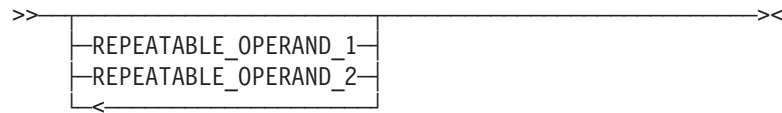
영숫자가 아닌 문자 및 공백

다이어그램이 영숫자가 아닌 문자(소괄호, 점, 쉼표, 등호 및 공백)를 표시하는 경우, 구문의 일부로 문자를 코딩해야 합니다. 이 예에서는 OPERAND=(001 0.001)을 코딩해야 합니다.

>>—OPERAND—=(—001— —0.001—)—><

둘 이상의 피연산자 선택하기

피연산자 그룹 왼쪽으로 리턴하는 화살표는 둘 이상을 선택할 수 있거나 하나만 반복될 수 있음을 의미합니다.



한 라인 초과

다이아그램이 한 라인보다 길면, 첫 라인은 한 개의 화살촉으로 끝나고 두 번째 라인은 화살촉으로 시작합니다.

```
>>—| The first line of a syntax diagram that is longer than one line |—>
>—| The continuation of the subcommands, parameters, or both |————><
```

구문 단편

일부 다이어그램에는 구문 단편이 포함될 수 있으며, 이는 너무 길거나 너무 복잡하거나 너무 반복적인 다이어그램을 분할하기 제공됩니다. 구문 단편 이름에서는 대소문자가 혼합되며 단편 표제 및 다이어그램에 표시됩니다. 단편은 기본 다이어그램 아래에 있습니다.

```
>>—| Syntax fragment |————><

Syntax fragment:
|—1ST_OPERAND—,—2ND_OPERAND—,—3RD_OPERAND—|
```

제 12 장 호스트 구성 참조서

이 절에서는 *Rational Developer for System z 호스트* 구성 참조서(SA30-4501)에 있는 정보의 요약을 제공합니다. 자세한 내용은 해당 서적을 참조하십시오.

Developer for System z 이해

Developer for System z 호스트 시스템은 클라이언트에게 호스트 시스템 서비스 및 데이터에 대한 액세스를 제공하기 위해 상호작용하는 여러 컴포넌트로 구성됩니다. 이러한 컴포넌트의 디자인을 이해하면 올바른 구성을 쉽게 결정할 수 있습니다.

보안 고려사항

Developer for System z는 비메인프레임 워크스테이션 사용자에게 메인프레임 액세스를 제공합니다. 따라서 연결 요청 유효성 검증, 호스트 시스템과 워크스테이션 간의 보안 통신 제공, 권한 부여 및 감사 활동은 제품 구성의 중요한 측면입니다.

TCP/IP 고려사항

Developer for System z는 TCP/IP를 사용하여 비메인프레임 워크스테이션 사용자에게 메인프레임 액세스를 제공합니다. 또한 다양한 컴포넌트와 기타 제품 간의 통신에도 TCP/IP를 사용합니다.

WLM 고려사항

전통적인 z/OS 애플리케이션과 달리 Developer for System z는 워크로드 관리자(WLM)가 쉽게 식별할 수 있는 단일 애플리케이션이 아닙니다. Developer for System z는 클라이언트에게 호스트 시스템 서비스 및 데이터에 대한 액세스를 제공하기 위해 상호작용하는 여러 컴포넌트로 구성됩니다. 이러한 서비스 중 일부는 서로 다른 주소 공간에서 활성화되므로 여러 WLM 분류가 발생합니다.

튜닝 고려사항

RSE(Remote Systems Explorer)는 Developer for System z의 핵심입니다. 클라이언트로부터의 연결과 워크로드를 관리하기 위해 RSE는 스레드 풀링 주소 공간을 제어하는 디먼 주소 공간으로 구성됩니다. 디먼은 연결 및 관리를 위한 포컬 포인트의 역할을 하는 반면, 스레드 풀은 클라이언트 워크로드를 처리합니다.

이 구성에서는 RSE를 Developer for System z 설정을 조정하기 위한 기본 대상으로 작성합니다. 그러나 각각 17개 이상의 스레드, 일정 양의 스토리지, 하나 이상의 주소 공간을 사용하는 수백 명의 사용자를 유지보수하려면 Developer for System z와 z/OS를 둘 다 올바르게 구성해야 합니다.

성능 고려사항

z/OS는 사용자 정의가 매우 용이한 운영 체제이므로 시스템 변경(때때로 사소한 변경 포함) 시 전체 성능에 막대한 영향을 줄 수 있습니다. 이 장에서는 Developer for System z 성능 향상을 위해 수행할 수 있는 몇 가지 변경사항을 강조합니다.

클라이언트로 푸시 고려사항

클라이언트로 푸시 또는 호스트 기반 클라이언트 제어는 다음 항목에 대한 중앙 관리를 지원합니다.

- 클라이언트 구성 파일
- 클라이언트 제품 버전
- 프로젝트 정의

CICSTS 고려사항

이 장에는 CICS Transaction Server 관리자에게 유용한 정보가 들어 있습니다.

사용자 종료 고려사항

이 장에서는 종료 루틴을 기록하여 Developer for System z의 기능을 개선하는 데 필요한 정보를 제공합니다.

TSO 환경 사용자 정의

이 장에서는 Developer for System z의 TSO 환경에 DD 문과 데이터 세트를 추가하여 TSO 로그인 프로시저를 모방하는 데 필요한 정보를 제공합니다.

다중 인스턴스 실행

Developer for System z의 여러 인스턴스들을 동일한 시스템에서 활성화시키려는 경우(예: 업그레이드 테스트)가 있습니다. 그러나 TCP/IP 포트와 같은 일부 자원은 공유할 수 없으므로 항상 기본값을 적용할 수 있는 것은 아닙니다. 이 장에 있는 정보를 사용하여 서로 다른 Developer for System z 인스턴스의 공존을 계획한 후 이 구성 안내서를 사용하여 사용자 정의하십시오.

구성 문제점 해결

이 장에서는 Developer for System z의 구성 중에 발생하는 몇 가지 공통 문제점을 해결하는 방법을 제공하며 다음 절로 구성됩니다.

- FEKLOGS를 사용한 로그 및 설정 분석
- 로그 파일
- 덤프 파일
- 추적
- z/OS UNIX 권한 비트
- 예약된 TCP/IP 포트
- 주소 공간 크기
- APPC 트랜잭션 및 TSO 명령 서비스
- 기타 정보

SSL 및 X.509 인증 설정

이 부록은 SSL(Secure Socket Layer) 설정 시 또는 기존 설정을 검사 또는 수정하는 동안 발생할 수 있는 일부 공통 문제점을 해결하는 데 도움이 되도록 제공되었습니다. 이 부록은 사용자가 X.509 인증서를 사용하여 스스로를 인증하는 것을 지원할 수 있도록 샘플 설정도 제공합니다.

TCP/IP 설정

이 부록은 TCP/IP 설정 시, 또는 기존 설정의 검사 또는 수정 도중 발생할 수 있는 몇 가지 공통 문제점을 해결하는 데 도움이 되도록 제공되었습니다.

참고 문헌

참조된 서적

이 책에 참조된 서적은 다음과 같습니다.

표 27. 참조된 서적

책 제목	주문 번호	참조	참조 웹 사이트
Program Directory for IBM Rational Developer for System z	GI11-8298	Developer for System z	http://www-01.ibm.com/support/docview.wss?uid=swg27038517
IBM Rational Developer for System z 호스트 유틸리티에 대한 프로그램 디렉토리	GI13-2864	Developer for System z	http://www-01.ibm.com/support/docview.wss?uid=swg27038517
Rational Developer for System z 전제조건	SA30-3650	Developer for System z	http://www-01.ibm.com/support/docview.wss?uid=swg27038517
Rational Developer for System z 호스트 구성 빠른 시작 안내서	GA30-4183	Developer for System z	http://www-01.ibm.com/support/docview.wss?uid=swg27038517
Rational Developer for System z 호스트 구성 안내서	SA30-4578	Developer for System z	http://www-01.ibm.com/support/docview.wss?uid=swg27038517
Rational Developer for System z 호스트 구성 참조서	SA30-4501	Developer for System z	http://www-01.ibm.com/support/docview.wss?uid=swg27038517
Rational Developer for System z 호스트 구성 유틸리티 안내서	SA30-4582	Developer for System z	http://www-01.ibm.com/support/docview.wss?uid=swg27038517
Rational Developer for System z 메시지 및 코드	SA30-4577	Developer for System z	http://www-01.ibm.com/support/docview.wss?uid=swg27038517
Rational Developer for System z Answers to common host configuration and maintenance issues	SA30-4502	Developer for System z	http://www-01.ibm.com/support/docview.wss?uid=swg27038517
Rational Developer for System z Common Access Repository Manager Developer's Guide	SC23-7660	Developer for System z	http://www-01.ibm.com/support/docview.wss?uid=swg27038517
Rational Developer for System z 전제조건	SA30-3650	Developer for System z	http://www.ibm.com/software/rational/products/developer/systemz/library/index.html
Rational Developer for System z 호스트 구성 빠른 시작 안내서	GA30-4183	Developer for System z	http://www.ibm.com/software/rational/products/developer/systemz/library/index.html
SCLM Developer Toolkit Administrator's Guide	SC23-9801	Developer for System z	http://www-01.ibm.com/support/docview.wss?uid=swg27038517
Using APPC to provide TSO command services	SC14-7291	백서	http://www-01.ibm.com/support/docview.wss?uid=swg27038517
Using ISPF Client Gateway to provide CARMA services	SC14-7292	백서	http://www-01.ibm.com/support/docview.wss?uid=swg27038517

표 27. 참조된 서적 (계속)

책 제목	주문 번호	참조	참조 웹 사이트
Communications Server IP Configuration Guide	SC31-8775	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
Communications Server IP Configuration Reference	SC31-8776	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
Communications Server IP Diagnosis Guide	GC31-8782	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
Communications Server IP System Administrator's Commands	SC31-8781	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
Communications Server SNA Network Implementation Guide	SC31-8777	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
Communications Server SNA Operations	SC31-8779	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
Cryptographic Services System SSL Programming	SC24-5901	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
DFSMS Macro Instructions for Data Sets	SC26-7408	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
DFSMS Using data sets	SC26-7410	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
Language Environment Customization	SA22-7564	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
Language Environment Debugging Guide	GA22-7560	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
I MVS 진단: 도구 및 서비스 지원	GA22-7589	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
MVS Initialization and Tuning Guide	SA22-7591	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
MVS Initialization and Tuning Reference	SA22-7592	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
MVS JCL Reference	SA22-7597	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
MVS Planning APPC/MVS Management	SA22-7599	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
MVS Planning Workload Management	SA22-7602	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
MVS System Commands	SA22-7627	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
Security Server RACF Command Language Reference	SA22-7687	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
Security Server RACF Security Administrator's Guide	SA22-7683	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
TSO/E Customization	SA22-7783	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
TSO/E REXX Reference	SA22-7790	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
UNIX System Services Command Reference	SA22-7802	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
UNIX System Services Planning	GA22-7800	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/

표 27. 참조된 서적 (계속)

책 제목	주문 번호	참조	참조 웹 사이트
UNIX System Services User's Guide	SA22-7801	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
Using REXX and z/OS UNIX System Services	SA22-7806	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
Java™ Diagnostic Guide	SC34-6650	Java 6.0	http://www.ibm.com/developerworks/java/jdk/diagnosis/
Java SDK and Runtime Environment User Guide	/	Java 6.0	http://www-03.ibm.com/servers/eserver/zseries/software/java/
Resource Definition Guide	SC34-6430	CICSTS 3.1	http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html
Resource Definition Guide	SC34-6815	CICSTS 3.2	http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html
Resource Definition Guide	SC34-7000	CICSTS 4.1	https://publib.boulder.ibm.com/infocenter/cicsts/v4r1/index.jsp?topic=/com.ibm.cics.ts.home.doc/library/library_html.html
Resource Definition Guide	SC34-7181	CICSTS 4.2	https://publib.boulder.ibm.com/infocenter/cicsts/v4r2/index.jsp?topic=/com.ibm.cics.ts.home.doc/library/library_html.html
RACF Security Guide	SC34-6454	CICSTS 3.1	http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html
RACF Security Guide	SC34-6835	CICSTS 3.2	http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html
RACF Security Guide	SC34-7003	CICSTS 4.1	https://publib.boulder.ibm.com/infocenter/cicsts/v4r1/index.jsp?topic=/com.ibm.cics.ts.home.doc/library/library_html.html
RACF Security Guide	SC34-7179	CICSTS 4.2	https://publib.boulder.ibm.com/infocenter/cicsts/v4r2/index.jsp?topic=/com.ibm.cics.ts.home.doc/library/library_html.html
Language Reference	SC27-1408	Enterprise COBOL for z/OS	http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html

이 책에 참조된 웹 사이트 다음과 같습니다.

표 28. 참조된 웹 사이트

설명	참조 웹 사이트
Developer for System z Information Center	http://pic.dhe.ibm.com/infocenter/ratdevz/v9r0/index.jsp
Developer for System z Library	http://www-01.ibm.com/support/docview.wss?uid=swg27038517
Developer for System z 홈 페이지	http://www-03.ibm.com/software/products/us/en/developerforsystemz/
Developer for System z 권장 서비스	http://www-01.ibm.com/support/docview.wss?rs=2294&context=SS2QJ2&uid=swg27006335
Developer for System z 향상 요청	https://www.ibm.com/developerworks/support/rational/rfe/
z/OS 인터넷 라이브러리	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
CICSTS Information Center	https://publib.boulder.ibm.com/infocenter/cicsts/v4r1/index.jsp

표 28. 참조된 웹 사이트 (계속)

설명	참조 웹 사이트
IBM Tivoli® Directory Server	http://www-01.ibm.com/software/tivoli/products/directory-server/
문제점 판별 도구 플러그인	http://www-01.ibm.com/software/awdtools/deployment/pdtpplugins/
Apache Ant 다운로드	http://ant.apache.org/
Java keytool 문서	http://java.sun.com/j2se/1.5.0/docs/tooldocs/solaris/keytool.html
CA 지원 홈 페이지	https://support.ca.com/

정보 서적

다음 서적은 필수 호스트 시스템 컴포넌트에 대한 설정 문제점을 이해하는 데 도움이 됩니다.

표 29. 정보 서적

책 제목	주문 번호	참조	참조 웹 사이트
ABCs of z/OS System Programming Volume 9 (z/OS UNIX)	SG24-6989	Redbook	http://www.redbooks.ibm.com/
System Programmer's Guide to: Workload Manager	SG24-6472	Redbook	http://www.redbooks.ibm.com/
TCPIP Implementation Volume 1: Base Functions, Connectivity, and Routing	SG24-7532	Redbook	http://www.redbooks.ibm.com/
TCPIP Implementation Volume 3: High Availability, Scalability, and Performance	SG24-7534	Redbook	http://www.redbooks.ibm.com/
TCP/IP Implementation Volume 4: Security and Policy-Based Networking	SG24-7535	Redbook	http://www.redbooks.ibm.com/
Tivoli Directory Server for z/OS	SG24-7849	Redbook	http://www.redbooks.ibm.com/

IBM Rational Developer for System z의 문서 주의사항

© Copyright IBM Corporation 2009, 2013.

U.S. Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

이 정보는 미국에서 제공되는 제품 및 서비스용으로 작성된 것입니다.

IBM은 다른 국가에서 이 책에 기술된 제품, 서비스 또는 기능을 제공하지 않을 수도 있습니다. 현재 사용할 수 있는 제품 및 서비스에 대한 정보는 한국 IBM 담당자에게 문의하십시오. 이 책에서 IBM 제품, 프로그램 또는 서비스를 언급했다고 해서 해당 IBM 제품, 프로그램 또는 서비스만을 사용할 수 있다는 것을 의미하지는 않습니다. IBM의 지적 재산을 침해하지 않는 한, 기능상으로 동등한 제품, 프로그램 또는 서비스를 대신 사용할 수도 있습니다. 그러나 비IBM 제품, 프로그램 또는 서비스의 운영에 대한 평가 및 검증은 사용자의 책임입니다.

IBM은 이 책에서 다루고 있는 특정 내용에 대해 특허를 보유하고 있거나 현재 특허 출원 중일 수 있습니다. 이 책을 제공한다고 해서 특허에 대한 라이선스까지 부여하는 것은 아닙니다. 라이선스에 대한 의문사항은 다음으로 문의하십시오.

135-700

서울특별시 강남구 도곡동 467-12

군인공제회관빌딩

한국 아이.비.엠 주식회사

고객만족센터

전화번호: 080-023-8080

2바이트(DBCS) 정보에 관한 라이선스 문의는 한국 IBM 고객만족센터에 문의하거나 다음 주소로 서면 문의하시기 바랍니다.

Intellectual Property Licensing

Legal and Intellectual Property Law

IBM Japan, Ltd.

1623-14, Shimotsuruma, Yamato-shi

Kanagawa 242-8502 Japan

다음 단락은 현지법과 상충하는 영국이나 기타 국가에서는 적용되지 않습니다. IBM은 타인의 권리 비침해, 상품성 및 특정 목적에의 적합성에 대한 묵시적 보증을 포함하여 (단, 이에 한하지 않음) 묵시적이든 명시적이든 어떠한 종류의 보증 없이 이 책을 "현

상태대로" 제공합니다. 일부 국가에서는 특정 거래에서 명시적 또는 묵시적 보증의 면책사항을 허용하지 않으므로, 이 사항이 적용되지 않을 수도 있습니다.

이 정보에는 기술적으로 부정확한 내용이나 인쇄상의 오류가 있을 수 있습니다. 이 정보는 주기적으로 변경되며, 변경된 사항은 최신판에 통합됩니다. IBM은 이 책에서 설명한 제품 및/또는 프로그램을 사전 통지 없이 언제든지 개선 및/또는 변경할 수 있습니다.

이 정보에서 언급되는 비IBM의 웹 사이트는 단지 편의상 제공된 것으로, 어떤 방식으로든 이들 웹 사이트를 옹호하고자 하는 것은 아닙니다. 해당 웹 사이트의 자료는 본 IBM 제품 자료의 일부가 아니므로 해당 웹 사이트 사용으로 인한 위험은 사용자 본인 이 감수해야 합니다.

IBM은 귀하의 권리를 침해하지 않는 범위 내에서 적절하다고 생각하는 방식으로 귀하가 제공한 정보를 사용하거나 배포할 수 있습니다.

(i) 독립적으로 작성된 프로그램과 기타 프로그램(본 프로그램 포함)간의 정보 교환 및
(ii) 교환된 정보의 상호 이용을 목적으로 본 프로그램에 관한 정보를 얻고자 하는 라이선스 사용자는 다음 주소로 문의하십시오.

135-700

서울특별시 강남구 도곡동 467-12

군인공제회관빌딩

한국 아이.비.엠 주식회사

고객만족센터

이러한 정보는 해당 조건(예를 들면, 사용료 지불 등)하에서 사용될 수 있습니다.

본 문서에 기술된 라이선스가 있는 프로그램 및 사용 가능한 모든 라이선스가 있는 자료는 IBM이 IBM 기본 계약, IBM 프로그램 라이선스 계약(IPLA) 또는 이와 동등한 계약에 따라 제공한 것입니다.

본 문서에 포함된 모든 성능 데이터는 제한된 환경에서 산출된 것입니다. 따라서 다른 운영 환경에서 얻어진 결과는 상당히 다를 수 있습니다. 일부 성능은 개발 단계의 시스템에서 측정되었을 수 있으므로 이러한 측정치가 일반적으로 사용되고 있는 시스템에서도 동일하게 나타날 것이라고는 보증할 수 없습니다. 또한 일부 성능은 추정을 통해 추측되었을 수도 있으므로 실제 결과는 다를 수 있습니다. 이 책의 사용자는 해당 데이터를 본인의 특정 환경에서 검증해야 합니다.

비IBM 제품에 관한 정보는 해당 제품의 공급업체, 공개 자료 또는 기타 범용 소스로부터 얻은 것입니다. IBM에서는 이러한 비IBM 제품을 반드시 테스트하지 않았으므로, 이들 제품과 관련된 성능의 정확성, 호환성 또는 기타 주장에 대해서는 확인할 수 없습니다. 비IBM 제품의 성능에 대한 의문사항은 해당 제품의 공급업체에 문의하십시오.

IBM이 제시하는 방향 또는 의도에 관한 모든 언급은 특별한 통지 없이 변경될 수 있습니다.

이 정보는 계획 수립 목적으로만 사용됩니다. 이 정보는 기술된 제품이 GA(General Availability)되기 전에 변경될 수 있습니다.

이 정보에는 일상의 비즈니스 운영에서 사용되는 자료 및 보고서에 대한 예제가 들어 있습니다. 이들 예제에는 개념을 가능한 완벽하게 설명하기 위하여 개인, 회사, 상표 및 제품의 이름이 사용될 수 있습니다. 이들 이름은 모두 가공의 것이며 실제 기업의 이름 및 주소와 유사하더라도 이는 전적으로 우연입니다.

저작권 라이선스

이 정보에는 여러 운영 플랫폼에서의 프로그래밍 기법을 보여주는 원어로 된 샘플 애플리케이션이 들어 있습니다. 귀하는 이러한 샘플 프로그램의 작성 기준이 된 운영 플랫폼의 애플리케이션 프로그래밍 인터페이스(API)에 부합하는 애플리케이션을 개발, 사용, 판매 또는 배포할 목적으로 추가 비용 없이 이들 샘플 프로그램을 어떠한 형태로든 복사, 수정 및 배포할 수 있습니다. 이러한 샘플 프로그램은 모든 조건하에서 완전히 테스트된 것은 아닙니다. 따라서 IBM은 이러한 프로그램의 신뢰성, 서비스 가능성 또는 기능을 보증하거나 진술하지 않습니다. 본 샘플 프로그램은 일체의 보증 없이 "현상태대로" 제공됩니다. IBM은 샘플 프로그램의 사용으로 인해 발생하는 어떤 손해에 대해서도 책임을 지지 않습니다.

이러한 샘플 프로그램 또는 파생 제품의 각 사본이나 그 일부에는 반드시 다음과 같은 저작권 표시가 포함되어야 합니다.

© (귀하의 회사명) (연도). 이 코드의 일부는 IBM Corp.의 샘플 프로그램에서 파생됩니다. © Copyright IBM Corp. 2009, 2013.

이 정보를 소프트웨어로 확인하는 경우에는 사진과 컬러 삽화가 제대로 나타나지 않을 수도 있습니다.

상표 정보

IBM, IBM 로고 및 ibm.com은 전세계 여러 국가에 등록된 International Business Machines Corp.의 상표 또는 등록상표입니다. 기타 제품 및 서비스 이름은 IBM 또는 타사의 상표입니다. 현재 IBM 상표 목록은 웹(www.ibm.com/legal/copytrade.shtml)에 있습니다.

Adobe, Adobe 로고, PostScript 및 PostScript 로고는 미국 또는 기타 국가에서 사용되는 Adobe Systems Incorporated의 상표 또는 등록상표입니다.

Linux는 미국 또는 기타 국가에서 사용되는 Linus Torvalds의 등록상표입니다.

Windows는 미국 또는 기타 국가에서 사용되는 Microsoft Corporation의 상표입니다.

UNIX는 미국 또는 기타 국가에서 사용되는 The Open Group의 등록상표입니다.

Java 및 모든 Java 기반 상표와 로고는 Oracle 및/또는 그 계열사의 상표 또는 등록상표입니다.

기타 제품 및 서비스 이름은 IBM 또는 타사의 상표입니다.

저작권 라이선스

이 정보에는 여러 운영 플랫폼에서의 프로그래밍 기법을 보여주는 원어로 된 샘플 애플리케이션이 들어 있습니다. 귀하는 이러한 샘플 프로그램의 작성 기준이 된 운영 플랫폼의 애플리케이션 프로그래밍 인터페이스(API)에 부합하는 애플리케이션을 개발, 사용, 판매 또는 배포할 목적으로 추가 비용 없이 이들 샘플 프로그램을 어떠한 형태로든 복사, 수정 및 배포할 수 있습니다. 이러한 샘플 프로그램은 모든 조건하에서 완전히 테스트된 것은 아닙니다. 따라서 IBM은 이러한 프로그램의 신뢰성, 서비스 가능성 또는 기능을 보증하거나 진술하지 않습니다. 본 샘플 프로그램은 일체의 보증 없이 "현상태대로" 제공됩니다. IBM은 샘플 프로그램의 사용으로 인해 발생하는 어떤 손해에 대해서도 책임을 지지 않습니다.

상표 정보

IBM, IBM 로고 및 `ibm.com`은 전세계 여러 국가에 등록된 International Business Machines Corp.의 상표 또는 등록상표입니다. 기타 제품 및 서비스 이름은 IBM 또는 타사의 상표입니다. 현재 IBM 상표 목록은 웹(www.ibm.com/legal/copytrade.shtml)에 있습니다.

CA Endevor는 CA Technologies의 등록상표입니다.

Rational은 미국 또는 기타 국가에서 사용되는 International Business Machines Corporation 및 Rational Software Corporation의 상표입니다.

Intel 및 Pentium은 미국 또는 기타 국가에서 사용되는 Intel Corporation의 상표입니다.

Microsoft, Windows 및 Windows 로고는 미국 또는 기타 국가에서 사용되는 Microsoft Corporation의 상표 또는 등록상표입니다.

Java 및 모든 Java 기반 상표와 로고는 미국 및 기타 국가에서 사용되는 Sun Microsystems, Inc.의 상표 또는 등록상표입니다.

UNIX는 미국 또는 기타 국가에서 사용되는 The Open Group의 등록상표입니다.

고객 의견서

IBM Rational Developer for System z

버전 9.0.1
구성 안내서

SA30-4578-10

성명

주소

회사 또는 단체명

전화번호

고객 의견서
SA30-4578-10



선을 따라
자르거나
접으십시오

접어서 붙이십시오

스테이플러를 사용하지 마시오

접어서 붙이십시오

우 표
붙이는
곳

135-700
Building 501
P.O Box 12195
Research Triangle Park, NC
USA 27709-2195

접어서 붙이십시오

스테이플러를 사용하지 마시오

접어서 붙이십시오

SA30-4578-10

선을 따라
자르거나
접으십시오



Printed in Korea

SA30-4578-10

