



Informatica® Multidomain MDM
10.4 HotFix 2

**버전 10.1, 10.2, 10.3 또는 10.4
에서 업그레이드**

Informatica Multidomain MDM 버전 10.1, 10.2, 10.3 또는 10.4에서 업그레이드
10.4 HotFix 2
2020년12월

© 저작권 Informatica LLC 1998, 2021

이 소프트웨어와 설명서는 사용 및 공개에 대한 제한 사항이 포함되어 있는 별도의 사용권 계약에 따라서만 제공됩니다. 본 문서의 어떤 부분도 Informatica LLC의 사전 통지 없이 어떠한 형태나 수단(전자적, 사진 복사, 녹음 등)으로 복제되거나 전송될 수 없습니다.

미국 정부 권한. 미국 정부 고객에게 제공되는 프로그램, 소프트웨어, 데이터베이스, 관련 문서 및 기술 데이터는 해당하는 연방 입수 규정 및 기관별 보안 규정에 따라 "상용 컴퓨터 소프트웨어" 또는 "상용 기술 데이터"입니다. 따라서 사용, 복제, 공개, 수정 및 조정은 해당하는 정부 계약에 규정된 제한 사항 및 라이선스 조건을 따르며, 정부 계약 조건에 의해 적용 가능한 한도 내에서, FAR 52.227-19, 상용 소프트웨어 라이선스에 규정된 추가 권한이 적용됩니다.

Informatica, Informatica 로고 및 ActiveVOS는 미국과 전 세계 여러 관할 국가에서 Informatica LLC의 상표 또는 등록 상표입니다. Informatica 상표의 현재 목록은 <https://www.informatica.com/trademarks.html>에서 확인할 수 있습니다. 다른 회사 및 제품명은 해당 소유자의 상표 또는 등록 상표일 수 있습니다.

이 소프트웨어 및/또는 설명서의 일부에는 타사의 저작권이 적용될 수 있습니다. 필요한 타사 고지 사항은 제품에 포함되어 있습니다.

이 설명서의 정보는 예고 없이 변경될 수 있습니다. 이 문서에서 문제가 발견되는 경우 infa_documentation@informatica.com으로 보고해 주십시오.

Informatica 제품은 제품이 제공될 당시의 계약 조건에 따라 보증됩니다. Informatica는 상품성과 특정 목적에의 적합성에 대한 보증 그리고 비침해에 대한 보증 또는 조건을 포함하여 어떠한 종류의 명시적이거나 묵시적인 보증 없이 이 문서의 정보를 "있는 그대로" 제공합니다.

발행 날짜: 2021-01-15

목차

서문	9
Informatica 리소스	9
Informatica 네트워크	9
Informatica 기술 자료	9
Informatica 설명서	9
Informatica Product Availability Matrix	10
Informatica Velocity	10
Informatica Marketplace	10
Informatica 글로벌 고객 지원 센터	10
장 1: 업그레이드 개요	11
소개	11
업그레이드 프로세스 선택	12
업그레이드 프로세스 선택 지침	12
클린 업그레이드에 대한 업그레이드 프로세스	12
인플레이스 업그레이드의 업그레이드 프로세스	13
업그레이드 문서화	13
장 2: 업그레이드 전 태스크	15
업그레이드 전 태스크	15
업그레이드 준비	15
환경 준비	16
기존 작업 처리	20
MDM Hub 구성 업데이트	20
원래 스키마 소유자에 연산 참조 저장소를 등록합니다	21
BPM 업그레이드 준비	23
워크플로우 어댑터 선택	24
독립 실행형 ActiveVOS를 포함한 ActiveVOS로 마이그레이션(인플레이스 업그레이드)	25
빌드 속성 파일 편집(인플레이스 업그레이드)	25
장 3: 데이터베이스 태스크	26
데이터베이스 태스크 개요	26
Oracle 데이터베이스 설정	26
1단계. Oracle 설치 및 구성	27
2단계. 데이터베이스 및 테이블스페이스 생성	27
3단계. 데이터베이스 권한 및 연결 설정	30
4단계. ActiveVOS 스키마 생성4단계. ActiveVOS 데이터베이스 생성	30
Microsoft SQL Server 설정	33
1단계. Microsoft SQL Server 설치 및 구성	33
2단계. Microsoft SQL Server 데이터 파일 저장소 생성	34

3단계. ODBC 드라이버 설치.	34
4단계. ActiveVOS 스키마 생성4단계. ActiveVOS 데이터베이스 생성.	34
IBM Db2 데이터베이스 설정.	37
1단계. IBM Db2 설치 및 구성.	37
2단계. 데이터베이스 및 테이블스페이스 생성.	37
수동으로 데이터베이스 및 테이블스페이스 생성.	38
스크립트를 사용하여 데이터베이스 및 테이블스페이스 생성.	40
3단계. 데이터베이스 서버에서 패키지 바인딩.	41
4단계. ActiveVOS 스키마 생성4단계. ActiveVOS 데이터베이스 생성.	42

장 4: 응용 프로그램 서버 태스크..... 45

응용 프로그램 서버 태스크 개요.	45
JBoss 설정.	45
Java Virtual Machine 구성.	46
전체 프로필에 대한 서버 속성 구성.	49
ActiveVOS 콘솔 관리자 생성.	50
JBoss 시작.	51
Oracle WebLogic 설정.	52
Java Virtual Machine 구성.	52
WebLogic 서버 인증 비활성화.	56
ActiveVOS 콘솔 관리자 생성.	56
추가 Oracle WebLogic 구성.	57
독립 실행형 처리 서버 인스턴스에 대한 WebLogic 구성.	57
여러 MDM Hub 마스터 데이터베이스에 대한 WebLogic 구성.	60
HTTPS 프로토콜 구성.	60
IBM WebSphere 설정.	61
Java Virtual Machine 구성.	61
MDM Hub 환경에서 암호 암호화.	65
WebSphere 환경에서 보안 프로필 생성.	65
ActiveVOS 콘솔 관리자 생성.	66
MDM Hub 배포를 위한 SOAP 요청 제한 시간 구성.	67
추가 IBM WebSphere 구성.	67
독립 실행형 처리 서버 인스턴스에 대한 WebSphere 구성.	67
여러 MDM Hub 마스터 데이터베이스에 대한 WebSphere 구성.	71
HTTPS 프로토콜 구성.	71
Informatica Data Director에 대한 WebSphere 구성.	72

장 5: Hub 저장소 업그레이드..... 73

Hub 저장소 업그레이드 개요.	73
Hub 저장소 복제(클린 업그레이드).	73
영어가 아닌 로캘로 설정된 데이터베이스.	74
자세한 정보 표시 모드에서 MDM Hub 마스터 데이터베이스 업그레이드.	74

자동 모드에서 MDM Hub 마스터 데이터베이스 업그레이드.	75
자세한 정보 표시 모드에서 연산 참조 저장소 데이터베이스 업그레이드.	76
자동 모드에서 연산 참조 저장소 데이터베이스 업그레이드.	79
업그레이드 스크립트가 성공적으로 실행되었는지 확인.	80

장 6: Hub 서버 업그레이드(인플레이스 업그레이드)..... 81

Hub 서버 업그레이드 개요.	81
그래픽 모드에서 Hub 서버 업그레이드.	82
콘솔 모드에서 Hub 서버 업그레이드.	85
자동 모드에서 Hub 서버 업그레이드.	87
속성 파일 구성.	88
자동 업그레이드 실행.	89
patchInstallSetup 스크립트 실행.	89
업그레이드 기록 폴더에 Hub 서버 로그 파일 복사.	90
Hub 서버 업그레이드 재적용(선택 사항).	91

장 7: 처리 서버 업그레이드(인플레이스 업그레이드)..... 92

처리 서버 업그레이드 개요.	92
그래픽 모드에서 처리 서버 업그레이드.	92
콘솔 모드에서 처리 서버 업그레이드.	94
자동 모드에서 처리 서버 업그레이드.	95
속성 파일 구성.	96
처리 서버 자동 업그레이드 실행.	96
Informatica 주소 확인 5 통합으로의 업그레이드 단계.	97
일치 인구집단 구성.	99
일치 인구집단 활성화.	99
업그레이드 설명서 디렉터리에 처리 서버 로그 파일 복사.	101
처리 서버 업그레이드 재적용(선택 사항).	101

장 8: 리소스 키트 업그레이드 (인플레이스 업그레이드)..... 102

리소스 키트 업그레이드 개요.	102
리소스 키트 제거(인플레이스 업그레이드).	102
리소스 키트 설치.	103

장 9: 업그레이드 후 태스크..... 107

업그레이드 후 태스크.	107
Microsoft SQL Server 2017용 JDBC 드라이버 구성.	108
속성 업데이트.	108
JBoss 업그레이드 후 태스크.	109
인플레이스 업그레이드의 업그레이드 후 태스크 수행.	109
개체, 열 및 사용되지 않는 개체에 대한 참조 삭제.	109
Hub 서버 배포를 위한 PostInstall 스크립트 실행(조건부).	110

Hub 콘솔 클라이언트 구성.	111
WebSphere 관리 보안 구성.	111
연산 참조 저장소 등록 해제.	111
EAR 파일 제거 및 데이터 소스 제거.	112
WebSphere 관리 콘솔에서 WebSphere 관리 보안 활성화.	112
Hub 서버 및 처리 서버 속성 구성.	112
Hub 서버 PostInstallSetup 스크립트 수동 실행.	113
처리 서버 PostInstallSetup 스크립트 실행.	113
연산 참조 저장소 등록.	114
클래스 로더 구성 WebSphere에서.	118
연산 참조 저장소 등록.	119
업그레이드된 메타데이터 유효성 검사.	123
메타데이터 유효성 검사.	123
유효성 검사 결과 저장.	124
메타데이터 유효성 검사 메시지 확인.	124
지역화된 스키마 업데이트.	125
클린 업그레이드의 업그레이드 후 태스크 수행.	125
스키마 암호 암호화.	126
스키마 암호 업데이트.	126
연산 참조 저장소 연결 테스트 및 업데이트.	126
ActiveVOS 연결 테스트 및 업데이트.	126
처리 서버 테스트 및 추가.	127
플랫폼 변환을 위한 정리 함수 구성.	127
MDM Hub 환경 보고서 검토.	128
MDM Hub 환경 보고서 저장.	128
외부 호출 및 응용 프로그램 업그레이드.	129
EJB 프로토콜용 SiperianClient 라이브러리 클래스 업그레이드.	130
MDM Hub 메타데이터 준비.	130
업그레이드 테스트.	131
MDM Hub 업그레이드 테스트.	131
사용자 지정 코드 업그레이드 테스트.	131
프로비저닝 도구 업그레이드 테스트.	131
비즈니스 항목이 있는 Data Director 업그레이드 테스트.	132
제목 영역이 있는 Data Director 업그레이드 테스트.	132
일반 Hub 서버 속성 구성.	132
Data Director 및 Hub 서버 속성.	133
Data Director 글로벌 속성.	134
비즈니스 항목 스키마 생성.	134
장 10: 검색 구성 업그레이드.	135
검색 구성 업그레이드 개요.	135
1단계. Elasticsearch 설치 및 설정.	135

설치 전 태스크 완료.	136
Elasticsearch 설치.	137
Elasticsearch JVM(Java Virtual Machine) 구성.	137
Elasticsearch 속성 파일 구성.	137
Elasticsearch 보호.	138
분석 플러그인 설치.	139
정지 단어, 동의어 및 문자 매핑 구성.	139
Elasticsearch 시작.	139
2단계. 검색에 대한 MDM Hub 속성 구성.	140
검색에 대한 Hub 서버 구성.	140
검색에 대한 처리 서버 구성.	141
3단계. 프로비저닝 도구를 사용하여 검색 구성.	141
Elasticsearch 클러스터 구성.	142
검색 또는 쿼리 결과 표시 구성.	142
4단계. 연산 참조 저장소 유효성 검사.	143
5단계. 검색 데이터 인덱싱.	144
Elasticsearch 버전 6.8.6으로 업그레이드(선택 사항).	144
Elasticsearch 버전 6.8.6으로 업그레이드하기 전 선행 조건.	144
Elasticsearch 속성 파일 구성.	144
Elasticsearch 인덱스 업그레이드.	145
장 11: 계층 업그레이드.	148
계층 업그레이드 개요.	148
계층 관계 및 네트워크 관계 이해.	149
계층 관계.	149
네트워크 관계.	150
계층 관리자 관계 복사 및 계층 관계 생성.	151
계층 관리자에서 관계 복사 및 계층 생성.	152
계층에 대한 Hub 서버 구성.	152
관계 기본 개체를 기본 개체로 변환.	153
계층에 대한 액세스 권한 구성.	153
계층 관리자 관계 복사 및 네트워크 관계 생성.	154
계층 관리자에서 관계를 복사하여 네트워크 관계 추가.	154
관계 기본 개체를 기본 개체로 변환.	154
장 12: 응용 프로그램 서버에 대한 ActiveVOS 설치 후 태스크.	155
응용 프로그램 서버에 대한 ActiveVOS 설치 후 태스크.	155
장 13: 비즈니스 항목 어댑터에 대한 ActiveVOS 업그레이드 후 태스크.	156
비즈니스 항목 어댑터에 대한 ActiveVOS 업그레이드 후 태스크.	156
비즈니스 항목 워크플로우 어댑터의 ActiveVOS URN 구성.	157
ActiveVOS 프로토콜을 HTTPS로 설정.	157

비즈니스 항목에 대한 사용자 지정된 워크플로우 업데이트.	158
비즈니스 항목에 대한 워크플로우에서 프레젠테이션 매개 변수 업데이트.	158
비즈니스 항목에 대한 워크플로우에서 첨부 파일 활성화.	160
ActiveVOS의 MDM ID 서비스 구성.	161
사용자 지정 BeMDMWorkflow 프로젝트(인플레이스 업그레이드).	162
병합 해제 및 병합 워크플로우 트리거 구성(인플레이스 업그레이드).	162
Entity 360 프레임워크 태스크 관리자 추가.	162
장 14: 제목 영역 어댑터에 대한 ActiveVOS 업그레이드 후 태스크.	163
제목 영역 어댑터에 대한 ActiveVOS 업그레이드 후 태스크.	163
ActiveVOS URN 업데이트.	164
ActiveVOS의 트러스트된 사용자 확인.	164
제목 영역을 기반으로 ActiveVOS 워크플로우에 대한 Informatica Data Director 태스크 구성 업데이 이트.	165
제목 영역 기반 ActiveVOS 어댑터에 대한 IDD 구성 업데이트.	165
제목 영역 워크플로우 어댑터의 태스크 트리거 구성.	166
제목 영역에 대한 사용자 지정된 워크플로우 업데이트.	167
제목 영역에 대한 워크플로우에서 프레젠테이션 매개 변수 업데이트.	167
제목 영역에 대한 워크플로우에서 첨부 파일 활성화.	168
제목 영역을 기반으로 하는 ActiveVOS 워크플로우 재배포.	169
비즈니스 항목 및 비즈니스 항목 서비스 구성 파일 생성.	170
부록 A: 업그레이드 프로세스 문제 해결.	171
부록 B: FAQ.	177
부록 C: 기존 ActiveVOS 태스크 처리.	179
기존 ActiveVOS 태스크 처리 개요.	179
마이그레이션 속성.	179
마이그레이션 스크립트 실행 속성 파일 사용.	180
마이그레이션 스크립트 실행 명령줄 속성 사용.	181
부록 D: 메타데이터 캐시 구성.	182
메타데이터 캐시 구성(선택 사항).	182
Infinispan 특성.	182
Infinispan 특성 편집.	183
인덱스.	185

서문

Informatica® *Multidomain MDM 업그레이드 가이드*의 지침에 따라 Multidomain MDM 구현을 최신 버전으로 업그레이드합니다. 업그레이드할 때 현재 설치된 버전에 적용되는 *Multidomain MDM 업그레이드 가이드*를 사용 중인지 확인하십시오.

Informatica 리소스

Informatica는 Informatica Network 및 기타 온라인 포털을 통해 다양한 범위의 제품 리소스를 제공합니다. 리소스를 통해 Informatica 제품 및 솔루션을 최대한 활용하고 다른 Informatica 사용자 및 주제별 전문가로부터 배울 수 있습니다.

Informatica 네트워크

Informatica Network는 Informatica 기술 자료, Informatica 글로벌 고객 지원 센터 등 여러 리소스로 연결되는 관문입니다. Informatica Network를 시작하려면 <https://network.informatica.com>을 방문하십시오.

Informatica Network 멤버인 경우 다음 옵션이 가능합니다.

- 기술 자료에서 제품 리소스를 검색할 수 있습니다.
- 제품 사용 가능 여부에 대한 정보를 봅니다.
- 지원 사례를 생성하고 검토할 수 있습니다.
- 거주 지역의 Informatica 사용자 그룹 네트워크를 검색하고 동료와 협업 관계 유지

Informatica 기술 자료

Informatica 기술 자료를 사용하여 사용 방법 문서, 모범 사례, 비디오 자습서, 자주 묻는 질문에 대한 답변 등 제품 리소스를 확인할 수 있습니다.

기술 자료를 검색하려면 <https://search.informatica.com>을 방문하십시오. 기술 자료에 대한 질문, 의견 또는 아이디어가 있는 경우 KB_Feedback@informatica.com을 통해 Informatica 기술 자료 팀에 문의해 주시기 바랍니다.

Informatica 설명서

Informatica 설명서 포털에서 확장된 설명서 라이브러리를 탐색하여 현재 및 최근 제품 릴리스를 확인할 수 있습니다. 설명서 포털을 탐색하려면 <https://docs.informatica.com>을 방문하십시오.

제품 설명서에 대한 질문, 의견 또는 아이디어가 있는 경우 infa_documentation@informatica.com에서 Informatica 설명서 팀에 문의해 주시기 바랍니다.

Informatica Product Availability Matrix

PAM(Product Availability Matrix)은 제품 릴리스에서 지원하는 운영 체제 버전, 데이터베이스 및 데이터 소스 유형과 대상을 나타냅니다.

<https://network.informatica.com/community/informatica-network/product-availability-matrices>에서 Informatica PAM을 찾을 수 있습니다.

Informatica Velocity

Informatica Velocity는 수백 가지 데이터 관리 프로젝트의 실제 경험을 토대로 Informatica 전문 서비스업에서 개발한 팁과 모범 사례 모음입니다. Informatica Velocity는 전 세계의 조직과 협력하여 성공적인 데이터 관리 솔루션을 계획, 개발, 배포 및 유지 관리하는 Informatica 컨설턴트의 포괄적인 지식을 보여줍니다.

Informatica Velocity 리소스는 <http://velocity.informatica.com>에서 확인할 수 있습니다. Informatica Velocity에 대한 질문, 주석 또는 아이디어가 있으시면 Informatica 전문 서비스업(ips@informatica.com)에 문의하십시오.

Informatica Marketplace

Informatica Marketplace는 Informatica 구현을 확대 및 개선하기 위한 솔루션을 찾을 수 있는 포럼입니다. Marketplace에서 Informatica 개발자와 파트너가 제공하는 수백 개의 솔루션을 활용하여 생산성을 향상시키고 프로젝트의 구현에 걸리는 시간을 줄일 수 있습니다. <https://marketplace.informatica.com>에서 Informatica Marketplace를 찾을 수 있습니다.

Informatica 글로벌 고객 지원 센터

전화 또는 Informatica Network를 통해 글로벌 지원 센터에 문의할 수 있습니다.

해당 지역의 Informatica 글로벌 고객 지원 전화 번호는 Informatica 웹 사이트 (<https://www.informatica.com/services-and-training/customer-success-services/contact-us.html>)를 방문하여 찾을 수 있습니다.

Informatica Network에서 온라인 지원 리소스를 찾으려면 <https://network.informatica.com>을 방문하고 eSupport 옵션을 선택하십시오.

제 1 장

업그레이드 개요

이 장에 포함된 항목:

- [소개, 11](#)
- [업그레이드 프로세스 선택, 12](#)
- [업그레이드 문서화, 13](#)

소개

Informatica Multidomain MDM을 업그레이드해 주셔서 감사합니다. 설치된 버전을 이 Multidomain MDM 버전으로 직접 업그레이드할 수 있습니다.

중요: 모든 MDM 구성 요소를 Multidomain MDM의 동일한 버전으로 업그레이드해야 합니다.

Multidomain MDM 환경에는 개발 환경, 테스트 환경 및 프로덕션 환경이 포함될 수 있습니다. 이러한 각 환경을 업그레이드해야 합니다. 모범 사례는 개발 환경을 먼저 업그레이드 하는 것입니다. 업그레이드 문제가 있으면 식별하고 해결하십시오. 개발 환경을 성공적으로 업그레이드한 후에는 높은 확신 하에 테스트 및 프로덕션 환경을 업그레이드할 수 있습니다.

시작하기 전에

시작하기 전에 필요한 가이드가 있는지 확인하고 **Product Availability Matrix**를 검토하십시오.

- 설치된 버전의 Multidomain MDM에 해당하는 올바른 업그레이드 가이드가 있는지 확인합니다. 제목 페이지의 *버전 x에서 업그레이드* 제목에서 주 버전 번호를 확인합니다. 이 버전 번호가 설치된 제품의 주 버전 번호와 일치하는지 확인합니다. 설치된 제품의 버전 번호를 찾으려면 MDM Hub 콘솔에서 **도움말 > 정보**를 클릭합니다. **설치 세부 정보**를 클릭합니다. 버전 번호가 릴리스 이름 옆에 나타납니다. 업그레이드에서는 설치된 제품에 핫픽스가 적용되었는지 여부가 중요하지 않습니다.
- 해당 데이터베이스 및 응용 프로그램 서버에 적용되는 **Multidomain MDM 설치 가이드**를 다운로드합니다. 원활한 업그레이드를 위해서는 설치 가이드에 설명된 일부 태스크를 수행해야 합니다.
- **Product Availability Matrix**를 검토하여 이 Multidomain MDM 버전의 시스템 요구 사항 변경에 대한 자세한 내용을 확인하십시오. 매트릭스는 Informatica 네트워크:

<https://network.informatica.com/community/informatica-network/product-availability-matrices>에서 찾을 수 있습니다.

업그레이드 프로세스 선택

다음 옵션을 사용하여 Multidomain MDM 업그레이드를 수행할 수 있습니다.

클린 업그레이드

클린 업그레이드에서는 현재 MDM 시스템 요구 사항을 충족하는 시스템을 사용하여 새 환경을 준비합니다. MDM Hub 마스터 데이터베이스 및 연산 참조 저장소를 기존 환경에서 복제하고, 복제한 데이터베이스를 새 환경에 복사하고, Hub 저장소 업그레이드를 실행합니다. *설치 가이드*의 지침에 따라 새 환경에서 Hub 서버 및 처리 서버를 설치합니다. 인프라 계획에 대한 자세한 내용은 *Multidomain MDM 인프라 계획 가이드*를 참조하십시오.

인플레이스 업그레이드

인플레이스 업그레이드 경우 Multidomain MDM를 실행하는 시스템을 현재 MDM 시스템 요구 사항과 호환되도록 업데이트합니다. 그런 다음 MDM 구성 요소를 업그레이드합니다.

업그레이드 프로세스 선택 지침

업그레이드 유형에 따라 선택할 수 있는 업그레이드 유형이 결정됩니다.

다음 테이블에는 업그레이드 유형이 나열되고, 사용할 수 있는 업그레이드 프로세스가 설명되어 있습니다.

업그레이드 유형 Multidomain MDM	클린 업그레이드	인플레이스 업그레이드
임의 MDM 버전으로 업그레이드(응용 프로그램 서버를 주 버전으로 업데이트하여 MDM 시스템 요구 사항을 충족해야 함)	예	아니요
임의 MDM 버전으로 업그레이드(응용 프로그램 서버를 주 버전으로 업데이트하여 MDM 시스템 요구 사항을 충족하지 않아도 됨)	예	예

클린 업그레이드에 대한 업그레이드 프로세스

클린 업그레이드에 대한 프로세스는 다음 단계로 구성됩니다.

단계	태스크	지침의 위치
1	데이터베이스 태스크	업그레이드 가이드
2	응용 프로그램 서버 태스크	업그레이드 가이드
3	업그레이드 전 태스크	업그레이드 가이드
4	Hub 저장소를 백업 및 복제한 후 Hub 저장소를 새 시스템에 복사	이 태스크는 DBA를 통해 수행합니다.
5	Hub 저장소 업그레이드	업그레이드 가이드
6	Hub 서버 설치 및 설치 후 태스크	설치 가이드
7	처리 서버 설치 및 설치 후 태스크	설치 가이드
8	업그레이드 후 태스크	업그레이드 가이드

단계	태스크	지침의 위치
9	응용 프로그램 서버에 대한 ActiveVOS 설치 후 태스크	설치 가이드
10	비즈니스 항목 어댑터에 대한 ActiveVOS 업그레이드 후 태스크	업그레이드 가이드
11	제목 영역 어댑터에 대한 ActiveVOS 업그레이드 후 태스크	업그레이드 가이드
12	리소스 키트 설치 태스크 및 리소스 키트 설치 후 태스크(해당하는 경우)	설치 가이드

인플레이스 업그레이드의 업그레이드 프로세스

인플레이스 업그레이드에 대한 프로세스는 다음 단계로 구성됩니다.

단계	태스크	지침의 위치
1	데이터베이스 태스크	업그레이드 가이드
2	응용 프로그램 서버 태스크	업그레이드 가이드
3	업그레이드 전 태스크	업그레이드 가이드
4	Hub 저장소 업그레이드	업그레이드 가이드
5	Hub 서버 업그레이드	업그레이드 가이드
6	처리 서버 업그레이드	업그레이드 가이드
7	리소스 키트 업그레이드	업그레이드 가이드
8	업그레이드 후 태스크	업그레이드 가이드
9	응용 프로그램 서버에 대한 ActiveVOS 설치 후 태스크	설치 가이드
10	비즈니스 항목 어댑터에 대한 ActiveVOS 설치 후 태스크	업그레이드 가이드
11	제목 영역 어댑터에 대한 ActiveVOS 설치 후 태스크	업그레이드 가이드

* 설치된 버전과 현재 버전 사이에 여러 개의 릴리스가 있는 경우 모범 사례는 클린 업그레이드를 수행하거나, 모든 설치 전 태스크를 검토하여 사용하는 데이터베이스 또는 응용 프로그램 서버에 발생한 모든 변경 사항을 확인하는 것입니다.

업그레이드 문서화

확인하고 필요한 경우 업그레이드 문제를 해결하기 위해 업그레이드하기 전, 업그레이드하는 동안, 업그레이드한 후 Informatica MDM Hub 환경의 세부 정보를 캡처해야 합니다. 업그레이드 문제 해결과 관련하여 도움이 필요한 경우 이 정보의 사본을 Informatica 글로벌 고객 지원 센터에 제공할 수 있습니다.

다음 테이블에는 업그레이드 정보 출처가 나와 있습니다.

업그레이드 정보	정보 출처
ORS(연산 참조 저장소) 메타데이터	Hub 콘솔의 리포지토리 관리자 도구에서 생성된 유효성 검사 결과 업그레이드 전후에 ORS 메타데이터에 대한 유효성 검사 결과를 생성합니다.
구성 설정	Hub 콘솔의 엔터프라이즈 관리자 도구에서 생성된 환경 보고서 환경 보고서는 Hub 서버, 처리 서버, 마스터 데이터베이스 및 ORS 데이터베이스 정보를 문서화합니다. 업그레이드 전후에 환경 보고서를 저장합니다.
Hub 서버 및 처리 서버 업그레이드에 대한 이벤트 업그레이드	Informatica MDM Hub 설치 프로그램 로그 파일
Hub 서버 및 처리 서버 정보	Hub 서버 및 처리 서버 로그 파일
Hub 콘솔 정보	Hub 콘솔 로그 파일

관련 항목:

- [“MDM Hub 환경 보고서 저장” 페이지 128](#)

제 2 장

업그레이드 전 태스크

이 장에 포함된 항목:

- [업그레이드 전 태스크, 15](#)
- [업그레이드 준비, 15](#)
- [환경 준비, 16](#)
- [기존 작업 처리, 20](#)
- [MDM Hub 구성 업데이트, 20](#)
- [BPM 업그레이드 준비, 23](#)

업그레이드 전 태스크

클린 업그레이드 또는 인플레이스 업그레이드를 수행하는 경우 업그레이드 전 태스크를 수행하여 환경이 올바르게 구성되었는지 확인해야 합니다.

업그레이드 준비

다음 태스크를 수행하여 업그레이드를 준비합니다.

업그레이드 태스크	세부 정보
릴리스 정보 읽기	릴리스 정보에는 설치 및 업그레이드 프로세스 업데이트에 대한 정보가 들어 있습니다. 중요: 일부 버전의 응용 프로그램 서버 및 데이터베이스에는 Multidomain MDM을 실행할 때 알려진 제한이 있습니다. 제안되는 모든 해결 방법을 수행하십시오.
릴리스 가이드 읽기	릴리스 가이드에는 새 기능 및 변경된 동작에 대한 정보가 들어 있습니다.
최신 라이선스 파일 열기	Multidomain MDM에 대한 업그레이드 소프트웨어를 요청할 때 최신 라이선스 파일을 요청하십시오.

업그레이드 태스크	세부 정보
업그레이드 제한 사항 검토	<p>업그레이드를 시작하기 전에 다음 조건을 검토합니다.</p> <ul style="list-style-type: none"> - Multidomain MDM 구현의 모든 구성 요소는 버전이 같아야 합니다. 여러 버전의 Multidomain MDM이 있는 경우 각 버전은 별도의 환경에 설치되어야 합니다. 서로 다른 버전의 Multidomain MDM을 동일한 환경에 설치할 수 없습니다. - Multidomain MDM을 업그레이드하는 동안에는 환경을 크게 변경하면 안 됩니다. 예를 들어 업그레이드 프로세스 동안 Oracle, IBM Db2 또는 Microsoft SQL Server 데이터베이스를 업그레이드하면 안 됩니다. - 제공된 업그레이드 스크립트를 사용하여 ORS(연산 참조 저장소)를 업그레이드해야 합니다. 리포지토리 관리자는 업그레이드 도구로 사용하면 안 됩니다. 아티팩트 중 일부가 전달되지 않을 수도 있고 한 버전에서 다른 버전으로 잘못 전달될 수도 있기 때문입니다. 자세한 내용은 <i>Multidomain MDM 릴리스 노트</i>를 참조하십시오.
업그레이드 기록 폴더 생성	메타데이터 유효성 검사 결과, 환경 보고서 및 로그 파일과 같은 모든 업그레이드 아티팩트의 사본을 저장할, 이름이 <i>upgradedoc</i> 인 폴더를 생성합니다. 업그레이드 중에 문제가 발생한 경우 분석을 위해 이 디렉터리의 사본을 Informatica 글로벌 고객 지원 센터에 보내야 합니다.
MDM Hub 환경 보고서 검토	<p>Hub 콘솔에서 엔터프라이즈 관리자 도구를 사용하여 Hub 서버, 처리 서버, MDM Hub 마스터 데이터베이스 및 연산 참조 저장소 데이터베이스에 대한 현재 MDM Hub 구성을 검토합니다. 또한 구성 요소의 버전 기록을 참고합니다.</p> <p>업그레이드 기록 폴더에 환경 보고서의 사본을 저장합니다.</p>

환경 준비

선택한 업그레이드 프로세스에 대한 태스크를 수행합니다.

클린 업그레이드

클린 업그레이드에 대한 다음 태스크를 수행합니다.

태스크	설명
최소 시스템 요구 사항 확인	<p>시스템이 MDM Hub 설치를 위한 하드웨어 및 소프트웨어 요구 사항을 충족하는지 확인합니다. 하드웨어 요구 사항은 데이터, 처리 볼륨 및 비즈니스 규칙에 따라 다릅니다.</p> <p>MDM Hub를 설치하려면 시스템이 다음과 같은 최소 요구 사항을 충족해야 합니다.</p> <ul style="list-style-type: none"> - 디스크 공간. 4.9GB - 개발 환경을 위한 RAM. 4GB <p>MDM Hub 구성 요소의 런타임 실제 메모리 요구 사항을 확인하려면 다음 수식을 사용합니다.</p> <p>Total run-time memory requirement for MDM Hub components = JDK JVM max heap size of the application server + operating system native heap size</p>
JDK(Java 개발 키트) 설치	<p>MDM Hub를 설치할 시스템에 지원되는 버전의 JDK를 설치합니다. JDK는 MDM Hub 설치 프로그램과 함께 번들로 제공되지 않습니다.</p> <p>JBoss 환경에서 Informatica가 제공하는 Azul Zulu JDK의 버전을 설치해야 합니다. Azul Zulu JDK를 다운로드하려면 Informatica Shipping에 문의하십시오.</p> <p>참고: Hub 콘솔을 실행할 시스템 및 응용 프로그램 서버 시스템에 동일한 Java 버전을 사용합니다.</p>

태스크	설명
Visual Studio 2015용 Visual C++ 재배포 가능 패키지를 설치 Windows만 해당	Windows 시스템에서 Multidomain MDM를 사용하려면 Visual Studio 2015용 Visual C++ 재배포 가능 패키지를 통해 이름 검색 기능 및 일치 기능을 지원해야 합니다.
환경 변수 설정	<p>MDM Hub 설치를 위한 환경 변수를 설정합니다.</p> <p>올바른 JDK를 사용하려면 JDK 디렉터리를 가리키도록 다음과 같은 환경 변수를 설정합니다.</p> <ul style="list-style-type: none"> - JAVA_HOME. 필수 - PATH. 필수 <p>Oracle에 대한 올바른 로컬 동작을 설정하려면 Oracle 로더와 같은 클라이언트 응용 프로그램 환경 및 MDM Hub 구성 요소에 대한 NLS_LANG 환경 변수를 설정합니다.</p> <p>다음 형식으로 NLS_LANG 매개 변수를 지정합니다.</p> <p>NLS_LANG = <language>.<territory>.<character set></p> <p>참고: 중국어, 일본어, 한국어 또는 액센트가 있는 문자가 포함된 레코드를 저장 및 검색하려면 문자 집합을 UTF-8로 설정합니다.</p> <p>NLS_LANG 설정에 대한 자세한 내용은 Oracle 설명서를 참조하십시오.</p>
운영 체제 로컬 설정	Hub 서버, MDM Hub 마스터 데이터베이스, 연산 참조 저장소 및 Hub 콘솔에 대한 동일한 운영 체제 로컬을 설정합니다.
UNIX에서 X Window System 설치	UNIX에서 그래픽 모드로 설치 프로그램을 실행할 경우 X Window System을 설치합니다. X Window System은 그래픽 디스플레이 서버입니다. X Window System 설치에 대한 자세한 내용은 운영 체제용 설명서를 참조하십시오.
응용 프로그램 서버의 루트 시작 페이지에 대한 액세스 비활성화	보안을 개선하려면 응용 프로그램 서버의 루트 시작 페이지에 대한 액세스를 비활성화합니다. 지침은 응용 프로그램 서버에 대한 설명서를 참조하십시오.
보안되지 않은 TLS 암호 그룹 비활성화	<p>보안을 개선하려면 Multidomain MDM을 사용하는 Java Runtime Environment에서 TLS 암호 그룹을 비활성화합니다.</p> <ol style="list-style-type: none"> 1. 다음 파일을 엽니다. ../jdk<버전>/jre/lib/security/java.security 2. jdk.tls.disabledAlgorithms 속성을 찾고 다음의 보안되지 않은 암호 그룹 목록을 포함하도록 값을 업데이트합니다. <p>jdk.tls.disabledAlgorithms = SSLv3, RC4, MD5withRSA, DH keySize < 1024, EC keySize < 224, DES40_CBC, RC4_40, 3DES_EDE_CBC, EDH-RSA-DES-CBC3-SHA, ECDHE-RSA-DES-CBC3-SHA, DES-CBC3-SHA</p> <p>속성에 대한 자세한 내용은 JDK 설명서를 참조하십시오.</p>

클린 업그레이드에 대한 다음 추가 태스크를 수행합니다.

태스크	세부 정보
메타데이터 유효성 검사	<p>ORS(연산 참조 저장소)에 유효성 검사 오류가 없는지 확인합니다. ORS에 올바르지 않은 메타데이터가 포함되어 있을 때 Hub 저장소를 업그레이드하면 업그레이드 중에 예기치 않은 결과가 발생할 수 있습니다. Hub 콘솔에서 리포지토리 관리자를 사용하여 메타데이터의 유효성을 검사합니다. 유효성 검사 오류를 해결한 다음 메타데이터 유효성을 다시 검사하여 유효성 검사 오류가 해결되었는지 확인합니다.</p> <p>upgradedoc 업그레이드 기록 폴더에 최종 유효성 검사 결과의 사본을 저장합니다.</p> <p>MDM Hub 콘솔에서 리포지토리 관리자 도구를 사용하여 유효성을 검사하고 유효성 검사 결과를 저장합니다.</p>
지속형 ID 구현 업데이트	<p>지속형 ID를 사용하는 경우 Informatica 글로벌 고객 지원 센터에 문의하십시오. 업그레이드된 버전의 Multidomain MDM과 호환되도록 지속형 ID 구현을 업데이트해야 합니다.</p>

인플레이스 업그레이드

인플레이스 업그레이드에 대한 다음 태스크를 수행합니다.

태스크	세부 정보
시스템 요구 사항을 충족하도록 MDM 환경 업데이트	<p>운영 체제, 응용 프로그램 서버, JDK 및 데이터베이스 서버를 업데이트해야 할 수 있습니다. 시스템 요구 사항은 Informatica Network에서 이 버전의 Multidomain MDM(MDM)에 대한 PAM(Product Availability Matrix)https://network.informatica.com/community/informatica-network/product-availability-matrices/overview 을 참조하십시오.</p> <p>JBoss 환경에서 Informatica가 제공하는 Azul Zulu JDK의 버전을 설치해야 합니다. Azul Zulu JDK를 다운로드하려면 Informatica Shipping에 문의하십시오.</p> <p>MDM을 실행하는 시스템을 현재 MDM 시스템 요구 사항과 호환되도록 업데이트합니다.</p>
Visual Studio 2015용 Visual C++ 재배포 가능 패키지 설치(Windows 만 해당)	<p>Windows 시스템에서 Multidomain MDM를 사용하려면 Visual Studio 2015용 Visual C++ 재배포 가능 패키지를 통해 이력 검색 기능 및 일치 기능을 지원해야 합니다.</p>
메타데이터 유효성 검사	<p>ORS(연산 참조 저장소)에 유효성 검사 오류가 없는지 확인합니다. ORS에 올바르지 않은 메타데이터가 포함되어 있을 때 Hub 저장소를 업그레이드하면 업그레이드 중에 예기치 않은 결과가 발생할 수 있습니다. Hub 콘솔에서 리포지토리 관리자를 사용하여 메타데이터의 유효성을 검사합니다. 유효성 검사 오류를 해결한 다음 메타데이터 유효성을 다시 검사하여 유효성 검사 오류가 해결되었는지 확인합니다.</p> <p>upgradedoc 업그레이드 기록 폴더에 최종 유효성 검사 결과의 사본을 저장합니다.</p> <p>MDM Hub 콘솔에서 리포지토리 관리자 도구를 사용하여 유효성을 검사하고 유효성 검사 결과를 저장합니다.</p>
지속형 ID 구현 업데이트	<p>지속형 ID를 사용하는 경우 Informatica 글로벌 고객 지원 센터에 문의하십시오. 업그레이드된 버전의 Multidomain MDM과 호환되도록 지속형 ID 구현을 업데이트해야 합니다.</p>

태스크	세부 정보
구현 백업	<p>사용자 지정을 유지하고 환경을 복원할 수 있도록 현재 구현을 백업합니다. 업그레이드 동안 문제가 발생한 경우 백업에서 환경을 복원할 수 있습니다.</p> <p>스키마 백업</p> <p>스키마에 대한 전체 백업을 수행합니다. 업그레이드 프로세스에서 나타나는 스키마 변경 사항은 롤백할 수 없습니다. 업그레이드 문제가 발생한 경우 백업에서 스키마를 복원할 수 있습니다. 스키마를 백업하려면 데이터베이스의 설명서를 참조하십시오.</p> <p>Hub 서버 및 처리 서버가 설치된 디렉터리 백업</p> <p>Hub 서버와 처리 서버를 설치하기 전에 Hub 서버와 처리 서버 설치 폴더를 현재 환경에 백업하십시오.</p> <p>Data Director 응용 프로그램 백업</p> <p>제목 영역이 있는 Data Director 응용 프로그램의 내보내기에 대한 자세한 내용은 <i>Multidomain MDM Data Director 구현 가이드</i>를 참조하십시오.</p> <p>사용자 지정 등록 또는 백업</p> <p>사용자 지정 쿼리, 사용자 지정 함수 및 사용자 종료와 같이 Hub 콘솔에서 등록하는 사용자 지정은 업그레이드 프로세스 중에 유지됩니다.</p> <p>Hub 콘솔에서 등록하지 않는 사용자 지정의 경우 최신 소스 코드를 백업합니다. 등록되지 않은 사용자 지정은 업그레이드 후에 사용하지 못할 수 있습니다.</p> <p>사용자 지정된 정리 엔진 구성 파일 백업</p> <p>사용자 지정된 정리 엔진 구성 파일을 모두 백업합니다.</p>
JAVA_HOME 환경 변수 구성	<p>JBoss 또는 WebLogic. 기존 설치 디렉터리에 업그레이드를 적용하는 경우 다음 단계를 수행합니다.</p> <ol style="list-style-type: none"> JBoss 환경의 다음 파일에서 JAVA_HOME 환경 변수 설정을 수동으로 제거합니다. <ul style="list-style-type: none"> Windows: <MDM Hub 설치 디렉터리>\setSiperianEnv.bat UNIX: <MDM Hub 설치 디렉터리>/setSiperianEnv.sh JBoss 또는 WebLogic 환경에서 JAVA_HOME 환경 변수를 지원되는 JDK 버전의 경로로 설정합니다.
응용 프로그램 서버의 루트 시작 페이지에 대한 액세스 비활성화	<p>보안을 개선하려면 응용 프로그램 서버의 루트 시작 페이지에 대한 액세스를 비활성화합니다. 지침은 응용 프로그램 서버에 대한 설명서를 참조하십시오.</p>

기존 작업 처리

다음 태스크를 수행하여 기존 작업을 처리합니다.

업그레이드 태스크	세부 정보
레코드가 포함된 준비 테이블에서 로드 작업 실행	Microsoft SQL Server 환경에서 레코드가 포함된 준비 테이블에서 로드 일괄 작업을 실행합니다. 준비 테이블에 레코드가 포함된 경우 연산 참조 저장소를 업그레이드하면 로그 파일 크기가 사용 가능한 하드 드라이브 공간을 초과할 수 있기 때문에 업그레이드가 실패할 수 있습니다.
준비 작업 완료 및 준비 테이블 콘텐츠 삭제	연산 참조 저장소를 업그레이드하기 전에 진행 중인 모든 준비 작업을 완료하고 준비 테이블의 콘텐츠를 삭제합니다. 준비 테이블의 콘텐츠를 삭제하지 않으면 Microsoft SQL Server 환경에서 연산 참조 저장소 업그레이드를 완료하는 데 시간이 예상보다 오래 걸립니다.

MDM Hub 구성 업데이트

다음 태스크를 수행하여 MDM Hub 구성을 업데이트합니다.

업그레이드 태스크	세부 정보
ORS 사용자에게 SYS.V_\$PARAMETER에 대한 선택 권한 부여	특정 권한을 부여하려면 다음 SQL 문을 실행합니다. <code>grant select on SYS.V_\$PARAMETER to <Operational Reference Store user>;</code>
제목 영역 구성에서 이름의 특수 문자 제거	프로비저닝 도구에서는 이름에 특수 문자를 사용할 수 없습니다. 제목 영역 구성에서 비즈니스 항목 스키마를 생성하려면 먼저 제목 영역 구성에 있는 이름에서 모든 특수 문자를 제거해야 합니다.
열 이름에 예약어가 포함되지 않는지 확인	최신 버전의 Multidomain MDM에서는 추가 키워드가 예약어로 표시되었습니다. 예약된 키워드를 구현에 사용하지 마십시오. 예약어의 전체 목록은 <i>Multidomain MDM 구성 가이드</i> 의 "스키마 개체를 정의하기 위한 요구 사항" 섹션을 참조하십시오. 데이터가 포함된 열의 이름을 변경하는 스크립트를 요청하려면 Informatica 글로벌 고객 지원 센터에 문의하십시오.
매핑된 열에 호환되는 데이터 유형이 있는지 확인	Oracle 데이터베이스를 연산 참조 저장소로 사용하는 경우에는 랜딩 테이블에서 준비 테이블로의 매핑을 확인하여 매핑된 열의 데이터 유형이 호환되는지 확인해야 합니다. 이전 버전에서는 매핑된 열의 데이터 유형이 호환되지 않는 경우에도 랜딩 테이블에서 준비 테이블로의 매핑을 작성할 수 있습니다. 이 버전에서는 매핑된 열의 데이터 유형이 호환되어야 합니다. 데이터를 로드할 때 데이터 유형이 일치하지 않으면 로드가 실패하고 로그 파일에 오류가 기록됩니다. 열 매핑에 대한 자세한 내용은 <i>Multidomain MDM 구성 가이드</i> 를 참조하십시오.

업그레이드 태스크	세부 정보
C_REPOS_DS_PREF 테이블 및 C_REPOS_DS_PREF_DETAIL 테이블에서 사용자 기본 설정 백업	Data Director 환경에 표시되거나 숨겨진 열에 대한 사용자 기본 설정이 포함되는 경우 이 버전의 암호화 해시 알고리즘 변경으로 인해 업그레이드 시 설정이 손실됩니다. 업그레이드 후 사용자 기본 설정을 다시 생성하십시오. Data Director 글로벌 속성에 대한 자세한 내용은 <i>Multidomain MDM Data Director 구현 가이드</i> 를 참조하십시오.
사용자에게 유효한 이메일 주소가 있는지 확인	Hub 콘솔에서 기존 사용자에게 사용자 이름과 연결된 유효한 전자 메일 주소가 있는지 확인합니다. MDM Hub에 액세스하기 위해 암호를 재설정하려면 사용자에게 유효한 전자 메일 주소가 있어야 합니다. Hub 콘솔의 사용자 정보 편집에 대한 자세한 내용은 <i>Multidomain MDM 보안 가이드</i> 를 참조하십시오. 참고: Hub 콘솔에서 admin 사용자의 전자 메일 주소를 변경할 수 없습니다. admin 사용자의 전자 메일 주소를 변경하려면 CMX_SYSTEM 스키마의 C_REPOS_USER 테이블에서 admin 사용자 항목을 직접 업데이트합니다.
인덱스 등록	사용자 지정 인덱스가 있는 스키마를 업그레이드하는 경우에는 사용자 지정 인덱스를 등록해야 합니다. RegisterCustomIndex SIF API를 사용하여 사용자 지정 인덱스를 등록합니다. RegisterCustomIndex SIF API에 대한 자세한 내용은 <i>Multidomain MDM 서비스 통합 프레임워크 가이드</i> 를 참조하십시오.

원래 스키마 소유자에 연산 참조 저장소를 등록합니다.

ORS(연산 참조 저장소) 데이터베이스가 프록시 사용자에게 등록된 경우, 원래 스키마 소유자에 이 ORS 데이터베이스를 등록합니다. 원래 스키마 소유자에 ORS를 등록하여 업그레이드 작업을 수행할 만한 권한이 있는지 확인해야 합니다. 업그레이드 후 프록시 사용자에게 ORS를 재등록할 수 있습니다. 원래 기본 스키마 소유자는 CMX_ORS입니다.

1. Hub 콘솔을 시작합니다.
데이터베이스 변경 대화 상자가 표시됩니다.
2. MDM Hub 마스터 데이터베이스를 선택하고 **연결**을 클릭합니다.
3. 데이터베이스 도구를 구성 작업 영역에서 선택합니다.
4. 쓰기 잠금 > 잠금 획득을 선택합니다.
5. 데이터베이스 등록 단추를 클릭합니다.
Informatica MDM Hub 연결 마법사가 표시되어 데이터베이스 유형을 선택하라는 메시지를 표시합니다.
6. 데이터베이스 유형을 선택하고 **다음**을 클릭합니다.

7. 데이터베이스의 연결 속성을 구성합니다.

a. Oracle 연결 방법을 선택한 다음 **다음**을 클릭합니다.

다음 Oracle 연결 방법을 선택할 수 있습니다.

서비스

서비스 이름을 사용하여 Oracle에 연결합니다.

SID

Oracle 시스템 ID를 사용하여 Oracle에 연결합니다.

SERVICE 및 SID 이름에 대한 자세한 내용은 Oracle 설명서를 참조하십시오.

연결 속성 페이지가 표시됩니다.

b. 선택한 연결 유형의 연결 속성을 지정하고 **다음**을 클릭합니다.

다음 연결 속성을 구성할 수 있습니다.

데이터베이스 표시 이름

Hub 콘솔에 표시되어야 하는 연산 참조 저장소의 이름입니다.

시스템 식별자

Hub 저장소 인스턴스에서 레코드를 고유하게 식별하기 위해 키에 지정되는 접두사입니다.

데이터베이스 호스트 이름

Oracle 데이터베이스를 호스팅하는 서버의 IP 주소 또는 이름입니다.

SID

서버에서 실행 중인 Oracle 데이터베이스의 인스턴스를 나타내는 Oracle 시스템 식별자입니다.

SID 필드는 **SID** 연결 유형을 선택한 경우에 표시됩니다.

서비스

Oracle 데이터베이스에 연결하는 데 사용되는 Oracle 서비스의 이름입니다. **서비스** 필드는 **서비스** 연결 유형을 선택한 경우에 표시됩니다.

포트

Oracle 데이터베이스 서버에서 실행 중인 Oracle 수신기의 TCP 포트입니다. 기본값은 1521입니다.

Oracle TNS 이름

응용 프로그램 서버의 TNSNAMES.ORA 파일에 정의된 대로 네트워크에서 데이터베이스로 알려진 이름입니다.

예를 들면 mydatabase.mycompany.com입니다.

Oracle 데이터베이스를 설치할 때 Oracle TNS 이름을 설정합니다. Oracle TNS 이름에 대한 자세한 내용은 Oracle 설명서를 참조하십시오.

스키마 이름

연산 참조 저장소의 이름입니다.

사용자 이름

ORS의 원래 사용자 이름을 지정합니다. 기본값은 CMX_ORS입니다.

암호

ORS의 원래 사용자와 연결된 암호입니다.

참고: cmxserver.properties 파일에서
cmx.server.database.authentication.method=windowsauthentication인 경우 기본 암호인 ChangeMe가
*****로 표시됩니다. 응용 프로그램은 Windows 인증을 사용하여 데이터 소스에 연결합니다.

요약 페이지가 표시됩니다.

- c. 요약을 검토하고, 추가 연결 속성을 지정합니다.

다음 테이블은 구성할 수 있는 추가 연결 속성을 설명합니다.

속성	설명
연결 URL	연결 URL입니다. 연결 마법사가 기본적으로 연결 URL을 생성합니다. 다음 목록에는 Oracle 연결 유형에 대한 연결 URL 형식이 나와 있습니다. 서비스 연결 유형 jdbc:oracle:thin:@//database_host:port/service_name SID 연결 유형 jdbc:oracle:thin:@//database_host:port:sid 서비스 연결 유형의 경우 다른 연결 URL을 사용자 지정하고 나중에 테스트할 수 있습니다.
등록 후 데이터 소스 작성	등록 후 응용 프로그램 서버에 데이터 소스를 작성하려면 선택합니다. 참고: 해당 옵션을 선택하지 않는 경우에는 데이터 소스를 수동으로 구성해야 합니다.

- d. 서비스 연결 유형의 경우 기본 URL을 변경하려면 **편집** 단추를 클릭하고 URL을 지정한 다음 **확인**을 클릭합니다.

8. **마침**을 클릭합니다.

데이터베이스 등록 대화 상자가 표시됩니다.

9. **확인**을 클릭합니다.

MDM Hub이 ORS를 등록합니다.

BPM 업그레이드 준비

비즈니스 프로세스 관리 시스템을 업그레이드하려면 우선 적절한 워크플로우 어댑터를 선택합니다. 독립 실행형 ActiveVOS를 사용하는 경우 포함된 ActiveVOS로 마이그레이션하고 build.properties 파일을 편집해야 합니다. 환경에 ActiveVOS가 설치된 경우 업그레이드 태스크에 따라 build.properties 파일을 편집합니다. Informatica는 ActiveVOS 9.2.4.6만 지원합니다.

워크플로우 어댑터 선택

워크플로우 어댑터에 대한 다음 업그레이드 옵션을 검토하고 요구 사항에 적합한 업그레이드 옵션을 선택합니다.

다음 테이블에는 현재 워크플로우 어댑터 및 해당 업그레이드 옵션이 설명되어 있습니다.

현재 워크플로우 어댑터	업그레이드 옵션
SIPERIAN	<p>더 이상 사용되지 않습니다.</p> <p>옵션 1</p> <p>SIPERIAN을 기본 워크플로우 어댑터로 유지합니다.</p> <p>옵션 2</p> <p>비즈니스 항목에 기반한 BE ActiveVOS 워크플로우 어댑터로 업그레이드합니다. BE ActiveVOS 워크플로우 어댑터를 사용하려면 비즈니스 항목 구성이 있어야 합니다.</p> <p>자세한 내용은 <i>비즈니스 항목 데이터 모델로 IDD 응용 프로그램 마이그레이션</i>을 참조하십시오.</p>
Informatica ActiveVOS	<p>이 워크플로우 어댑터는 제목 영역에 기반하며 SIF API를 사용하여 작동합니다.</p> <p>옵션 1</p> <p>Informatica ActiveVOS를 기본 워크플로우 어댑터로 유지합니다.</p> <p>옵션 2</p> <p>비즈니스 항목에 기반한 BE ActiveVOS 워크플로우 어댑터로 업그레이드합니다. BE ActiveVOS 워크플로우 어댑터를 사용하려면 비즈니스 항목 구성이 있어야 합니다.</p> <p>자세한 내용은 <i>Multidomain MDM Data Director 마이그레이션 가이드</i>를 참조하십시오.</p>
BE ActiveVOS	<p>이 워크플로우 어댑터는 비즈니스 항목에 기반하며 비즈니스 항목 서비스를 사용하여 작동합니다.</p> <p>Product Availability Matrix에 정의된 버전의 ActiveVOS를 사용해야 합니다. 환경에서 이전 버전이 검색되면 업그레이드 프로세스에서 필요한 버전의 ActiveVOS가 설치됩니다.</p>

독립 실행형 ActiveVOS를 포함한 ActiveVOS로 마이그레이션(인플레이스 업그레이드)

독립 실행형 ActiveVOS를 사용하는 경우 포함된 ActiveVOS로 마이그레이션해야 합니다. 독립 실행형 ActiveVOS는 지원되지 않습니다.

다음 테이블에는 독립 실행형 ActiveVOS를 포함한 ActiveVOS로 마이그레이션하는 데 필요한 업그레이드 전 태스크가 설명되어 있습니다.

업그레이드 태스크	설명
ActiveVOS 데이터 소스 제거	독립 실행형 ActiveVOS 서버가 Multidomain MDM과 동일한 JBoss 인스턴스, WebSphere 프로파일 또는 WebLogic 도메인에 있는 경우 응용 프로그램 서버에서 ActiveVOS에 대한 데이터 소스를 제거합니다.
WebSphere 환경의 URN 매핑 편집	<ol style="list-style-type: none"> 1. ActiveVOS 콘솔을 시작합니다. 브라우저에서 다음 URL을 올바른 호스트 이름과 포트 번호로 대체하여 입력합니다. <ul style="list-style-type: none"> - 보안 연결. <code>https://<호스트>:<포트>/activevos</code> - 비보안 연결. <code>http://<호스트>:<포트>/activevos</code> 2. ActiveVOS 콘솔의 홈 페이지에서 관리 > 서버 구성 > URN 매핑을 클릭합니다. 3. URN <code>java:comp/env/jdbc/ActiveVOS</code>가 URL <code>java:comp/env/jdbc/ActiveVOS</code>에 매핑되는지 확인합니다.

중요: 업그레이드 프로세스의 일부로 Hub 서버 설치 프로그램을 실행하는 경우 포함된 ActiveVOS를 설치해야 합니다.

빌드 속성 파일 편집(인플레이스 업그레이드)

MDM 환경에 ActiveVOS가 설치된 경우 업그레이드 태스크를 수행하여 `build.properties` 파일을 편집해야 합니다.

다음 테이블에는 `build.properties` 파일을 편집하는 데 필요한 업그레이드 태스크가 설명되어 있습니다.

업그레이드 태스크	세부 정보
IBM Db2에 대한 데이터베이스 연결 속성 구성	<p>업그레이드하기 전에 IBM Db2 환경에서 ActiveVOS 데이터베이스 연결 속성을 <MDM Hub 설치 디렉터리>/hub/server/bin에 있는 <code>build.properties</code> 파일에 추가합니다.</p> <p>다음 샘플은 ActiveVOS 데이터베이스 연결 속성을 샘플 항목과 함께 보여줍니다.</p> <pre> activevos.db.type=db2 activevos.db.server=localhost activevos.db.port=50000 activevos.db.user=AVOS activevos.db.dbname=INFA102 activevos.db.schemaname=AVOS activevos.db.jdbc.url=jdbc:db2://localhost:50000/INFA102 activevos.b4p.url=http://localhost:9080/active-bpel/services/AeB4PTaskClient-taskOperations </pre>
ActiveVOS 설치 디렉터리 구성	<p>업그레이드 프로세스에서 ActiveVOS의 설치된 버전이 시스템 요구 사항을 충족하지 않는 것으로 검색되면 필요한 버전의 ActiveVOS가 새 디렉터리에 설치됩니다.</p> <p>ActiveVOS를 이전 설치와 동일한 디렉터리에 설치하려면 <code>build.properties</code> 파일에서 다음 항목을 삭제하거나 주석 처리합니다.</p> <pre> activevosinstall.dir=<ActiveVOS installation directory> </pre>

제 3 장

데이터베이스 태스크

이 장에 포함된 항목:

- [데이터베이스 태스크 개요, 26](#)
- [Oracle 데이터베이스 설정, 26](#)
- [Microsoft SQL Server 설정, 33](#)
- [IBM Db2 데이터베이스 설정, 37](#)

데이터베이스 태스크 개요

Multidomain MDM에서 작동하도록 데이터베이스를 구성합니다. 데이터베이스에 대한 지침을 따르십시오.

클린 업그레이드를 수행하는 경우 Multidomain MDM의 새 설치와 마찬가지로 데이터베이스에 대한 모든 단계를 수행합니다.

데이터베이스 버전이 계속해서 지원되는 인플레이스 업그레이드를 수행하는 경우 이 버전의 Multidomain MDM에서 작동하도록 데이터베이스가 구성되었는지 확인합니다.

Oracle 데이터베이스 설정

MDM Hub 마스터 데이터베이스 및 연산 참조 저장소를 생성하기 전에 데이터베이스 환경을 설정합니다.

데이터베이스 환경을 설정하려면 다음 태스크를 수행합니다.

1. Oracle을 설치 및 구성합니다.
2. 데이터베이스 인스턴스를 설정합니다.
3. 데이터베이스 권한 및 연결을 설정합니다.
4. ActiveVOS 스키마를 생성합니다.

1단계. Oracle 설치 및 구성

Oracle 설명서의 지침에 따라 Oracle 데이터베이스를 설치 및 구성할 수 있습니다.

다음 테이블에는 Oracle 인스턴스를 원하는 각 시스템에서 수행해야 할 Oracle 설치 및 구성 태스크가 설명되어 있습니다.

태스크	설명
Oracle 설치	지원되는 버전의 Oracle 데이터베이스를 설치합니다. 참고: Oracle 멀티테넌트 기능을 사용하려는 경우 MDM Hub 설치에 대한 PDB(플러그인 가능한 데이터베이스)를 설정합니다.
클라이언트 및 유틸리티 설치	MDM Hub와 통신하고 MDM Hub 프로세스를 실행하려면 Oracle 클라이언트 및 유틸리티를 설치합니다. Hub 서버 또는 처리 서버를 실행할 각 시스템에 다음 소프트웨어를 설치합니다. - Oracle 클라이언트 - SQL*Loader
휴지통 비활성화	휴지통 USER_RECYCLEBIN 및 DBA_RECYCLEBIN을 비활성화합니다. 휴지통은 MDM Hub 프로세스를 간섭할 수 있습니다. 시스템 또는 세션 수준에서 휴지통을 비활성화합니다. 또한 휴지통에서 기존 개체를 제거합니다.
초기화 매개 변수 설정	init.ora 파일에서 Oracle 초기화 매개 변수를 구성합니다. Oracle 초기화 매개 변수에 대한 자세한 내용은 <i>MDM Multidomain Edition 성능 조정 가이드</i> 를 참조하십시오.

Oracle 설치 및 구성에 대한 자세한 내용은 Oracle 설명서를 참조하십시오.

2단계. 데이터베이스 및 테이블스페이스 생성

Oracle 데이터베이스를 설치 및 구성한 후 데이터베이스 및 테이블스페이스를 생성 및 구성합니다.

다음 테이블에는 데이터베이스 구성에 필요한 태스크가 설명되어 있습니다.

태스크	설명
데이터베이스 생성	각 데이터베이스 인스턴스에 대한 데이터베이스를 생성합니다.
테이블스페이스 생성	MDM Hub 데이터에 대한 테이블스페이스를 생성합니다. MDM Hub에 로드할 데이터 볼륨을 기반으로 기본 테이블스페이스 크기 및 데이터 파일 수를 조정합니다. 다음 테이블스페이스를 생성합니다. - CMX_DATA. MDM Hub의 메타데이터 및 사용자 데이터를 포함합니다. - CMX_INDX. MDM Hub가 생성 및 사용하는 인덱스를 포함합니다. - CMX_TEMP. MDM Hub에 대한 임시 테이블을 포함합니다. - BPM_DATA. 비즈니스 프로세스 관리에 대한 ActiveVOS 데이터를 포함합니다. 참고: 여러 MDM Hub 마스터 데이터베이스를 생성하려는 경우 각 MDM Hub 마스터 데이터베이스에 대해 고유한 테이블스페이스를 생성합니다.

내부에 테이블스페이스 생성

내부에 Oracle을 사용하는 경우 테이블스페이스를 로컬로 관리하거나 수동으로 구성하는 영구 테이블스페이스로 생성합니다.

참고: 큰 데이터 로드에 대한 데이터베이스 관리를 간소화하려면 큰 파일 테이블스페이스를 생성합니다. 그러나 필요한 경우 작은 파일 테이블스페이스를 생성할 수 있습니다.

1. Oracle에 SYSTEM과 같은 데이터베이스 관리자로 로그인합니다.
2. 테이블스페이스를 생성합니다.

다음 테이블에는 테이블스페이스를 생성하기 위한 샘플 SQL 문이 포함되어 있습니다.

테이블스페이스 이름	샘플 SQL 문
CMX_DATA	참고: 리포지토리 관리자 유효성 검사 오류를 방지하려면 기본 테이블스페이스 이름을 변경하지 마십시오. CREATE BIGFILE TABLESPACE CMX_DATA NOLOGGING DATAFILE '<Oracle install directory>/CMX_DATA1.dbf' SIZE 2048M REUSE EXTENT MANAGEMENT LOCAL;
CMX_INDX	CREATE BIGFILE TABLESPACE CMX_INDX NOLOGGING DATAFILE '<Oracle install directory>/CMX_INDX1.dbf' SIZE 2048M REUSE EXTENT MANAGEMENT LOCAL;
CMX_TEMP	CREATE BIGFILE TABLESPACE CMX_TEMP NOLOGGING DATAFILE '<Oracle install directory>/CMX_TEMP1.dbf' SIZE 2048M REUSE EXTENT MANAGEMENT LOCAL;
BPM_DATA	CREATE BIGFILE TABLESPACE BPM_DATA NOLOGGING DATAFILE '<Oracle install directory>/BPM_DATA1.dbf' SIZE 2048M REUSE EXTENT MANAGEMENT LOCAL;

Amazon 관계형 데이터베이스 서비스에서 테이블스페이스 생성

Oracle용 Amazon RDS(관계형 데이터베이스 서비스)를 사용하는 경우 Amazon RDS에서 테이블스페이스를 영구 테이블스페이스로 생성합니다.

참고: 큰 데이터 로드에 대한 데이터베이스 관리를 간소화하려면 큰 파일 테이블스페이스를 생성합니다. 그러나 필요한 경우 작은 파일 테이블스페이스를 생성할 수 있습니다.

1. Oracle용 Amazon RDS에 데이터베이스 관리자로 로그인합니다.
2. 테이블스페이스를 생성합니다.

다음 테이블에는 테이블스페이스를 생성하기 위한 샘플 SQL 문이 포함되어 있습니다.

기본 테이블스페이스 이름	샘플 SQL 문
CMX_DATA	<pre>CREATE BIGFILE TABLESPACE CMX_DATA DATAFILE SIZE 2048M AUTOEXTEND ON NEXT 2048M;</pre> <p>리포지토리 관리자 유효성 검사 오류를 방지하려면 기본 테이블스페이스 이름을 변경하지 마십시오.</p>
CMX_INDX	<pre>CREATE BIGFILE TABLESPACE CMX_INDX DATAFILE SIZE 2048M AUTOEXTEND ON NEXT 2048M;</pre>
CMX_TEMP	<pre>CREATE BIGFILE TABLESPACE CMX_TEMP DATAFILE SIZE 2048M AUTOEXTEND ON NEXT 2048M;</pre>

사용자 지정 테이블스페이스 이름에 대한 데이터베이스 환경 구성

CMX_INDX 또는 CMX_TEMP에 대해 기본 테이블스페이스 이름이 아닌 테이블스페이스 이름을 사용하는 경우 DEFERRED_SEGMENT_CREATION 초기화 매개 변수를 비활성화합니다. 리포지토리 관리자 유효성 검사 오류를 방지하려면 이 매개 변수를 비활성화합니다.

- ▶ DEFERRED_SEGMENT_CREATION을 비활성화하려면 다음과 같은 SQL 문을 실행하고 데이터베이스를 다시 시작합니다.

```
ALTER SYSTEM SET DEFERRED_SEGMENT_CREATION=FALSE SCOPE=BOTH;
```

3단계. 데이터베이스 권한 및 연결 설정

데이터베이스 권한 및 연결을 설정합니다.

다음 테이블에는 데이터베이스 권한 및 연결을 설정하기 위해 수행해야 하는 태스크가 설명되어 있습니다.

태스크	설명
데이터베이스 관리자에게 권한 부여	<p>데이터베이스 관리자를 사용하여 MDM Hub 마스터 데이터베이스 및 연산 참조 저장소를 생성하려는 경우 데이터베이스 관리자에게 권한을 부여합니다. 사용자에게 배포 트랜잭션 및 DBMS_LOCK을 위한 옵션을 부여할 수 있는 권한이 있어야 합니다.</p> <p>데이터베이스 관리자에게 권한을 부여하려면 옵션을 부여할 수 있는 권한이 있는 사용자로 데이터베이스에 연결하고 다음 SQL 문을 실행합니다.</p> <pre>GRANT SELECT ON sys.pending_trans\$ TO <DBA user> with grant option; GRANT SELECT ON sys.dbms_pending_transactions TO <DBA user> with grant option; GRANT SELECT ON sys.dbms_2pc_pending TO <DBA user> with grant option; GRANT EXECUTE ON sys.dbms_xa TO <DBA user> with grant option; GRANT EXECUTE ON sys.dbms_lock TO <DBA user> with grant option;</pre>
Oracle TNS 이름 추가	<p>Oracle 데이터베이스에 대한 연결의 경우 Hub 서버 및 처리 서버 시스템의 tnsnames.ora 파일에 TNS 이름 항목을 추가합니다.</p> <p>TNS 이름 항목을 추가하려면 다음 구문을 사용합니다.</p> <pre><TNS_NAME> = (DESCRIPTION = (ADDRESS_LIST = (ADDRESS = (PROTOCOL = TCP)(Host = <Oracle server host name>)(Port = <Oracle server port>))) (CONNECT_DATA = (SERVICE_NAME = <Oracle SID>)))</pre> <p>참고: Hub 서버 및 처리 서버 시스템의 TNS 이름은 동일해야 합니다.</p>
데이터베이스 연결 테스트	<p>Hub 서버 또는 처리 서버를 실행하려는 각 시스템에서 데이터베이스 연결을 테스트합니다.</p> <p>SQL*Plus에서 다음 SQL 문 구문을 사용합니다.</p> <pre>sqlplus <사용자 이름>/@<암호>@<TNS 이름></pre>

4단계. ActiveVOS 스키마 생성4단계. ActiveVOS 데이터베이스 생성

ActiveVOS를 설치하려면 ActiveVOS 스키마를 생성해야 합니다. 스키마를 생성하려면 create_bpm 스크립트를 실행합니다. ActiveVOS를 설치하려면 ActiveVOS 데이터베이스를 생성해야 합니다. 데이터베이스를 생성하려면 create_bpm 스크립트를 실행합니다.

여러 MDM Hub 마스터 데이터베이스를 생성하려는 경우 각 MDM Hub 마스터 데이터베이스에 대해 ActiveVOS 스키마를 생성합니다.

참고: Oracle 멀티테넌트 기능을 사용하려는 경우 PDB(플러그인 가능한 데이터베이스)에 ActiveVOS 스키마를 생성합니다.

여러 MDM Hub 마스터 데이터베이스를 생성하려는 경우 각 MDM Hub 마스터 데이터베이스에 대해 ActiveVOS 데이터베이스를 생성합니다.

1. 명령 프롬프트를 열고 다음 디렉터리로 변경합니다.

```
<MDM Hub 배포 디렉터리>/database/bin
```

2. 다음 명령을 실행합니다.

UNIX의 경우. `./sip_ant.sh create_bpm`

Windows의 경우. `sip_ant.bat create_bpm`

3. 나타나는 프롬프트에 응답합니다.

프롬프트에 대괄호로 묶인 기본 텍스트가 표시됩니다. **Enter** 키를 눌러 기본값을 사용하고 다음 프롬프트로 이동합니다.

참고: IBM Db2 데이터 소스는 대/소문자를 구분합니다. 대/소문자 구분과 관련된 문제를 방지하려면 대문자를 사용하여 스키마, 열 및 트리거에 대한 이름을 정의하는 것이 좋습니다.

속성	설명
데이터베이스 유형	사용할 데이터베이스 유형입니다. IBM Db2의 경우 DB2를 지정합니다. 데이터베이스 유형은 MDM Hub 마스터 데이터베이스 및 연산 참조 저장소에 대해 선택한 데이터베이스 유형과 동일해야 합니다.
ActiveVOS 데이터베이스 호스트 이름	데이터베이스를 호스팅하는 시스템의 이름입니다.
ActiveVOS 데이터베이스 TCP/IP 포트	데이터베이스 수신기에서 사용하는 포트 번호입니다.
ActiveVOS 데이터베이스 이름	데이터베이스 이름입니다.
ActiveVOS 데이터베이스 스키마/사용자 이름	ActiveVOS 서버 관리자의 사용자 이름입니다.
ActiveVOS 사용자 암호	관리자의 암호입니다.
DBA 사용자 이름	데이터베이스 관리자의 사용자 이름입니다.
DBA 암호	관리자의 암호입니다.
ActiveVOS 테이블스페이스 이름	MDM 워크플로우와 관련된 레코드가 포함된 테이블스페이스의 이름입니다.

속성	설명
데이터베이스 유형	데이터베이스의 유형입니다. Oracle 데이터베이스의 경우 Oracle을 지정합니다. 데이터베이스 유형은 MDM Hub 마스터 데이터베이스 및 연산 참조 저장소에 대해 선택한 데이터베이스 유형과 동일해야 합니다.
Oracle 연결 유형	연결 유형입니다. 다음 값 중 하나를 사용합니다. - SERVICE. 서비스 이름을 사용하여 Oracle에 연결합니다. - SID. Oracle 시스템 ID를 사용하여 Oracle에 연결합니다.
ActiveVOS 데이터베이스 호스트 이름	데이터베이스를 호스팅하는 시스템의 이름입니다.
ActiveVOS 데이터베이스 포트	데이터베이스 수신기에서 사용하는 포트 번호입니다.
데이터베이스 서비스 이름	Oracle 서비스 이름입니다. 선택한 Oracle 연결 유형이 SERVICE인 경우 이 속성이 필요합니다.

속성	설명
Oracle Net 연결 식별자(TNS 이름)	Oracle TNS 이름입니다.
데이터베이스 SID	Oracle 시스템 ID의 이름입니다. 선택한 Oracle 연결 유형이 SID인 경우 이 속성이 필요합니다.
DBA 사용자 이름	데이터베이스 관리자의 사용자 이름입니다.
DBA 암호	관리자의 암호입니다.
ActiveVOS 사용자 이름	ActiveVOS 서버 관리자의 사용자 이름입니다.
ActiveVOS 사용자 암호	관리자의 암호입니다.
ActiveVOS 사용자 테이블스페이스	MDM 워크플로우와 관련된 레코드가 포함된 테이블스페이스의 이름입니다.
ActiveVOS 사용자 임시 테이블스페이스	임시 테이블스페이스의 이름입니다.

속성	설명
데이터베이스 유형	사용할 데이터베이스 유형입니다. Microsoft SQL Server의 경우 MSSQL을 지정합니다. 데이터베이스 유형은 MDM Hub 마스터 데이터베이스 및 연산 참조 저장소에 대해 선택한 데이터베이스 유형과 동일해야 합니다.
ActiveVOS 사용자 이름	ActiveVOS 서버 관리자의 사용자 이름입니다.
ActiveVOS 사용자 암호	관리자의 암호입니다.
ActiveVOS 데이터 정렬 이름	ActiveVOS 데이터베이스 정렬의 이름입니다. 예: Latin1_General_CI_AS.
ActiveVOS 데이터베이스 호스트 이름	데이터베이스를 호스팅하는 시스템의 이름입니다.
ActiveVOS 데이터베이스 경로	데이터베이스 위치의 경로입니다.
DBA 사용자 이름	SA 사용자의 사용자 이름입니다.
DBA 암호	SA 사용자 계정의 암호입니다.

4. 스키마를 생성한 후 다음 디렉터리에서 sip_ant.log 파일을 검토합니다. 데이터베이스를 생성한 후 다음 디렉터리에서 sip_ant.log 파일을 검토합니다.

<MDM Hub 배포 디렉터리>/database/bin

sip_ant.log 파일은 sip_ant 스크립트를 실행하여 ActiveVOS 데이터베이스 스키마를 생성할 때 발생할 수도 있는 모든 오류를 기록합니다.

Microsoft SQL Server 설정

MDM Hub 마스터 데이터베이스 및 연산 참조 저장소를 생성하기 전에 데이터베이스 환경을 설정합니다.

데이터베이스 환경을 설정하려면 다음 태스크를 수행합니다.

1. Microsoft SQL Server를 설치 및 구성합니다.
2. 데이터 파일 저장소를 생성합니다.
3. ODBC 드라이버를 설치합니다.
4. ActiveVOS 스키마를 생성합니다.

1단계. Microsoft SQL Server 설치 및 구성

Microsoft SQL Server 설명서의 지침에 따라 Microsoft SQL Server를 설치 및 구성할 수 있습니다.

다음 테이블에는 Microsoft SQL Server 설치 및 구성 태스크가 설명되어 있습니다.

태스크	설명
Microsoft SQL Server 설치	지원되는 버전의 Microsoft SQL Server를 설치합니다. 참고: 데이터베이스 엔진 인증 보안 모드로 혼합 모드를 지정해야 합니다.
JDBC 드라이버 구성	Microsoft SQL Server 2017을 사용하는 환경에 Multidomain MDM을 설치하려는 경우 다음 태스크를 수행합니다. 1. Microsoft 웹 사이트에서 지원되는 최신 버전의 Microsoft JDBC 드라이버를 다운로드합니다. 2. 파일을 Microsoft SQL Server가 설치되어 있는 시스템의 Binn 디렉터리에 복사합니다.
배포 트랜잭션 구성	MDM Hub에 배포 트랜잭션 환경이 필요하기 때문에 배포 트랜잭션을 위한 Microsoft SQL Server를 구성합니다. 배포 트랜잭션을 위한 Microsoft SQL Server를 구성하려면 XA 트랜잭션을 위한 MS DTC를 활성화하고 JDBC 배포 트랜잭션 구성 요소를 구성합니다.
TCP/IP 네트워크 프로토콜 활성화	MDM Hub 환경에 필요한 TCP/IP 네트워크 프로토콜을 사용하도록 Microsoft SQL Server를 구성합니다.

배포 트랜잭션을 위한 Microsoft SQL Server 구성

Microsoft SQL Server를 시작하기 전에 Microsoft SQL Server를 배포 트랜잭션용으로 구성해야 합니다. MDM Hub를 사용하려면 배포 트랜잭션 환경이 필요합니다.

1. Microsoft SQL Server를 시작할 때 MS DTC(Microsoft Distributed Transaction Coordinator) 서비스가 시작되도록 하려면 서비스 관리자에서 MS DTC를 자동으로 표시합니다.
2. XA 트랜잭션에 대해 MS DTC를 활성화합니다.
3. SNA LU 6.2 트랜잭션을 위한 MS DTC를 활성화합니다.
4. JDBC 배포 트랜잭션 구성 요소를 구성합니다.
 - a. 지원되는 Microsoft SQL Server JDBC 드라이버 버전을 Microsoft 웹 사이트에서 시스템의 디렉터리로 다운로드 및 추출합니다.
 - b. sqljdbc_xa.dll 파일을 Microsoft SQL Server가 설치되어 있는 시스템의 Binn 디렉터리에 복사합니다.
 - c. xa_install.sql 스크립트를 Microsoft SQL Server 인스턴스에서 관리자로서 실행합니다.

xa_install.sql 스크립트는 JDBC 드라이버의 xa 디렉터리에 있습니다.

- d. 스크립트가 Microsoft SQL Server 마스터 데이터베이스에 SqlJDBCXAUser 역할을 생성하는지 확인합니다.

배포 트랜잭션용 Microsoft SQL Server를 설치 및 구성하는 방법에 대한 자세한 내용은 Microsoft SQL Server 설명서를 참조하십시오.

TCP/IP 네트워크 프로토콜 활성화

SQL Server 구성 관리자를 사용하여 MDM Hub 환경에 필요한 TCP/IP 네트워크 프로토콜을 활성화합니다.

1. SQL Server 구성 관리자를 시작합니다.
2. 콘솔 창에서 **SQL Server 네트워크 구성**을 확장한 다음 **MSSQLSERVER용 프로토콜**을 클릭합니다.
3. 세부 정보 창에서 **TCP/IP**를 마우스 오른쪽 단추로 클릭하고 **사용**을 클릭합니다.
4. SQL Server 서비스를 다시 시작합니다.

TCP/IP 네트워크 프로토콜이 활성화됩니다.

2단계. Microsoft SQL Server 데이터 파일 저장소 생성

Microsoft SQL Server 데이터 파일 저장소가 없는 경우 생성해야 합니다. MDM Hub 마스터 데이터베이스 및 연산 참조 저장소를 생성하는 경우 Microsoft SQL Server 데이터 파일 저장소에 대한 경로를 제공해야 합니다.

3단계. ODBC 드라이버 설치

UNIX 또는 Linux에 응용 프로그램 서버를 설치하는 경우 Microsoft SQL Server용 ODBC 드라이버를 설치해야 데이터 소스를 생성할 수 있습니다.

- ▶ 운영 체제에 대한 ODBC 드라이버를 다운로드 및 설치합니다.
 - Linux의 경우, Microsoft 웹 사이트에서 Microsoft SQL Server ODBC 드라이버를 다운로드합니다.
 - UNIX의 경우, unixODBC 프로젝트 홈 페이지에서 UNIX ODBC 드라이버를 다운로드합니다.

4단계. ActiveVOS 스키마 생성4단계. ActiveVOS 데이터베이스 생성

ActiveVOS를 설치하려면 ActiveVOS 스키마를 생성해야 합니다. 스키마를 생성하려면 create_bpm 스크립트를 실행합니다. ActiveVOS를 설치하려면 ActiveVOS 데이터베이스를 생성해야 합니다. 데이터베이스를 생성하려면 create_bpm 스크립트를 실행합니다.

여러 MDM Hub 마스터 데이터베이스를 생성하려는 경우 각 MDM Hub 마스터 데이터베이스에 대해 ActiveVOS 스키마를 생성합니다.

참고: Oracle 멀티테넌트 기능을 사용하려는 경우 PDB(플러그인 가능한 데이터베이스)에 ActiveVOS 스키마를 생성합니다.

여러 MDM Hub 마스터 데이터베이스를 생성하려는 경우 각 MDM Hub 마스터 데이터베이스에 대해 ActiveVOS 데이터베이스를 생성합니다.

1. 명령 프롬프트를 열고 다음 디렉터리로 변경합니다.

<MDM Hub 배포 디렉터리>/database/bin

2. 다음 명령을 실행합니다.

UNIX의 경우. `./sip_ant.sh create_bpm`

Windows의 경우. `sip_ant.bat create_bpm`

3. 나타나는 프롬프트에 응답합니다.

프롬프트에 대괄호로 묶인 기본 텍스트가 표시됩니다. **Enter** 키를 눌러 기본값을 사용하고 다음 프롬프트로 이동합니다.

참고: IBM Db2 데이터 소스는 대/소문자를 구분합니다. 대/소문자 구분과 관련된 문제를 방지하려면 대문자를 사용하여 스키마, 열 및 트리거에 대한 이름을 정의하는 것이 좋습니다.

속성	설명
데이터베이스 유형	사용할 데이터베이스 유형입니다. IBM Db2의 경우 DB2를 지정합니다. 데이터베이스 유형은 MDM Hub 마스터 데이터베이스 및 연산 참조 저장소에 대해 선택한 데이터베이스 유형과 동일해야 합니다.
ActiveVOS 데이터베이스 호스트 이름	데이터베이스를 호스팅하는 시스템의 이름입니다.
ActiveVOS 데이터베이스 TCP/IP 포트	데이터베이스 수신기에서 사용하는 포트 번호입니다.
ActiveVOS 데이터베이스 이름	데이터베이스 이름입니다.
ActiveVOS 데이터베이스 스키마/사용자 이름	ActiveVOS 서버 관리자의 사용자 이름입니다.
ActiveVOS 사용자 암호	관리자의 암호입니다.
DBA 사용자 이름	데이터베이스 관리자의 사용자 이름입니다.
DBA 암호	관리자의 암호입니다.
ActiveVOS 테이블스페이스 이름	MDM 워크플로우와 관련된 레코드가 포함된 테이블스페이스의 이름입니다.

속성	설명
데이터베이스 유형	데이터베이스의 유형입니다. Oracle 데이터베이스의 경우 Oracle을 지정합니다. 데이터베이스 유형은 MDM Hub 마스터 데이터베이스 및 연산 참조 저장소에 대해 선택한 데이터베이스 유형과 동일해야 합니다.
Oracle 연결 유형	연결 유형입니다. 다음 값 중 하나를 사용합니다. - SERVICE. 서비스 이름을 사용하여 Oracle에 연결합니다. - SID. Oracle 시스템 ID를 사용하여 Oracle에 연결합니다.
ActiveVOS 데이터베이스 호스트 이름	데이터베이스를 호스팅하는 시스템의 이름입니다.
ActiveVOS 데이터베이스 포트	데이터베이스 수신기에서 사용하는 포트 번호입니다.
데이터베이스 서비스 이름	Oracle 서비스 이름입니다. 선택한 Oracle 연결 유형이 SERVICE인 경우 이 속성이 필요합니다.

속성	설명
Oracle Net 연결 식별자(TNS 이름)	Oracle TNS 이름입니다.
데이터베이스 SID	Oracle 시스템 ID의 이름입니다. 선택한 Oracle 연결 유형이 SID인 경우 이 속성이 필요합니다.
DBA 사용자 이름	데이터베이스 관리자의 사용자 이름입니다.
DBA 암호	관리자의 암호입니다.
ActiveVOS 사용자 이름	ActiveVOS 서버 관리자의 사용자 이름입니다.
ActiveVOS 사용자 암호	관리자의 암호입니다.
ActiveVOS 사용자 테이블스페이스	MDM 워크플로우와 관련된 레코드가 포함된 테이블스페이스의 이름입니다.
ActiveVOS 사용자 임시 테이블스페이스	임시 테이블스페이스의 이름입니다.

속성	설명
데이터베이스 유형	사용할 데이터베이스 유형입니다. Microsoft SQL Server의 경우 MSSQL을 지정합니다. 데이터베이스 유형은 MDM Hub 마스터 데이터베이스 및 연산 참조 저장소에 대해 선택한 데이터베이스 유형과 동일해야 합니다.
ActiveVOS 사용자 이름	ActiveVOS 서버 관리자의 사용자 이름입니다.
ActiveVOS 사용자 암호	관리자의 암호입니다.
ActiveVOS 데이터 정렬 이름	ActiveVOS 데이터베이스 정렬의 이름입니다. 예: Latin1_General_CI_AS.
ActiveVOS 데이터베이스 호스트 이름	데이터베이스를 호스팅하는 시스템의 이름입니다.
ActiveVOS 데이터베이스 경로	데이터베이스 위치의 경로입니다.
DBA 사용자 이름	SA 사용자의 사용자 이름입니다.
DBA 암호	SA 사용자 계정의 암호입니다.

4. 스키마를 생성한 후 다음 디렉터리에서 sip_ant.log 파일을 검토합니다. 데이터베이스를 생성한 후 다음 디렉터리에서 sip_ant.log 파일을 검토합니다.

<MDM Hub 배포 디렉터리>/database/bin

sip_ant.log 파일은 sip_ant 스크립트를 실행하여 ActiveVOS 데이터베이스 스키마를 생성할 때 발생할 수도 있는 모든 오류를 기록합니다.

IBM Db2 데이터베이스 설정

MDM Hub 마스터 데이터베이스 및 연산 참조 저장소를 생성하기 전에 데이터베이스 환경을 설정합니다.

데이터베이스 환경을 설정하려면 다음 태스크를 수행합니다.

1. IBM Db2를 설치 및 구성합니다.
2. 데이터베이스 및 테이블스페이스를 생성합니다.
3. 데이터베이스 서버에서 패키지를 바인딩합니다.
4. ActiveVOS 스키마를 생성합니다.

1단계. IBM Db2 설치 및 구성

IBM Db2 설명서의 지침에 따라 IBM Db2를 설치 및 구성할 수 있습니다.

다음 테이블에는 IBM Db2 인스턴스를 원하는 각 시스템에서 수행해야 할 IBM Db2 설치 및 구성 태스크가 설명되어 있습니다.

태스크	설명
IBM Db2 설치	지원되는 버전의 IBM Db2를 설치합니다.
클라이언트 및 유틸리티 설치	MDM Hub와 통신하고 MDM Hub 프로세스를 실행하려면 IBM Db2 클라이언트 및 유틸리티를 설치합니다. Hub 서버 또는 처리 서버를 실행할 각 시스템에 다음 소프트웨어를 설치합니다. <ul style="list-style-type: none">- Db2 클라이언트- Db2 클라이언트용 Db2 Java 유틸리티 각 Db2 클라이언트에서 IBM Db2 데이터베이스 카탈로그를 생성해야 합니다.
IBM Db2 드라이버 구성	IBM Db2 드라이버를 구성하려면 소스에서 대상 디렉터리로 db2jcc.jar 및 db2jcc_license_cu.jar 드라이버 파일을 복사합니다. 소스: <IBM Db2 설치 디렉터리>/java 대상: <MDM Hub 배포 디렉터리>/database/lib
MDM Hub 스키마 사용자 생성	다음 MDM Hub 스키마에 액세스할 사용자를 생성합니다. <ul style="list-style-type: none">- MDM Hub 마스터 데이터베이스- 연산 참조 저장소

2단계. 데이터베이스 및 테이블스페이스 생성

IBM Db2를 설치 및 구성한 후 데이터베이스와 테이블스페이스를 생성 및 구성합니다. 각 데이터베이스 인스턴스에 대한 데이터베이스를 생성해야 합니다.

참고: 여러 MDM Hub 마스터 데이터베이스를 생성하려는 경우 각 MDM Hub 마스터 데이터베이스에 대해 고유한 테이블스페이스를 생성합니다.

다음 테이블에는 MDM Hub 스키마에 필요한 테이블스페이스가 설명되어 있습니다.

테이블스페이스 이름	설명
CMX_DATA	연산 참조 저장소 스키마용 기본 테이블스페이스입니다. MDM Hub의 메타데이터 및 사용자 데이터를 포함합니다.
CMX_INDX	MDM Hub에서 생성 및 사용하는 인덱스가 포함되는 테이블스페이스입니다.
CMX_TEMP	MDM Hub에서 생성 및 사용하는 임시 테이블이 포함되는 테이블스페이스입니다.
CMX_REPOS	연산 참조 저장소 개체가 포함되는 테이블스페이스입니다.
CMX_USER_TEMP	연산 임시 테이블이 포함되는 임시 테이블스페이스입니다.
CMX_SYS_TEMP	SQL 연산용 임시 테이블스페이스입니다.

다음 프로시저 중 하나를 사용하여 데이터베이스 및 테이블스페이스를 생성합니다.

- 수동으로 데이터베이스 및 테이블스페이스 생성
- 스크립트를 사용하여 데이터베이스 및 테이블스페이스 생성

수동으로 데이터베이스 및 테이블스페이스 생성

데이터베이스 및 테이블스페이스를 수동으로 생성할 수 있습니다. 호환성 벡터를 설정하고 UTF-8 TERRITORY US 로캘과 함께 데이터베이스를 생성해야 합니다.

Db2 환경 변수 및 Db2 레지스트리 변수 설정

데이터베이스를 수동으로 생성하는 경우 MDM Hub에 필요한 Db2 환경 변수 및 Db2 레지스트리 변수를 설정합니다.

다음 명령을 사용하여 Db2 환경 변수 및 Db2 레지스트리 변수를 설정합니다.

```
db2set DB2CODEPAGE=1208
db2set DB2_COMPATIBILITY_VECTOR=
db2set DB2_DEFERRED_PREPARE_SEMANTICS=YES
db2set DB2_RESTORE_GRANT_ADMIN_AUTHORITIES=ON
db2set DB2_HASH_JOIN=YES
db2set DB2_ANTIJOIN=YES
db2set DB2_INLIST_TO_NLJN=NO
db2set DB2_SELECTIVITY=ALL
db2set DB2_SKIPINSERTED=YES
db2set DB2_SKIPDELETED=YES
db2set DB2_EXTENDED_OPTIMIZATION=ON,ENHANCED_MULTIPLE_DISTINCT,IXOR,SNHD
db2set DB2NTNOCACHE=ON
db2set DB2_REDUCED_OPTIMIZATION=REDUCE_LOCKING
```

데이터베이스 인스턴스에 대해 데이터베이스 관리자 구성 설정

데이터베이스 인스턴스에 대해 데이터베이스 관리자 구성을 최적화해야 합니다.

다음 명령을 사용하여 데이터베이스 관리자 구성을 최적화합니다.

```
db2 update dbm cfg using MON_HEAP_SZ AUTOMATIC
db2 update dbm cfg using JAVA_HEAP_SZ 2048
db2 update dbm cfg using AGENT_STACK_SZ 256
db2 update dbm cfg using SHEAPTHRES 0
db2 update dbm cfg using INTRA_PARALLEL YES
```

참고: 명령에 지정된 값은 MDM Hub에 대한 최소 요구 사항입니다.

데이터베이스 구성 매개 변수 설정

데이터베이스에 대한 구성 매개 변수를 설정합니다.

다음 명령을 사용하여 데이터베이스 구성 매개 변수를 설정합니다.

```
db2 update db cfg using LOCKLIST AUTOMATIC
db2 update db cfg using MAXLOCKS AUTOMATIC
db2 update db cfg using PCKCACHESZ 128000
db2 update db cfg using DBHEAP AUTOMATIC
db2 update db cfg using CATALOGCACHE_SZ 25000
db2 update db cfg using LOGBUFSZ 4096
db2 update db cfg using UTIL_HEAP_SZ 50000
db2 update db cfg using BUFFPAGE 250
db2 update db cfg using STMTHEAP AUTOMATIC
db2 update db cfg using APPLHEAPSZ AUTOMATIC
db2 update db cfg using APPL_MEMORY AUTOMATIC
db2 update db cfg using STAT_HEAP_SZ AUTOMATIC
db2 update db cfg using LOGFILSIZ 128000
db2 update db cfg using LOGPRIMARY 10
db2 update db cfg using LOGSECOND 200
db2 update db cfg using auto_reval deferred_force
db2 update db cfg using decflt_rounding round_half_up
db2 update db cfg using SHEAPTHRES_SHR AUTOMATIC
db2 update db cfg using DFT_DEGREE 1
```

참고: 명령에 지정된 값은 MDM Hub에 대한 최소 요구 사항입니다.

SYSIBMADM 모듈에 대한 권한 부여

UTL_DIR, UTL_FILE 및 DBMS_SQL SYSIBMADM 모듈에 대한 권한을 부여해야 합니다.

다음 명령을 사용하여 모듈에 대한 권한을 부여합니다.

```
GRANT EXECUTE ON MODULE SYSIBMADM.UTL_DIR TO PUBLIC WITH GRANT OPTION
GRANT EXECUTE ON MODULE SYSIBMADM.UTL_FILE TO PUBLIC WITH GRANT OPTION
GRANT EXECUTE ON MODULE SYSIBMADM.DBMS_SQL TO PUBLIC WITH GRANT OPTION
```

데이터베이스 관리자용 버퍼 풀 정의

REPOS_POOL 및 CMX_POOL 버퍼 풀을 정의합니다.

다음 명령을 사용하여 버퍼 풀을 정의합니다.

```
CREATE BUFFERPOOL REPOS_POOL IMMEDIATE SIZE 1500 PAGESIZE 32 K
CREATE BUFFERPOOL CMX_POOL IMMEDIATE SIZE 3000 PAGESIZE 32 K
```

테이블스페이스 생성

MDM Hub 스키마에 필요한 테이블스페이스를 생성해야 합니다.

다음 순서대로 테이블스페이스를 생성합니다.

1. CMX_DATA
2. CMX_INDX
3. CMX_REPOS
4. CMX_TEMP
5. CMX_USER_TEMP
6. CMX_SYS_TEMP

다음 문을 사용하여 MDM Hub 스키마용 테이블스페이스를 생성합니다.

```
CREATE TABLESPACE CMX_DATA PAGESIZE 32 K
  MANAGED BY DATABASE USING ( FILE '<Db2 storage path>\CMX_DATA\cmx_data01.dat' 500 M )
  EXTENTSIZE 16
  AUTORESIZE YES
  OVERHEAD 10.5
  PREFETCHSIZE 16
  BUFFERPOOL CMX_POOL

CREATE TABLESPACE CMX_INDX PAGESIZE 32 K
  MANAGED BY DATABASE USING ( FILE '<Db2 storage path>\CMX_INDX\cmx_indx01.dat' 500 M )
  EXTENTSIZE 16
  AUTORESIZE YES
  OVERHEAD 10.5
  PREFETCHSIZE 16
  BUFFERPOOL CMX_POOL

CREATE TABLESPACE CMX_REPOS PAGESIZE 32 K
  MANAGED BY DATABASE USING ( FILE '<Db2 storage path>\CMX_REPOS\cmx_repos01.dat' 500 M )
  EXTENTSIZE 16
  AUTORESIZE YES
  OVERHEAD 10.5
  PREFETCHSIZE 16
  BUFFERPOOL REPOS_POOL

CREATE TABLESPACE CMX_TEMP PAGESIZE 32 K
  MANAGED BY DATABASE USING ( FILE '<Db2 storage path>\CMX_TEMP\cmx_temp01.dat' 500 M )
  EXTENTSIZE 16
  AUTORESIZE YES
  OVERHEAD 10.5
  PREFETCHSIZE 16
  BUFFERPOOL CMX_POOL

CREATE USER TEMPORARY TABLESPACE CMX_USER_TEMP PAGESIZE 32 K
  MANAGED BY DATABASE USING ( FILE '<Db2 storage path>\USER_TEMP\cmx_user_temp01.dat' 500 M )
  EXTENTSIZE 16
  AUTORESIZE YES
  OVERHEAD 10.5
  PREFETCHSIZE 16
  BUFFERPOOL CMX_POOL

CREATE SYSTEM TEMPORARY TABLESPACE CMX_SYS_TEMP PAGESIZE 32 K
  MANAGED BY DATABASE USING ( FILE '<Db2 storage path>\SYSTEM_TEMP\cmx_sys_temp01.dat' 500 M )
  EXTENTSIZE 16
  AUTORESIZE YES
  OVERHEAD 10.5
  PREFETCHSIZE 16
  BUFFERPOOL CMX_POOL
```

필요한 경우 삭제된 테이블 복구 기능을 활성화하여 테이블스페이스를 생성하려면 CREATE TABLESPACE 문에 다음 절을 추가합니다.

```
DROPPED TABLE RECOVERY ON
```

스크립트를 사용하여 데이터베이스 및 테이블스페이스 생성

MDM Hub 배포에는 데이터베이스와 관련 테이블스페이스를 생성하는 스크립트가 포함되어 있습니다. 스크립트를 실행하려면 Db2 데이터 디렉터리에 대해 읽기 및 실행 권한이 있는 관리 권한이 필요합니다.

UNIX에서 데이터베이스를 생성하기 전에 database.properties 파일의 db2.storage.path 속성을 올바른 데이터베이스 저장소 경로로 업데이트합니다. database.properties 파일은 다음 디렉터리에 있습니다.

```
<MDM Hub 배포 디렉터리>/database/bin/db2
```

1. 명령 프롬프트를 열고 다음 디렉터리로 변경합니다.

```
<MDM Hub 배포 디렉터리>/database/bin
```


- 데이터베이스를 생성하려면 다음 명령을 실행합니다.
UNIX의 경우. `./sip_ant.sh create_db`
Windows의 경우. `sip_ant.bat create_db`
- 다음 테이블에 설명되어 있는 프롬프트에 응답합니다.

프롬프트	설명
데이터베이스 유형(ORACLE, MSSQL, DB2) 입력	데이터베이스 유형입니다. DB2를 지정합니다.
데이터베이스 인스턴스 이름[db2]을 입력하십시오.	데이터베이스 인스턴스의 이름입니다. 기본값은 db2입니다.
데이터베이스 이름[SIP97]을 입력하십시오.	데이터베이스 이름입니다. 기본값은 SIP97입니다.
데이터베이스 저장소 경로 입력 [C:\DB2DATA]	데이터베이스를 저장할 디렉터리의 경로입니다. 기본값은 C:\DB2DATA입니다. 참고: UNIX에서는 기본값을 그대로 사용합니다. database.properties 파일에서 지정하는 데이터베이스 저장소 경로가 사용됩니다.
DBA 사용자 이름[DB2ADMIN]을 입력하십시오.	관리자의 사용자 이름입니다. 기본값은 DB2ADMIN입니다.
DBA 암호 입력	관리자의 암호입니다.

스크립트가 데이터베이스 및 다음 테이블스페이스를 생성합니다.

- CMX_DATA
- CMX_INDX
- CMX_TEMP
- CMX_REPOS
- CMX_USER_TEMP
- CMX_SYS_TEMP

데이터베이스가 성공적으로 생성되었는지 확인하려면 <MDM Hub 배포 디렉터리>/database/bin 디렉터리에서 sip_ant.log 파일을 검토합니다.

3단계. 데이터베이스 서버에서 패키지 바인딩

IBM Db2 클라이언트가 DB2 명령을 실행할 데이터베이스 서버에 연결할 수 있으면 데이터베이스 서버에서 패키지를 바인딩합니다.

- IBM Db2 명령 창을 열고 다음 디렉터리로 변경합니다.
<IBM Db2 설치 디렉터리>/SQLLIB/bnd
- 다음과 같은 명령을 실행하여 데이터베이스에 연결합니다.
`db2 connect to <데이터베이스 이름> user <데이터베이스 사용자> using <데이터베이스 사용자 암호>`
참고: 데이터베이스 사용자에게 바인딩 권한이 있어야 합니다.
- 다음과 같은 바인딩 명령을 실행합니다.
`db2 bind @db2cli.lst blocking all grant public sqlerror continue CLIPKG 10`

필수 패키지가 데이터베이스 서버에 바인딩되었습니다.

4단계. ActiveVOS 스키마 생성4단계. ActiveVOS 데이터베이스 생성

ActiveVOS를 설치하려면 ActiveVOS 스키마를 생성해야 합니다. 스키마를 생성하려면 `create_bpm` 스크립트를 실행합니다. ActiveVOS를 설치하려면 ActiveVOS 데이터베이스를 생성해야 합니다. 데이터베이스를 생성하려면 `create_bpm` 스크립트를 실행합니다.

여러 MDM Hub 마스터 데이터베이스를 생성하려는 경우 각 MDM Hub 마스터 데이터베이스에 대해 ActiveVOS 스키마를 생성합니다.

참고: Oracle 멀티테넌트 기능을 사용하려는 경우 PDB(플러그인 가능한 데이터베이스)에 ActiveVOS 스키마를 생성합니다.

여러 MDM Hub 마스터 데이터베이스를 생성하려는 경우 각 MDM Hub 마스터 데이터베이스에 대해 ActiveVOS 데이터베이스를 생성합니다.

1. 명령 프롬프트를 열고 다음 디렉터리로 변경합니다.

<MDM Hub 배포 디렉터리>/database/bin

2. 다음 명령을 실행합니다.

UNIX의 경우. `./sip_ant.sh create_bpm`

Windows의 경우. `sip_ant.bat create_bpm`

3. 나타나는 프롬프트에 응답합니다.

프롬프트에 대괄호로 묶인 기본 텍스트가 표시됩니다. **Enter** 키를 눌러 기본값을 사용하고 다음 프롬프트로 이동합니다.

참고: IBM Db2 데이터 소스는 대/소문자를 구분합니다. 대/소문자 구분과 관련된 문제를 방지하려면 대문자를 사용하여 스키마, 열 및 트리거에 대한 이름을 정의하는 것이 좋습니다.

속성	설명
데이터베이스 유형	사용할 데이터베이스 유형입니다. IBM Db2의 경우 DB2를 지정합니다. 데이터베이스 유형은 MDM Hub 마스터 데이터베이스 및 연산 참조 저장소에 대해 선택한 데이터베이스 유형과 동일해야 합니다.
ActiveVOS 데이터베이스 호스트 이름	데이터베이스를 호스팅하는 시스템의 이름입니다.
ActiveVOS 데이터베이스 TCP/IP 포트	데이터베이스 수신기에서 사용하는 포트 번호입니다.
ActiveVOS 데이터베이스 이름	데이터베이스 이름입니다.
ActiveVOS 데이터베이스 스키마/사용자 이름	ActiveVOS 서버 관리자의 사용자 이름입니다.
ActiveVOS 사용자 암호	관리자의 암호입니다.
DBA 사용자 이름	데이터베이스 관리자의 사용자 이름입니다.

속성	설명
DBA 암호	관리자의 암호입니다.
ActiveVOS 테이블스페이스 이름	MDM 워크플로우와 관련된 레코드가 포함된 테이블스페이스의 이름입니다.

속성	설명
데이터베이스 유형	데이터베이스의 유형입니다. Oracle 데이터베이스의 경우 Oracle을 지정합니다. 데이터베이스 유형은 MDM Hub 마스터 데이터베이스 및 연산 참조 저장소에 대해 선택한 데이터베이스 유형과 동일해야 합니다.
Oracle 연결 유형	연결 유형입니다. 다음 값 중 하나를 사용합니다. - SERVICE. 서비스 이름을 사용하여 Oracle에 연결합니다. - SID. Oracle 시스템 ID를 사용하여 Oracle에 연결합니다.
ActiveVOS 데이터베이스 호스트 이름	데이터베이스를 호스팅하는 시스템의 이름입니다.
ActiveVOS 데이터베이스 포트	데이터베이스 수신기에서 사용하는 포트 번호입니다.
데이터베이스 서비스 이름	Oracle 서비스 이름입니다. 선택한 Oracle 연결 유형이 SERVICE인 경우 이 속성이 필요합니다.
Oracle Net 연결 식별자(TNS 이름)	Oracle TNS 이름입니다.
데이터베이스 SID	Oracle 시스템 ID의 이름입니다. 선택한 Oracle 연결 유형이 SID인 경우 이 속성이 필요합니다.
DBA 사용자 이름	데이터베이스 관리자의 사용자 이름입니다.
DBA 암호	관리자의 암호입니다.
ActiveVOS 사용자 이름	ActiveVOS 서버 관리자의 사용자 이름입니다.
ActiveVOS 사용자 암호	관리자의 암호입니다.
ActiveVOS 사용자 테이블스페이스	MDM 워크플로우와 관련된 레코드가 포함된 테이블스페이스의 이름입니다.
ActiveVOS 사용자 임시 테이블스페이스	임시 테이블스페이스의 이름입니다.

속성	설명
데이터베이스 유형	사용할 데이터베이스 유형입니다. Microsoft SQL Server의 경우 MSSQL을 지정합니다. 데이터베이스 유형은 MDM Hub 마스터 데이터베이스 및 연산 참조 저장소에 대해 선택한 데이터베이스 유형과 동일해야 합니다.
ActiveVOS 사용자 이름	ActiveVOS 서버 관리자의 사용자 이름입니다.

속성	설명
ActiveVOS 사용자 암호	관리자의 암호입니다.
ActiveVOS 데이터 정렬 이름	ActiveVOS 데이터베이스 정렬의 이름입니다. 예: Latin1_General_CI_AS.
ActiveVOS 데이터베이스 호스트 이름	데이터베이스를 호스팅하는 시스템의 이름입니다.
ActiveVOS 데이터베이스 경로	데이터베이스 위치의 경로입니다.
DBA 사용자 이름	SA 사용자의 사용자 이름입니다.
DBA 암호	SA 사용자 계정의 암호입니다.

4. 스키마를 생성한 후 다음 디렉터리에서 sip_ant.log 파일을 검토합니다. 데이터베이스를 생성한 후 다음 디렉터리에서 sip_ant.log 파일을 검토합니다.

<MDM Hub 배포 디렉터리>/database/bin

sip_ant.log 파일은 sip_ant 스크립트를 실행하여 ActiveVOS 데이터베이스 스키마를 생성할 때 발생할 수도 있는 모든 오류를 기록합니다.

제 4 장

응용 프로그램 서버 태스크

이 장에 포함된 항목:

- [응용 프로그램 서버 태스크 개요, 45](#)
- [JBoss 설정, 45](#)
- [Oracle WebLogic 설정, 52](#)
- [추가 Oracle WebLogic 구성, 57](#)
- [IBM WebSphere 설정, 61](#)
- [추가 IBM WebSphere 구성, 67](#)

응용 프로그램 서버 태스크 개요

Multidomain MDM에서 작동하도록 응용 프로그램 서버를 구성합니다. 응용 프로그램 서버에 대한 지침을 따르십시오.

클린 업그레이드를 수행하는 경우 Multidomain MDM을 새로 설치하는 것과 마찬가지로 응용 프로그램 서버에 대한 모든 단계를 수행합니다.

응용 프로그램 서버가 계속해서 지원되는 인플레이스 업그레이드를 수행하는 경우 이 버전의 Multidomain MDM에서 작동하도록 응용 프로그램 서버가 구성되었는지 확인합니다.

JBoss 설정

JBoss 클러스터 환경 또는 독립 실행형 JBoss 인스턴스에 MDM Hub를 설치할 수 있습니다. JBoss 설명서의 지침에 따라 JBoss를 설치 및 구성합니다. MDM Hub를 JBoss 클러스터 환경 또는 독립 실행형 JBoss 인스턴스에 설치할 때는 JBoss 독립 실행형 구성을 설치하고 구성의 전체 프로필을 사용합니다.

JBoss 클러스터는 하나 이상의 시스템에 있는 하나 이상의 클러스터 노드로 구성됩니다. 클러스터 노드를 원하는 모든 시스템에 JBoss를 설치 및 구성합니다. 클러스터 환경에서 JBoss 설치의 디렉터리 구조가 모든 클러스터 노드에서 동일한지 확인합니다.

참고: 데이터베이스 서버와 동일한 시간대로 응용 프로그램 서버를 설치합니다.

Java Virtual Machine 구성

JVM(Java Virtual Machine)을 구성하려면 JAVA_OPTS 환경 변수를 사용하여 Java 옵션을 설정합니다.

다음과 같은 파일에서 Java 옵션을 설정할 수 있습니다.

UNIX의 경우. <JBoss 설치 디렉터리>/bin/standalone.conf

Windows의 경우. <JBoss 설치 디렉터리>\bin\standalone.conf.bat

다음 테이블에는 Java 옵션 설정이 설명되어 있습니다.

Java 옵션	설명
-server	시작은 느리지만 후속 작업은 빨라집니다.
-De360.connection.channel -De360.mdm.host -De360.mdm.port	응용 프로그램 서버 통신 프로토콜, 호스트 및 포트입니다. 4447 이외의 JBoss 포트에 MDM Hub 응용 프로그램을 배포하려면 다음 Java 옵션을 설정합니다. - -De360.connection.channel. 사용할 통신 프로토콜로 설정합니다. 유효한 값은 HTTP 및 HTTPS입니다. 기본값은 HTTP입니다. - -De360.mdm.host. JBoss 호스트의 IP 주소로 설정합니다. 환경에서 HTTPS 통신 프로토콜을 사용하고 보안 인증서가 FQDN(정규화된 도메인 이름)으로 발행되는 경우 FQDN으로 설정합니다. - -De360.mdm.port. 4447 대신 구성된 JBoss 원격 포트 설정합니다. 이러한 매개 변수를 구성하지 않으면 Entity 360 프레임워크에 기반한 Data Director 화면이 예상대로 작동하지 않을 수 있습니다.
-Didd.mdm.host -Didd.mdm.port -Didd.protocol	제목 영역이 있는 Data Director에 필요합니다. 제목 영역이 있는 Data Director를 배포하려면 다음 Java 옵션을 설정합니다. - -Didd.mdm.host. 응용 프로그램 서버 호스트의 호스트 이름 또는 IP 주소로 설정합니다. - -Didd.mdm.port. 기본값이 아닌 HTTP 또는 HTTPS 수신기 포트를 사용하여 배포하는 데 필요합니다. 이 속성은 서버 초기화 중 제목 영역 응용 프로그램을 배포하는 데 사용됩니다. 응용 프로그램용 JVM에 사용되는 HTTP 또는 HTTPS 수신기 포트를 지정합니다. 기본값은 8080입니다. - -Didd.protocol. 서버 초기화 중 제목 영역 응용 프로그램을 배포하는 데 사용되는 필수 속성입니다. 사용할 통신 프로토콜을 HTTP 또는 HTTPS로 지정합니다. 기본값은 HTTP입니다.
-Dio.undertow.legacy.cookie.ALLOW_HTTP_SEPARATORS_IN_V0	JBoss 버전 7.3을 사용하는 경우 필수 속성입니다. 이 속성을 시스템 속성으로 설정하고 잘린 쿠키 값으로 인해 필요한 자격 증명에 포함되지 않은 비즈니스 항목 서비스 요청의 REST API 인증 실패를 방지하려면 true로 설정합니다. 기본값은 false입니다. JBoss 버전 7.1 또는 7.2의 경우 이 속성을 설정하지 않아도 됩니다.
-Ddb2.jcc.charsetDecoderEncoder	MDM Hub 샘플 연산 참조 저장소를 사용하는 데 필요합니다. UTF-8 문자열이 아닌 바이트 시퀀스 대신에 유니코드 교체 문자(U+FFFD)를 반환하기 위한 JDBC 드라이버를 활성화합니다. 3으로 설정합니다.

Java 옵션	설명
-Djava.net.preferIPv4Stack	Java가 IPv4(인터넷 프로토콜 버전 4)를 사용할지 여부를 지정합니다. 운영 체제에서 IPv6(인터넷 프로토콜 버전 6)을 사용하는 경우 true로 설정합니다.
-Djavax.net.ssl.trustStore -Djavax.net.ssl.trustStorePassword	처리 서버에 HTTPS 포트를 사용하려는 경우 필요합니다. -Djavax.net.ssl.trustStore Java 옵션은 클라이언트 인증서 유효성 검사에 사용할 트러스트 저장소 파일에 대한 경로를 지정합니다. -Djavax.net.ssl.trustStorePassword Java 옵션은 트러스트 저장소 파일에 액세스하기 위한 암호를 지정합니다.
-Djava.security.egd	Linux 환경에서 Data Director의 시작 시간을 줄여줍니다. 값을 file:/dev/.urandom으로 설정합니다.
-Djboss.as.management.blocking.timeout	JBoss 배포를 기다리는 시간(초)입니다. JBoss가 시작되도록 하려면 값을 5000으로 설정합니다. 환경을 기반으로 시간을 조정합니다. 기본값은 300입니다. 매개 변수를 구성하지 않으면 JBoss 배포 시간 초과가 발생할 수 있습니다.
-Djgroups.bind_addr	JGroup이 메시지를 보내고 받아야 하는 인터페이스입니다. 다중 노드 또는 클러스터 환경에서 필요합니다. 각 노드가 고유한 네트워크 인터페이스에 바인딩되어야 합니다.
-DFrameworksLogConfigurationPath	log4j.xml 파일의 경로입니다.
-Dmdm.node.groupid	MDM Hub 구현에서 Java Virtual Machine에 대한 그룹 ID를 지정합니다. Hub 서버 및 처리 서버를 논리적으로 그룹화하려는 경우에만 필요합니다.
-DUseESLegacyFqSearch	필드가 지정된 검색에서 비즈니스 항목 유형에 대한 하위 노드 내의 정확한 일치 여부를 반환할지 여부를 지정합니다. 여러 필드에서 필드가 지정된 검색을 수행할 때만 적용됩니다. 하위 수준에 다수의 쿼리 필드가 있는 경우 동일한 하위 노드의 값이 포함된 레코드를 검색에서 반환해야 하는지 여부를 나타냅니다. 다른 하위 노드의 하위 수준 쿼리 필드와 일치할 수 있는 레코드를 반환하려면 true로 설정합니다. 기본값은 false입니다.
-Dfile.encoding -Dorg.apache.catalina.connector.URI_ENCODING	Informatica Data Director를 사용하고 REST API를 사용하여 레코드를 검색하려는 경우에 필요합니다. UTF-8 문자가 포함된 레코드를 찾아서 저장하려면 두 Java 옵션을 모두 UTF-8로 설정합니다.
- Dorg.apache.coyote.http11.Http11Protocol.MAX_HEADER_SIZE	HTTP 헤더의 최대 크기(바이트 단위)입니다. 헤더 크기가 작을 경우 검색 요청이 실패할 수 있습니다. 16384로 설정합니다.
-Dtask.pageSize=<최대 태스크 수>	각 요청에서 검색되는 최대 ActiveVOS 태스크 수를 지정합니다. 기본값은 5000입니다. 환경에 많은 수의 태스크가 있는 경우 수를 늘리십시오.

Java 옵션	설명
-Dstricttransportsecurity.flag	웹 브라우저에서 HTTP 요청을 사용한 모든 Data Director 액세스 시도를 HTTPS 요청으로 변환할지 여부를 지정합니다. true로 설정합니다.
-Xms	초기 힙 크기입니다. 2048m로 설정합니다.
-Xmx	최대 JVM 힙 크기입니다. 6GB 이상으로 설정합니다. 예를 들어 -Xmx를 6144m로 설정하려면 다음과 같은 JAVA_OPTIONS 환경 변수 설정을 사용합니다. set "JAVA_OPTIONS=-server ... -Xmx6144m"
XX:+UseCodeCacheFlushing	코드 캐시가 가득 찬 경우 컴파일된 코드의 JVM 처리 여부를 지정합니다.
-XX:ReservedCodeCacheSize	JIT 코드 캐시 크기입니다. MDM Hub 환경의 성능을 개선하려면 512m로 설정합니다.
-XX:MaxMetaspaceSize	최대 메타스페이스 크기입니다. JVM의 메모리 부족을 방지하려면 16로 설정하십시오.

Java Virtual Machine 논리적 그룹화의 예

JVM(Java Virtual Machine)를 그룹화하면 Hub 서버 및 처리 서버의 논리 그룹이 생성됩니다. Hub 서버 및 처리 서버 응용 프로그램을 논리적 JVM 그룹에 배포하면 Hub 서버와 처리 서버 응용 프로그램 간의 통신이 그룹 내에 유지됩니다. JVM을 그룹화하려면 MDM Hub 환경의 각 JVM에 그룹 ID를 할당합니다.

참고: 처리 서버 그룹화는 정리 및 일치 프로세스에만 적용됩니다. 논리 그룹은 MDM Hub의 내부 서버 캐시에 적용되지 않습니다.

다음 테이블에는 논리적 JVM 그룹의 예제가 나와 있습니다.

JVM 그룹	JVM	Hub 서버	처리 서버
Group1	JVM1	예	예
Group1	JVM4	-	예
Group2	JVM2	예	예
Group3	JVM3	-	예

JVM1의 경우 시작 스크립트에 다음 Java 옵션을 추가합니다.

```
-Dmdm.node.groupid=Group1
```

JVM2의 경우 시작 스크립트에 다음 Java 옵션을 추가합니다.

```
-Dmdm.node.groupid=Group2
```

JVM3의 경우 시작 스크립트에 다음 Java 옵션을 추가합니다.

```
-Dmdm.node.groupid=Group3
```


JVM4의 경우 시작 스크립트에 다음 Java 옵션을 추가합니다.

```
-Dmdm.node.groupid=Group1
```

JVM을 구성하고 Hub 서버 및 처리 서버를 배포한 후 그룹은 다음과 같은 특징을 갖습니다.

- Group1에는 처리 서버 2개가 포함되고 Group2에는 처리 서버 1개가 포함되며 Group3에는 처리 서버 1개가 포함됩니다.
- 모든 정리 및 일괄 호출은 검색을 제외하고 자체 그룹에 유지됩니다. 예를 들어 Group1의 Hub 서버에 대한 실시간 호출은 Group1 처리 서버(JVM1 및 JVM4)에만 영향을 미칩니다.

전체 프로필에 대한 서버 속성 구성

standalone-full.xml 파일에서 독립 실행형 모드의 전체 프로필에 대한 서버 속성을 구성합니다. 이 파일은 다음 디렉터리에 있습니다. <JBoss 설치 디렉터리>/standalone/configuration 서버 속성을 구성하려면 JBoss CLI(명령줄 인터페이스)에서 구성 명령을 실행할 수 있습니다.

JBoss CLI를 사용하려면 다음 단계를 수행하십시오.

1. 다음 디렉터리로 이동합니다. <JBoss 설치 디렉터리>/bin
2. JBoss CLI를 실행하려면 다음 스크립트를 실행합니다.
UNIX의 경우. jboss-cli.sh
Windows의 경우. jboss-cli.bat
3. 서버에 연결하려면 다음 명령을 실행합니다.

```
connect
```

서버 속성을 구성하는 방법에 대한 자세한 내용은 JBoss 설명서를 참조하십시오.

트랜잭션 제한 시간

트랜잭션 제한 시간 속성은 MDM Hub 트랜잭션이 완료될 때까지 대기하는 시간(초)을 지정합니다. 현재 환경에 따라 값을 설정하십시오.

명령:

```
/subsystem=transactions:write-attribute(name=default-timeout,value=<timeout in seconds>)
```

샘플 구성:

```
<subsystem xmlns="urn:jboss:domain:transactions:4.0">
  ...
  <coordinator-environment default-timeout="3600"/>
</subsystem>
```

최대 게시 크기

최대 게시 크기 속성은 업로드하는 파일의 최대 크기를 바이트 단위로 구성합니다. Data Director 응용 프로그램에 연결하려는 파일의 크기 제한에 값을 설정합니다.

값을 20000000 이상으로 설정하십시오. 기본값은 10000000입니다.

참고: MDM Hub를 설치한 후 cmxserver.properties 파일에서 Hub 서버 속성 cmx.file.max_file_size_mb에 동일한 값을 설정합니다.

명령:

```
/subsystem=undertow/server=default-server/<listener type=><listener name>/:write-attribute(name=max-post-size,value=<maximum file size in bytes>)
```

샘플 구성:

```
<subsystem xmlns="urn:jboss:domain:undertow:4.0">
  ...
  <server name="default-server">
```

```

        <http-listener name="default" socket-binding="http" redirect-socket="https" enable-http2="true"
max-post-size="2000000"/>
        <https-listener name="https" socket-binding="https" security-realm="ApplicationRealm" enable-
http2="true" max-post-size="2000000"/>
        ...
    </host>
</server>
...
</subsystem>

```

원격 커넥터

원격 커넥터 속성은 원격 커넥터 포트 및 소켓 바인딩을 구성합니다.

원격 시스템에서 Hub 콘솔에 로그인하려면 포트를 4447로 설정하고 소켓 바인딩을 remoting으로 설정합니다.

참고: 기본적으로 원격 커넥터 보안은 비활성화되어 있습니다. MDM Hub에 원격 커넥터 보안을 구성하려면 지원되는 Elytron 보안 프레임워크만 구성해야 합니다.

명령:

```

/socket-binding-group=standard-sockets/socket-binding=remoting:add(port=4447)

/subsystem=remoting/connector=remoting-connector:add(socket-binding=remoting)

```

원격 커넥터 포트의 샘플 구성:

```

<socket-binding-group name="standard-sockets" default-interface="public" port-offset="$
{jboss.socket.binding.port-offset:0}">
    ...
    <socket-binding name="remoting" port="4447"/>
    ...
</socket-binding-group>

```

원격 커넥터 소켓 바인딩의 샘플 구성:

```

<subsystem xmlns="urn:jboss:domain:remoting:4.0">
    ...
    <connector name="remoting-connector" socket-binding="remoting"/>
    ...
</subsystem>

```

ActiveVOS 콘솔 관리자 생성

ActiveVOS를 사용하려는 경우 응용 프로그램 서버 컨테이너에서 abAdmin 역할이 있는 ActiveVOS 콘솔 관리자를 생성합니다. ActiveVOS를 사용하려는 경우 abAdmin 역할이 있는 ActiveVOS 콘솔 관리자를 생성합니다. 관리자를 생성하지 않으면 Hub 서버 배포가 실패합니다. Hub 서버 설치 프로그램에서 ActiveVOS 콘솔의 관리자 자격 증명을 입력하라는 메시지를 표시하면 ActiveVOS 콘솔 관리자 이름과 암호를 사용합니다.

1. 다음 디렉터리로 변경합니다.

```
<JBoss 설치 디렉터리>/bin
```

2. add-user 유틸리티를 실행하려면 다음 스크립트를 사용합니다.

UNIX의 경우. add-user.sh

Windows의 경우. add-user.bat

3. 나타나는 프롬프트에 응답합니다.

다음 테이블에는 각 프롬프트에 대해 지정할 값이 설명되어 있습니다.

프롬프트	지정할 값
어떤 사용자 유형을 추가하시겠습니까? a) 관리 사용자 또는 b) 응용 프로그램 사용자	응용 프로그램 사용자를 선택하려면 b를 입력합니다.
영역(ApplicationRealm)	영역 이름. standalone-full.xml 파일에 추가한 login-module에서 지정한 영역 이름을 입력합니다.
사용자 이름	ActiveVOS 콘솔 관리자 이름.
암호	JBoss 암호 표준을 준수하는 암호.
이 사용자에게 어떤 역할을 지정하시겠습니까?	abAdmin.
영역 <영역 이름>에 대해 사용자 <사용자 이름>을 추가하려고 합니다. 맞습니까?	사용자를 추가하려면 yes를 입력합니다.
하나의 AS 프로세스가 다른 AS 프로세스에 연결하는 데 이 새로운 사용자가 사용됩니까?	yes.

4. WebSphere 콘솔에 로그인하고 ActiveVOS 콘솔 관리자를 생성합니다.

참고: ActiveVOS 콘솔 사용자는 설치 후 또는 업그레이드 후 프로세스 중 postInstallSetup 또는 patchInstallSetup 스크립트를 실행하면 abAdmin 역할에 매핑됩니다.

5. WebLogic 콘솔에 로그인합니다.
6. abAdmin 역할을 생성합니다.
7. ActiveVOS 콘솔 관리자를 생성합니다.
8. abAdmin 역할에 관리자 할당

JBoss 시작

Hub 서버 및 처리 서버를 설치하기 전에 JBoss 응용 프로그램 서버를 시작합니다. 환경을 기반으로 독립 실행형 JBoss 인스턴스 또는 JBoss 클러스터 노드를 시작합니다.

독립 실행형 JBoss 인스턴스 시작

독립 실행형 JBoss 인스턴스를 사용하는 경우 MDM Hub 구성 요소를 설치할 각 인스턴스를 시작합니다.

1. 다음 디렉터리로 이동합니다.

<JBoss 설치 디렉터리>/bin

2. JBoss 인스턴스를 시작하려면 다음 명령을 실행합니다.

UNIX의 경우. standalone.sh -c standalone-full.xml -b 0.0.0.0 -Djboss.as.management.blocking.timeout=5000

Windows의 경우. standalone.bat -c standalone-full.xml -b 0.0.0.0 -Djboss.as.management.blocking.timeout=5000

JBoss가 사용 가능한 네트워크 인터페이스에서 시작되고 /etc/hosts 디렉터리의 호스트 파일에 정의된 현재 호스트를 수신합니다. 환경을 기반으로 제한 시간 간격을 조정합니다.

3. 동일한 시스템에 여러 JBoss 인스턴스가 있는 경우 두 번째와 후속 JBoss 인스턴스를 시작하려면 시작 명령에 다음 인수를 추가합니다.

-Djboss.socket.binding.port-offset=<port offset range such as, 0,100,200,...n>

JBoss 클러스터 노드 시작

JBoss 클러스터 환경을 사용하는 경우 MDM Hub 구성 요소를 설치할 클러스터 노드를 시작합니다.

1. 다음 디렉터리로 이동합니다.

<JBoss 설치 디렉터리>/bin

2. JBoss 클러스터 노드를 시작하려면 클러스터 노드가 있는 시스템에서 다음 명령을 실행합니다.

UNIX의 경우. standalone.sh -c standalone-full.xml -b 0.0.0.0 -Djboss.node.name=<클러스터 노드 이름> -Djboss.server.base.dir=./<노드 경로> -Djboss.as.management.blocking.timeout=5000 -u <멀티캐스트 주소> -Djgroups.bind_addr=<바인딩 주소> -Djboss.socket.binding.port-offset=<포트 오프셋 값> -Djboss.partition.name=<파티션 이름>

Windows의 경우. standalone.bat -c standalone-full.xml -b 0.0.0.0 -Djboss.node.name=<클러스터 노드 이름> -Djboss.server.base.dir=./<노드 경로> -Djboss.as.management.blocking.timeout=5000 -u <멀티캐스트 주소> -Djgroups.bind_addr=<바인딩 주소> -Djboss.socket.binding.port-offset=<포트 오프셋 값> -Djboss.partition.name=<파티션 이름>

여러 클러스터 노드가 같은 시스템에서 실행될 경우 포트 오프셋 값을 설정합니다. 기본 파티션 이름을 사용하거나 클러스터에 속한 모든 노드의 파티션 이름이 동일한지 확인합니다. 환경을 기반으로 제한 시간 간격을 조정합니다.

Oracle WebLogic 설정

Oracle WebLogic 클러스터 환경 또는 독립 실행형 WebLogic 인스턴스에 MDM Hub를 설치할 수 있습니다. WebLogic 설명서의 지침에 따라 WebLogic을 설치 및 구성합니다.

중요: Multidomain MDM과 일부 Oracle WebLogic 버전(버전 12.2.1.3 등)의 상호 작용에는 몇 가지 알려진 제한 사항이 있습니다. 시작하기 전에 *Multidomain MDM 릴리스* 노트의 “설치 및 업그레이드” 장을 참조하십시오.

WebLogic 응용 프로그램 서버에 Hub 서버와 처리 서버를 설치하기 전에 WebLogic 도메인을 생성합니다. WebLogic 관리 콘솔을 사용하여 Hub 서버 및 처리 서버 응용 프로그램에 대한 도메인을 생성합니다. 자세한 내용은 WebLogic 설명서를 참조하십시오.

참고: 데이터베이스 서버와 동일한 시간대로 응용 프로그램 서버를 설치합니다.

Java Virtual Machine 구성

JVM(Java Virtual Machine)을 구성하려면 JAVA_OPTIONS 환경 변수를 사용하여 Java 옵션을 설정합니다.

다음과 같은 파일에서 Java 옵션을 설정할 수 있습니다.

UNIX의 경우. <WebLogic 도메인>/bin/setDomainEnv.sh

Windows의 경우. <WebLogic 도메인>\bin\setDomainEnv.cmd

다음 테이블에는 Java 옵션이 설명되어 있습니다.

Java 옵션	설명
-server	시작은 느리지만 후속 작업은 빨라집니다.
-Djgroups.bind_addr	JGroup이 메시지를 보내고 받아야 하는 인터페이스입니다. 다중 노드 또는 클러스터된 환경에서 필요합니다. 각 노드가 고유한 네트워크 인터페이스에 바인딩되어야 합니다.
-Djava.net.preferIPv4Stack	Java가 IPv4(인터넷 프로토콜 버전 4)를 사용할지 여부를 지정합니다. 운영 체제에서 IPv6(인터넷 프로토콜 버전 6)을 사용하는 경우 true로 설정합니다.
-Djava.security.egd	Linux 환경에서 Informatica Data Director의 시작 시간을 줄여줍니다. 값을 다음으로 설정합니다. Djava.security.egd=file:/dev/./urandom
-Doracle.jdbc.J2EE13Compliant	oracle.jdbc.J2EE13Compliant 시스템 변수를 설정합니다. true로 설정합니다. 매개 변수를 true로 설정하지 않으면 JDBC(Java Database Connectivity) 문제가 발생할 수 있습니다.
-Djavax.wsdl.factory.WSDLFactory	WebLogic 12.2.1 이상 환경에서 기본 네임스페이스 접두사가 포함되었을 수 있는 서비스 WSDL을 호출하는 데 필요합니다. com.ibm.wsdl.factory.WSDLFactoryImpl로 설정합니다.
-DANTLR_USE_DIRECT_CLASS_LOADING	MDM Hub가 배포된 WebLogic 12.2.1 이상 환경을 시작하는 데 필요합니다. true로 설정합니다.
-Dmdm.node.groupid	MDM Hub 구현에서 Java Virtual Machine에 대한 그룹 ID를 지정합니다. Hub 서버 및 처리 서버를 논리적으로 그룹화하려는 경우에만 필요합니다.
-De360.connection.channel	응용 프로그램 서버에서 Entity 360 프레임워크에 기반한 Hub 서버 및 Data Director 화면 간 통신에 사용해야 하는 통신 프로토콜입니다. 유효한 값은 HTTP 및 HTTPS입니다. 기본값은 HTTP입니다. 다음과 같은 지원 Java 옵션을 설정했는지 확인합니다. - -De360.mdm.host - -De360.mdm.port
-DUseESLegacyFqSearch	필드가 지정된 검색에서 비즈니스 항목 유형에 대한 하위 노드 내의 정확한 일치 여부를 지정합니다. 여러 필드에서 필드가 지정된 검색을 수행할 때만 적용됩니다. 하위 수준에 다수의 쿼리 필드가 있는 경우 동일한 하위 노드의 값이 포함된 레코드를 검색에서 반환해야 하는지 여부를 나타냅니다. 다른 하위 노드의 하위 수준 쿼리 필드와 일치할 수 있는 레코드를 반환하려면 true로 설정합니다. 기본값은 false입니다.

Java 옵션	설명
-Dweblogic.security.SSL.trustedCAKeyStore -Dweblogic.security.SSL.enable.renegotiation -Dweblogic.security.SSL.verbose	<p>WebLogic에 대해 HTTPS가 활성화된 경우 필요합니다.</p> <p>-Dweblogic.security.SSL.trustedCAKeyStore Java 옵션은 트러스트된 인증서가 포함된 키 저장소의 경로를 지정합니다. 절대 키 저장소 경로로 설정합니다.</p> <p>-Dweblogic.security.SSL.enable.renegotiation Java 옵션은 SSL 재협상을 활성화합니다. true로 설정합니다.</p> <p>-Dweblogic.security.SSL.verbose Java 옵션은 추가적인 SSL 디버깅을 활성화합니다. true로 설정합니다.</p>
-De360.mdm.host	<p>응용 프로그램 서버 호스트입니다.</p> <p>환경에서 HTTPS 통신 프로토콜을 사용하고 보안 인증서가 FQDN(정규화된 도메인 이름)으로 발행되는 경우 FQDN으로 설정합니다.</p> <p>다음과 같은 시나리오 중 하나에서 Java 옵션을 설정합니다.</p> <ul style="list-style-type: none"> - MDM Hub 응용 프로그램을 관리되는 서버에만 배포하려면 각 관리되는 서버의 스크립트에 Java 옵션을 추가합니다. 관리되는 서버의 IP 주소 또는 기호 이름으로 설정합니다. - 7001이 아닌 WebLogic 포트에 MDM Hub 응용 프로그램을 배포하려면 응용 프로그램 서버 시작 스크립트에 Java 옵션을 추가합니다. WebLogic 호스트의 IP 주소 또는 기호 이름으로 설정합니다. <p>이 매개 변수를 구성하지 않으면 Entity 360 프레임워크에 기반한 Data Director 화면이 예상대로 작동하지 않을 수 있습니다.</p>
-De360.mdm.port	<p>응용 프로그램 서버 포트입니다.</p> <p>다음과 같은 시나리오 중 하나에서 Java 옵션을 설정합니다.</p> <ul style="list-style-type: none"> - MDM Hub 응용 프로그램을 관리되는 서버에만 배포하려면 각 관리되는 서버의 스크립트에 Java 옵션을 추가합니다. 관리되는 서버의 포트 번호로 설정합니다. - 7001이 아닌 WebLogic 포트에 MDM Hub 응용 프로그램을 배포하려면 응용 프로그램 서버 시작 스크립트에 Java 옵션을 추가하고 포트 번호로 설정합니다. <p>이 매개 변수를 구성하지 않으면 Entity 360 프레임워크에 기반한 Data Director 화면이 예상대로 작동하지 않을 수 있습니다.</p>
-Didd.mdm.host -Didd.mdm.port -Didd.protocol	<p>제목 영역이 있는 Data Director에 필요합니다.</p> <p>제목 영역이 있는 Data Director를 배포하려면 다음 Java 옵션을 설정합니다.</p> <ul style="list-style-type: none"> - -Didd.mdm.host. 응용 프로그램 서버 호스트의 호스트 이름 또는 IP 주소로 설정합니다. - -Didd.mdm.port. 기본값이 아닌 HTTP 또는 HTTPS 수신기 포트를 사용하여 배포하는 데 필요합니다. 이 속성은 서버 초기화 중 제목 영역 응용 프로그램을 배포하는 데 사용됩니다. 응용 프로그램 JVM에 사용되는 HTTP 또는 HTTPS 수신기 포트를 지정합니다. 기본값은 8080입니다. - -Didd.protocol. 서버 초기화 중 제목 영역 응용 프로그램을 배포하는 데 사용되는 필수 속성입니다. 사용할 통신 프로토콜을 HTTP 또는 HTTPS로 지정합니다. 기본값은 HTTP입니다.
-Dfile.encoding -Dweblogic.http.URIDecodeEncoding	<p>Data Director를 사용하고 REST API를 사용하여 레코드를 검색하려는 경우에 필요합니다.</p> <p>UTF-8 문자가 포함된 레코드를 찾아서 저장하려면 두 Java 옵션을 모두 UTF-8로 설정합니다.</p>
-DframeworksLogConfigurationPath	log4j.xml 파일 구성 경로를 설정합니다.

Java 옵션	설명
-Dtask.pageSize=<최대 태스크 수>	각 요청에서 검색되는 최대 ActiveVOS 태스크 수를 지정합니다. 기본값은 5000입니다. 환경에 많은 수의 태스크가 있는 경우 수를 늘리십시오.
-Dstricttransportsecurity.flag	웹 브라우저에서 HTTP 요청을 사용한 모든 Data Director 액세스 시도를 HTTPS 요청으로 변환할지 여부를 지정합니다. true로 설정합니다.
WLS_MEM_ARGS_64BIT	메모리 변수입니다. 변수를 적절한 값으로 설정합니다.
-Xms	초기 힙 크기입니다. 2048m로 설정합니다.
-Xmx	최대 JVM 힙 크기입니다. 6GB 이상으로 설정합니다. 예를 들어 -Xmx를 6144m로 설정하려면 다음과 같은 JAVA_OPTIONS 환경 변수 설정을 사용합니다. set "JAVA_OPTIONS=-server ... -Xmx6144m"
XX:+UseCodeCacheFlushing	코드 캐시가 가득 찬 경우 컴파일된 코드의 JVM 처리 여부를 지정합니다.
-XX:ReservedCodeCacheSize	JIT 코드 캐시 크기입니다. MDM Hub 환경의 성능을 개선하려면 512m로 설정합니다.

Java Virtual Machine 논리적 그룹화의 예

JVM(Java Virtual Machine)를 그룹화하면 Hub 서버 및 처리 서버의 논리 그룹이 생성됩니다. Hub 서버 및 처리 서버 응용 프로그램을 논리적 JVM 그룹에 배포하면 Hub 서버와 처리 서버 응용 프로그램 간의 통신이 그룹 내에 유지됩니다. JVM을 그룹화하려면 MDM Hub 환경의 각 JVM에 그룹 ID를 할당합니다.

참고: 처리 서버 그룹화는 정리 및 일치 프로세스에만 적용됩니다. 논리 그룹은 MDM Hub의 내부 서버 캐시에 적용되지 않습니다.

다음 테이블에는 논리적 JVM 그룹의 예제가 나와 있습니다.

JVM 그룹	JVM	Hub 서버	처리 서버
Group1	JVM1	예	예
Group1	JVM4	-	예
Group2	JVM2	예	예
Group3	JVM3	-	예

JVM1의 경우 시작 스크립트에 다음 Java 옵션을 추가합니다.

```
-Dmdm.node.groupid=Group1
```

JVM2의 경우 시작 스크립트에 다음 Java 옵션을 추가합니다.

```
-Dmdm.node.groupid=Group2
```

JVM3의 경우 시작 스크립트에 다음 Java 옵션을 추가합니다.

```
-Dmdm.node.groupid=Group3
```

JVM4의 경우 시작 스크립트에 다음 Java 옵션을 추가합니다.

```
-Dmdm.node.groupid=Group1
```

JVM을 구성하고 Hub 서버 및 처리 서버를 배포한 후 그룹은 다음과 같은 특징을 갖습니다.

- Group1에는 처리 서버 2개가 포함되고 Group2에는 처리 서버 1개가 포함되며 Group3에는 처리 서버 1개가 포함됩니다.
- 모든 정리 및 일괄 호출은 검색을 제외하고 자체 그룹에 유지됩니다. 예를 들어 Group1의 Hub 서버에 대한 실시간 호출은 Group1 처리 서버(JVM1 및 JVM4)에만 영향을 미칩니다.

WebLogic 서버 인증 비활성화

MDM Hub는 HTTP 기본 인증을 사용합니다. 따라서 WebLogic 서버 인증을 비활성화해야 합니다. WebLogic 서버 인증을 비활성화하려면 config.xml 파일을 편집합니다.

1. 다음 WebLogic 디렉터리로 이동합니다.

```
<WebLogic 설치 디렉터리>/user_projects/domains/<user domain>/config
```

2. 텍스트 편집기에서 config.xml 파일을 엽니다.

3. <security-configuration> 요소 내에 다음 요소를 추가합니다.

```
<enforce-valid-basic-auth-credentials>
false
</enforce-valid-basic-auth-credentials>
```

ActiveVOS 콘솔 관리자 생성

ActiveVOS를 사용하려는 경우 응용 프로그램 서버 컨테이너에서 abAdmin 역할이 있는 ActiveVOS 콘솔 관리자를 생성합니다. ActiveVOS를 사용하려는 경우 abAdmin 역할이 있는 ActiveVOS 콘솔 관리자를 생성합니다. 관리자를 생성하지 않으면 Hub 서버 배포가 실패합니다. Hub 서버 설치 프로그램에서 ActiveVOS 콘솔의 관리자 자격 증명을 입력하라는 메시지를 표시하면 ActiveVOS 콘솔 관리자 이름과 암호를 사용합니다.

1. 다음 디렉터리로 변경합니다.

```
<JBoss 설치 디렉터리>/bin
```

2. add-user 유틸리티를 실행하려면 다음 스크립트를 사용합니다.

UNIX의 경우. add-user.sh

Windows의 경우. add-user.bat

3. 나타나는 프롬프트에 응답합니다.

다음 테이블에는 각 프롬프트에 대해 지정할 값이 설명되어 있습니다.

프롬프트	지정할 값
어떤 사용자 유형을 추가하시겠습니까? a) 관리 사용자 또는 b) 응용 프로그램 사용자	응용 프로그램 사용자를 선택하려면 b를 입력합니다.
영역(ApplicationRealm)	영역 이름. standalone-full.xml 파일에 추가한 login-module에서 지정한 영역 이름을 입력합니다.
사용자 이름	ActiveVOS 콘솔 관리자 이름.

프롬프트	지정할 값
암호	JBoss 암호 표준을 준수하는 암호.
이 사용자에게 어떤 역할을 지정하시겠습니까?	abAdmin.
영역 <영역 이름>에 대해 사용자 <사용자 이름>을 추가하려고 합니다. 맞습니까?	사용자를 추가하려면 yes를 입력합니다.
하나의 AS 프로세스가 다른 AS 프로세스에 연결하는 데 이 새로운 사용자가 사용됩니까?	yes.

4. WebSphere 콘솔에 로그인하고 ActiveVOS 콘솔 관리자를 생성합니다.

참고: ActiveVOS 콘솔 사용자는 설치 후 또는 업그레이드 후 프로세스 중 postInstallSetup 또는 patchInstallSetup 스크립트를 실행하면 abAdmin 역할에 매핑됩니다.

5. WebLogic 콘솔에 로그인합니다.
6. abAdmin 역할을 생성합니다.
7. ActiveVOS 콘솔 관리자를 생성합니다.
8. abAdmin 역할에 관리자 할당

추가 Oracle WebLogic 구성

MDM Hub 환경의 요구 사항을 기반으로 추가 WebLogic 구성을 수행합니다.

다음 테이블에는 수행할 수 있는 구성이 설명되어 있습니다.

구성	설명
독립 실행형 처리 서버 인스턴스에 대한 WebLogic 구성	다음과 같은 시나리오에서 독립 실행형 처리 서버 인스턴스에 대한 WebLogic을 구성하는 데 필요합니다. - Hub 서버가 설치되지 않은 WebLogic 인스턴스에 처리 서버 인스턴스를 설치하려고 합니다. - 여러 독립 실행형 처리 서버 인스턴스를 설치하려고 합니다.
여러 MDM Hub 마스터 데이터베이스에 대한 WebLogic 구성	여러 MDM Hub 마스터 데이터베이스 인스턴스를 구성하려는 경우에 필요합니다.
HTTPS 프로토콜 구성	MDM Hub 통신을 위한 HTTPS 프로토콜을 구성하려는 경우에 필요합니다.

독립 실행형 처리 서버 인스턴스에 대한 WebLogic 구성

여러 독립 실행형 처리 서버 인스턴스를 설치하려는 경우 적절한 데이터 소스를 사용하도록 WebLogic을 구성합니다. 또한 Hub 서버가 설치되지 않은 WebLogic 인스턴스에 처리 서버 인스턴스를 설치하려는 경우 데이터 소스를 구성합니다.

적절한 데이터 소스를 사용하도록 WebLogic을 구성하려면 다음과 같은 작업을 수행합니다.

1. JDBC 드라이버를 설치합니다.
2. MDM Hub 마스터 데이터베이스 데이터 소스를 생성합니다.

3. 연산 참조 저장소 데이터 소스를 생성합니다.

1단계. JDBC 드라이버 설치

MDM Hub 마스터 데이터베이스 및 ORS(연산 참조 저장소)에 대한 데이터 소스를 생성하기 전에 JDBC 드라이버를 설치합니다.

지원되는 버전의 JDBC 드라이버를 가져오려면 Oracle에 문의합니다.

지원되는 버전의 JDBC 드라이버를 가져오려면 Microsoft에 문의합니다.

지원되는 버전의 JDBC 드라이버를 가져오려면 IBM에 문의합니다.

1. JDBC 드라이버를 다음 디렉터리에 복사합니다.

<WebLogic 설치 디렉터리>/wlserver/server/lib

2. JDBC 드라이버의 경로를 다음 파일의 CLASSPATH 변수에 추가합니다.

UNIX의 경우. <WebLogic 도메인>/bin/commEnv.sh

Windows의 경우. <WebLogic 도메인>\bin\commEnv.cmd

참고: 다른 WebLogic 서버 라이브러리의 경로 앞에 JDBC 드라이버 경로를 배치합니다.

2단계. MDM Hub 마스터 데이터베이스 데이터 소스 생성

처리 서버 시스템에 JDBC 드라이버를 설치한 후 MDM Hub 마스터 데이터베이스에 대한 데이터 소스를 생성합니다.

1. WebLogic 관리 콘솔에서 **잠금 및 편집** 단추를 클릭하여 잠금을 획득합니다.
2. **서비스 > JDBC > 데이터 소스**를 클릭한 다음 **새로 만들기**를 클릭합니다.
JDBC 데이터 소스 속성 페이지가 표시됩니다.
3. 다음 데이터 소스 속성을 지정합니다.

속성	설명
이름	JDBC 데이터 소스의 이름입니다. 이름을 MDM Master Data Source로 설정합니다.
JNDI 이름	JDBC 데이터 소스를 바인딩할 JNDI 경로입니다. jdbc/siperian-cmx_system-ds를 지정합니다.
데이터베이스 유형	연결할 데이터베이스 유형입니다. Oracle을 선택합니다. MS SQL Server를 선택합니다.
데이터베이스 드라이버	데이터베이스에 연결하는 데 사용할 JDBC 드라이버입니다. Oracle 드라이버(Thin XA) 를 선택합니다. MS SQL Server 드라이버(Type 4 XA) 버전: 2005 이상 을 선택합니다.

4. **다음**을 클릭하고 다시 **다음**을 클릭합니다.
연결 속성 페이지가 표시됩니다.

5. 다음 연결 속성의 값을 입력합니다.

속성	설명
데이터베이스 이름	연결할 데이터베이스의 이름입니다.
호스트 이름	데이터베이스를 호스팅하는 서버의 IP 주소 또는 DNS 이름입니다. 7001 이외의 WebLogic 포트에 MDM Hub 응용 프로그램을 배포하려면 WebLogic 호스트의 호스트 이름과 IP 주소를 설정합니다. localhost는 사용하지 마십시오.
포트	데이터베이스 서버가 연결 요청을 수신하는 포트입니다.
데이터베이스 사용자 이름	데이터 소스의 각 연결에 대해 사용할 데이터베이스 사용자 이름입니다.
암호	데이터베이스 사용자 계정의 암호입니다.
암호 확인	데이터베이스 사용자 계정의 암호입니다.

6. 다음을 클릭합니다.

데이터베이스 연결 테스트 페이지가 표시됩니다.

7. 구성 테스트를 클릭하여 드라이버 연결을 테스트합니다.

테스트가 실패하면 연결 속성 페이지의 값을 업데이트한 다음 연결이 성공할 때까지 다시 시도해야 합니다.

8. 다음을 클릭한 다음 데이터 소스를 배포할 서버를 선택합니다.
9. 마침을 클릭한 다음 변경 내용 활성화를 클릭합니다.

3단계. 연산 참조 저장소 데이터 소스 생성

처리 서버 시스템에서 각 연산 참조 저장소에 대한 데이터 소스를 생성합니다.

1. WebLogic 관리 콘솔에서 **잠금 및 편집** 단추를 클릭하여 잠금을 획득합니다.
2. 서비스 > **JDBC** > 데이터 소스를 클릭한 다음 **새로 만들기**를 클릭합니다.
JDBC 데이터 소스 속성 페이지가 표시됩니다.
3. 다음 데이터 소스 속성을 지정합니다.

속성	설명
이름	JDBC 데이터 소스의 이름입니다. 이름을 MDM ORS Data Source로 설정합니다.
JNDI 이름	JDBC 데이터 소스를 바인딩할 JNDI 경로입니다. jdbc/siperian-<oracle 호스트 이름>-<oracle sid>-<연산 참조 저장소 이름>-ds를 지정합니다. jdbc/siperian-<Microsoft SQL Server 호스트 이름>-<연산 참조 저장소 이름>-ds를 지정합니다.
데이터베이스 유형	연결할 데이터베이스 유형입니다. Oracle을 선택합니다. MS SQL Server를 선택합니다.
데이터베이스 드라이버	데이터베이스에 연결하는 데 사용할 JDBC 드라이버입니다. Oracle 드라이버 (Thin XA)를 선택합니다. MS SQL Server 드라이버 (Type 4 XA) 버전 : 2005 이상을 선택합니다.

4. 다음을 클릭하고 다시 다음을 클릭합니다.

연결 속성 페이지가 표시됩니다.

5. 다음 연결 속성의 값을 입력합니다.

속성	설명
데이터베이스 이름	연결할 데이터베이스의 이름입니다.
호스트 이름	데이터베이스를 호스팅하는 서버의 IP 주소 또는 DNS 이름입니다. 7001 이외의 WebLogic 포트에 MDM Hub 응용 프로그램을 배포하려면 WebLogic 호스트의 호스트 이름과 IP 주소를 설정합니다. localhost는 사용하지 마십시오.
포트	데이터베이스 서버가 연결 요청을 수신하는 포트입니다.
데이터베이스 사용자 이름	데이터 소스의 각 연결에 대해 사용할 데이터베이스 사용자 이름입니다.
암호	데이터베이스 사용자 계정의 암호입니다.
암호 확인	데이터베이스 사용자 계정의 암호입니다.

6. 다음을 클릭합니다.

데이터베이스 연결 테스트 페이지가 표시됩니다.

7. 구성 테스트를 클릭하여 드라이버 연결을 테스트합니다.

테스트가 실패하면 연결 속성 페이지의 값을 업데이트한 다음 연결이 성공할 때까지 다시 시도해야 합니다.

8. 다음을 클릭한 다음 데이터 소스를 배포할 서버를 선택합니다.

9. 마침을 클릭한 다음 변경 내용 활성화를 클릭합니다.

여러 MDM Hub 마스터 데이터베이스에 대한 WebLogic 구성

여러 MDM Hub 마스터 데이터베이스 인스턴스를 구성하려는 경우 MDM Hub 마스터 데이터베이스 인스턴스 수 만큼 많은 WebLogic 도메인을 구성합니다. 각 MDM Hub 마스터 데이터베이스 인스턴스에는 고유한 MDM Hub 인스턴스가 있어야 합니다. 따라서 별도의 WebLogic 도메인에 각 MDM Hub 인스턴스를 배포할 만큼 많은 WebLogic 도메인을 생성합니다.

HTTPS 프로토콜 구성

MDM Hub 통신을 위해 HTTPS 프로토콜을 구성할 수 있습니다. WebLogic 서버 관리 콘솔을 사용하여 HTTPS 프로토콜을 구성합니다. 또는 기본 JDK 보안 인증서를 사용하여 HTTPS를 활성화할 수도 있습니다.

참고: 키 저장소를 구성하기 전에 Java Keytool 또는 OpenSSL을 사용하여 WebLogic 서버에 대해 자체 서명된 인증서를 생성합니다.

1. WebLogic 서버 관리 콘솔의 **환경** 섹션에서 **서버**를 클릭합니다.
2. **서버 요약** 페이지의 서버 목록에서 **AdminServer(관리)** 링크를 클릭합니다.
3. **SSL 수신 포트 활성화됨** 옵션을 활성화하고 SSL 수신 포트 번호를 입력합니다.
4. **SSL** 탭을 클릭하고 **호스트 이름 확인**에서 **없음**을 클릭합니다.
5. ID 및 트러스트 키 저장소를 구성할 서버의 이름을 선택합니다.
6. **구성 > 키 저장소**를 클릭합니다.

7. 다음 구성 옵션의 암호를 변경합니다.

구성 옵션	설명
데모 ID 키 저장소 암호	데모 ID 키 저장소의 암호화된 암호입니다.
데모 트러스트 키 저장소 암호	데모 트러스트 키 저장소의 암호화된 암호입니다.
Java 표준 트러스트 키 저장소 암호	Java 표준 트러스트 키 저장소의 암호입니다.

8. 변경 내용을 저장합니다.

IBM WebSphere 설정

IBM WebSphere 클러스터 환경 또는 독립 실행형 WebSphere 인스턴스에 MDM Hub를 설치할 수 있습니다. WebSphere 설명서의 지침에 따라 WebSphere를 설치 및 구성합니다.

WebSphere 설치 디렉터리 경로에 공백이 없는지 확인합니다.

참고: 데이터베이스 서버와 동일한 시간대로 응용 프로그램 서버를 설치합니다.

다음 테이블에는 설치하기 전에 구성할 속성 및 해당 값과 함께 속성을 설정해야 할 위치에 대한 간략한 설명이 나열되어 있습니다.

사용자 지정 속성	값	설명
com.ibm.ws.scripting.echoparams	false	다음 디렉터리에 있는 wsadmin.properties 파일에서 이 속성을 설정합니다. <WebSphere 설치 디렉터리>\WebSphere\AppServer\profiles\<프로필 이름>\properties ActiveVOS 데이터베이스 암호가 patchinstallSetup.log 파일의 스크립트 텍스트에 표시되지 않도록 하려면 이 값을 false로 설정합니다. 기본값은 true입니다.

Java Virtual Machine 구성

JVM(Java Virtual Machine)을 구성하려면 JAVA_OPTIONS 환경 변수를 사용하여 Java 옵션을 설정합니다. Java 옵션을 편집하거나 추가한 후 JVM을 다시 시작합니다.

WebSphere 클러스터된 환경을 사용하는 경우 다음과 같은 클러스터 구성 요소에 대해 Java 옵션을 설정합니다.

- 서버, 클러스터의 각 서버에서 모든 필수 Java 옵션을 설정합니다.
- 배포 관리자, 모든 필수 Java 옵션을 설정합니다.
- 노드 에이전트, -Xmx 및 -Xms Java 옵션을 사용하여 힙 크기만 설정합니다.

다음 테이블에는 **Java** 옵션 설정이 설명되어 있습니다.

Java 옵션	설명
-server	시작은 느리지만 후속 작업은 빨라집니다.
-Djava.net.preferIPv4Stack	Java가 IPv4(인터넷 프로토콜 버전 4)를 사용할지 여부를 지정합니다. 운영 체제에서 IPv6(인터넷 프로토콜 버전 6)을 사용하는 경우 true로 설정합니다.
-Djava.security.egd	Linux 환경에서 Data Director의 시작 시간을 줄여줍니다. 값을 file:/dev/./urandom으로 설정합니다.
-DUseESLegacyFqSearch	필드가 지정된 검색에서 비즈니스 항목 유형에 대한 하위 노드 내의 정확한 일치율 반환할지 여부를 지정합니다. 여러 필드에서 필드가 지정된 검색을 수행할 때만 적용됩니다. 하위 수준에 다수의 쿼리 필드가 있는 경우 동일한 하위 노드의 값이 포함된 레코드를 검색에서 반환해야 하는지 여부를 나타냅니다. 다른 하위 노드의 하위 수준 쿼리 필드와 일치할 수 있는 레코드를 반환하려면 true로 설정합니다. 기본값은 false입니다.
-Ddb2.jcc.charsetDecoderEncoder	MDM Hub 샘플 연산 참조 저장소를 사용하는 데 필요합니다. UTF-8 문자열이 아닌 바이트 시퀀스 대신에 유니코드 교체 문자(U+FFFD)를 반환하기 위한 JDBC 드라이버를 활성화합니다. 3으로 설정합니다.
-Dcom.ibm.crypto.provider.DoRSATypeChecking	Java에서 개인 키를 사용한 데이터의 RSA 유형 암호화 및 공개 키를 사용한 암호 해독을 허용할지 여부를 지정합니다. MDM Hub 설치 프로그램에서 라이선스 인증서를 읽을 수 있고, MDM Hub에서 암호 해시가 작동하도록 하려면 필요합니다. false로 설정합니다. -Dcom.ibm.crypto.provider.DoRSATypeChecking을 false로 설정하지 않으면 Hub 서버가 시작되지 않고 라이선스 오류가 발생할 수 있습니다.
-Djgroups.bind_addr	JGroup이 메시지를 보내고 받아야 하는 인터페이스입니다. 다중 노드 또는 클러스터된 환경에서 필요합니다. 각 노드가 고유한 네트워크 인터페이스에 바인딩되어야 합니다.
-De360.mdm.host -De360.mdm.port -De360.connection.channel	응용 프로그램 서버 통신 프로토콜, 호스트 및 포트입니다. 2809 이외의 Bootstrap 포트에 MDM Hub 응용 프로그램을 배포하려면 다음 Java 옵션을 설정합니다. - -De360.connection.channel. 사용할 통신 프로토콜로 설정합니다. 유효한 값은 HTTP 및 HTTPS입니다. 기본값은 HTTP입니다. - -De360.mdm.host. WebSphere 호스트의 IP 주소로 설정합니다. 환경에서 HTTPS 통신 프로토콜을 사용하고 보안 인증서가 FQDN(정규화된 도메인 이름)으로 발행되는 경우 FQDN으로 설정합니다. - -De360.mdm.port. 2809 대신 구성된 WebSphere Bootstrap 포트 설정합니다. 이 매개 변수를 구성하지 않으면 Entity 360 프레임워크에 기반한 Data Director 화면이 예상대로 작동하지 않을 수 있습니다.

Java 옵션	설명
-Didd.mdm.host -Didd.mdm.port -Didd.protocol	<p>제목 영역이 있는 Data Director에 필요합니다.</p> <p>제목 영역이 있는 Data Director를 배포하려면 다음 Java 옵션을 설정합니다.</p> <ul style="list-style-type: none"> -Didd.mdm.host. 응용 프로그램 서버 호스트의 호스트 이름 또는 IP 주소로 설정합니다. -Didd.mdm.port. 기본값이 아닌 HTTP 또는 HTTPS 수신기 포트를 사용하여 배포하는 데 필요합니다. 이 속성은 서버 초기화 중 제목 영역 응용 프로그램을 배포하는 데 사용됩니다. 응용 프로그램용 JVM에 사용되는 HTTP 또는 HTTPS 수신기 포트를 지정합니다. 기본값은 8080입니다. -Didd.protocol. 서버 초기화 중 제목 영역 응용 프로그램을 배포하는 데 사용되는 필수 속성입니다. 사용할 통신 프로토콜을 HTTP 또는 HTTPS로 지정합니다. 기본값은 HTTP입니다.
-DFrameworksLogConfigurationPath	log4j.xml 파일의 경로입니다.
-Doracle.jdbc.J2EE13Compliant	<p>Oracle 드라이버의 시스템 변수가 J2EE와 완전하게 호환될지 여부를 지정합니다. true로 설정합니다.</p> <p>매개 변수를 true로 설정하지 않으면 JDBC(Java Database Connectivity) 문제가 발생할 수 있습니다.</p>
-Dmdm.node.groupid	MDM Hub 구현에서 Java Virtual Machine에 대한 그룹 ID를 지정합니다. Hub 서버 및 처리 서버를 논리적으로 그룹화하려는 경우에만 필요합니다.
-Dfile.encoding -Dclient.encoding.override	<p>Informatica Data Director를 사용하고 REST API를 사용하여 레코드를 검색하려는 경우에 필요합니다.</p> <p>UTF-8 문자가 포함된 레코드를 찾아서 저장하려면 두 Java 옵션을 모두 UTF-8로 설정합니다.</p>
-Dstricttransportsecurity.flag	웹 브라우저에서 HTTP 요청을 사용한 모든 Data Director 액세스 시도를 HTTPS 요청으로 변환할지 여부를 지정합니다. true로 설정합니다.
-XX:codecachetotal	JIT 코드 캐시 크기입니다. MDM Hub 환경의 성능을 개선하려면 512m로 설정합니다.
-Xmx	<p>최대 JVM 힙 크기입니다. 6GB 이상으로 설정합니다.</p> <p>예를 들어 -Xmx를 6144m로 설정하려면 다음과 같은 JAVA_OPTIONS 환경 변수 설정을 사용합니다.</p> <pre>set "JAVA_OPTIONS=-server ... -Xmx6144m"</pre>
-Xms	초기 힙 크기입니다. 2048m로 설정합니다.
-Xmso	처리 서버 JVM에 필요합니다. 운영 체제 스레드에 대한 초기 스택 크기입니다. 낮은 시스템 스레드 스택 크기로 인해 응용 프로그램 서버가 예기치 않게 종료되는 것을 방지합니다. 4096k로 설정합니다.
-Xss	초기 스택 크기입니다. 2000k로 설정합니다.

Java 옵션	설명
XX:+UseCodeCacheFlushing	코드 캐시가 가득 찬 경우 컴파일된 코드의 JVM 처리 여부를 지정합니다.
-Dtask.pageSize=<최대 태스크 수>	각 요청에서 검색되는 최대 ActiveVOS 태스크 수를 지정합니다. 기본값은 5000입니다. 환경에 많은 수의 태스크가 있는 경우 수를 늘리십시오.

Java Virtual Machine 논리적 그룹화의 예

JVM(Java Virtual Machine)를 그룹화하면 Hub 서버 및 처리 서버의 논리 그룹이 생성됩니다. Hub 서버 및 처리 서버 응용 프로그램을 논리적 JVM 그룹에 배포하면 Hub 서버와 처리 서버 응용 프로그램 간의 통신이 그룹 내에 유지됩니다. JVM을 그룹화하려면 MDM Hub 환경의 각 JVM에 그룹 ID를 할당합니다.

참고: 처리 서버 그룹화는 정리 및 일치 프로세스에만 적용됩니다. 논리 그룹은 MDM Hub의 내부 서버 캐시에 적용되지 않습니다.

다음 테이블에는 논리적 JVM 그룹의 예제가 나와 있습니다.

JVM 그룹	JVM	Hub 서버	처리 서버
Group1	JVM1	예	예
Group1	JVM4	-	예
Group2	JVM2	예	예
Group3	JVM3	-	예

JVM1의 경우 시작 스크립트에 다음 Java 옵션을 추가합니다.

```
-Dmdm.node.groupid=Group1
```

JVM2의 경우 시작 스크립트에 다음 Java 옵션을 추가합니다.

```
-Dmdm.node.groupid=Group2
```

JVM3의 경우 시작 스크립트에 다음 Java 옵션을 추가합니다.

```
-Dmdm.node.groupid=Group3
```

JVM4의 경우 시작 스크립트에 다음 Java 옵션을 추가합니다.

```
-Dmdm.node.groupid=Group1
```

JVM을 구성하고 Hub 서버 및 처리 서버를 배포한 후 그룹은 다음과 같은 특징을 갖습니다.

- Group1에는 처리 서버 2개가 포함되고 Group2에는 처리 서버 1개가 포함되며 Group3에는 처리 서버 1개가 포함됩니다.
- 모든 정리 및 일괄 호출은 검색을 제외하고 자체 그룹에 유지됩니다. 예를 들어 Group1의 Hub 서버에 대한 실시간 호출은 Group1 처리 서버(JVM1 및 JVM4)에만 영향을 미칩니다.

MDM Hub 환경에서 암호 암호화

MDM Hub 환경에서 로그 파일에 표시되는 암호와 같은 중요 데이터를 암호화하려면 WebSphere에서 스크립팅 관리를 구성합니다.

1. 다음 디렉터리에서 wsadmin.properties 파일을 엽니다.
 <WebSphere 설치 디렉터리>/profiles/<응용 프로그램 서버 프로파일 이름>/properties
2. com.ibm.ws.scripting.echoparams Java 속성을 false로 설정합니다.

WebSphere 환경에서 보안 프로파일 생성

WebSphere에서 Multidomain MDM 및 Informatica ActiveVOS에 사용할 보안 프로파일을 구성합니다.

1. 명령줄에서 다음 샘플 코드에 표시된 대로 보안 프로파일을 생성합니다.

Windows의 경우

```
<app_server_root>\bin\manageprofiles.bat -create -profileName AppSrv01  
-profilePath <app_server_root>\profiles\AppSrv01  
-templatePath <app_server_root>\profileTemplates\default  
-adminUserName administrator -adminPassword password1 -enableAdminSecurity true
```

UNIX의 경우

```
<app_server_root>/bin/manageprofiles.sh -create -profileName AppSrv01  
-profilePath <app_server_root>/profiles/AppSrv01  
-templatePath <app_server_root>/profileTemplates/default  
-adminUserName administrator -adminPassword password1 -enableAdminSecurity true
```

2. WebSphere 콘솔에서 보안 전송 유형을 SSL 지원됨 유형으로 변경합니다.
 - a. 보안을 확장하고 **글로벌 보안**을 클릭합니다.
 - b. 인증 아래에서 **RMI/IIOP 보안**을 확장하고 **CSiv2 인바운드 통신**을 클릭합니다.
 - c. CSiv2 전송 계층 아래의 전송 목록에서 **SSL 지원됨**을 선택합니다.
 - d. 적용을 클릭한 다음 **저장**을 클릭합니다.
 - e. **CSiv2 아웃바운드 통신**을 클릭합니다.
 - f. CSiv2 전송 계층 아래의 전송 목록에서 **SSL 지원됨**을 선택합니다.
 - g. 적용을 클릭한 다음 **저장**을 클릭합니다.
3. WebSphere 콘솔에서 응용 프로그램 보안이 설정되었는지 확인합니다.
 - a. 보안을 확장하고 **글로벌 보안**을 클릭합니다.
 - b. 응용 프로그램 보안 아래에서 **응용 프로그램 보안 활성화**를 선택합니다.
 - c. 적용을 클릭한 다음 **저장**을 클릭합니다.
4. 페더레이션된 리포지토리를 설정합니다.
 - a. 보안을 확장하고 **글로벌 보안**을 클릭합니다.
 - b. 사용자 계정 리포지토리 아래의 사용 가능한 영역 정의 목록에서 **페더레이션된 리포지토리**를 선택합니다.
 - c. 구성을 클릭합니다.
 - d. 영역의 리포지토리 아래에서 **기본 제공 리포지토리 사용**을 클릭합니다.
 - e. 관리자에 대한 암호를 지정합니다.
 - f. 적용을 클릭한 다음 **저장**을 클릭합니다.
5. WebSphere 프로파일을 다시 시작합니다.

ActiveVOS 콘솔 관리자 생성

ActiveVOS를 사용하려는 경우 응용 프로그램 서버 컨테이너에서 abAdmin 역할이 있는 ActiveVOS 콘솔 관리자를 생성합니다. ActiveVOS를 사용하려는 경우 abAdmin 역할이 있는 ActiveVOS 콘솔 관리자를 생성합니다. 관리자를 생성하지 않으면 Hub 서버 배포가 실패합니다. Hub 서버 설치 프로그램에서 ActiveVOS 콘솔의 관리자 자격 증명을 입력하라는 메시지를 표시하면 ActiveVOS 콘솔 관리자 이름과 암호를 사용합니다.

1. 다음 디렉터리로 변경합니다.
<JBoss 설치 디렉터리>/bin
2. add-user 유틸리티를 실행하려면 다음 스크립트를 사용합니다.
UNIX의 경우. add-user.sh
Windows의 경우. add-user.bat
3. 나타나는 프롬프트에 응답합니다.
다음 테이블에는 각 프롬프트에 대해 지정할 값이 설명되어 있습니다.

프롬프트	지정할 값
어떤 사용자 유형을 추가하시겠습니까? a) 관리 사용자 또는 b) 응용 프로그램 사용자	응용 프로그램 사용자를 선택하려면 b를 입력합니다.
영역(ApplicationRealm)	영역 이름. standalone-full.xml 파일에 추가한 login-module에서 지정한 영역 이름을 입력합니다.
사용자 이름	ActiveVOS 콘솔 관리자 이름.
암호	JBoss 암호 표준을 준수하는 암호.
이 사용자에게 어떤 역할을 지정하시겠습니까?	abAdmin.
영역 <영역 이름>에 대해 사용자 <사용자 이름>을 추가하려고 합니다. 맞습니까?	사용자를 추가하려면 yes를 입력합니다.
하나의 AS 프로세스가 다른 AS 프로세스에 연결하는 데 이 새로운 사용자가 사용됩니까?	yes.

4. WebSphere 콘솔에 로그인하고 ActiveVOS 콘솔 관리자를 생성합니다.
참고: ActiveVOS 콘솔 사용자는 설치 후 또는 업데이트 후 프로세스 중 postInstallSetup 또는 patchInstallSetup 스크립트를 실행하면 abAdmin 역할에 매핑됩니다.
5. WebLogic 콘솔에 로그인합니다.
6. abAdmin 역할을 생성합니다.
7. ActiveVOS 콘솔 관리자를 생성합니다.
8. abAdmin 역할에 관리자 할당

MDM Hub 배포를 위한 SOAP 요청 제한 시간 구성

MDM Hub 구성 요소의 배포가 제한 시간을 초과하지 않도록 하려면 SOAP 요청 제한 시간 속성을 설정합니다. 설치가 성공적으로 완료되면 이 속성을 기본값으로 재설정할 수 있습니다.

1. 다음 디렉터리에서 `soap.client.props` 파일을 엽니다.
 <WebSphere 설치 디렉터리>/profiles/<응용 프로그램 서버 프로파일 이름>/properties
2. `com.ibm.SOAP.requestTimeout` 속성을 1800 이상으로 설정합니다.

추가 IBM WebSphere 구성

MDM Hub 환경의 요구 사항을 기반으로 추가 WebSphere 구성을 수행합니다.

다음 테이블에는 수행할 수 있는 구성이 설명되어 있습니다.

구성	설명
독립 실행형 처리 서버 인스턴스에 대한 WebSphere 구성	다음과 같은 시나리오에서 독립 실행형 처리 서버 인스턴스에 대한 WebSphere를 구성하는 데 필요합니다. <ul style="list-style-type: none">- Hub 서버가 설치되지 않은 WebSphere 인스턴스에 처리 서버 인스턴스를 설치하려고 합니다.- 여러 독립 실행형 처리 서버 인스턴스를 설치하려고 합니다.
여러 MDM Hub 마스터 데이터베이스에 대한 WebSphere 구성	여러 MDM Hub 마스터 데이터베이스 인스턴스를 구성하려는 경우에 필요합니다.
HTTPS 프로토콜 구성	MDM Hub 통신을 위한 HTTPS 프로토콜을 구성하려는 경우에 필요합니다.
Informatica Data Director에 대한 WebSphere 구성	Data Director를 사용하려는 경우에 필요합니다.

독립 실행형 처리 서버 인스턴스에 대한 WebSphere 구성

여러 독립 실행형 처리 서버 인스턴스를 설치하려는 경우 적절한 데이터 소스를 사용하도록 WebSphere를 구성합니다. 또한 Hub 서버가 설치되지 않은 WebSphere 인스턴스에 처리 서버 인스턴스를 설치하려는 경우 데이터 소스를 구성합니다.

적절한 데이터 소스를 사용하도록 WebSphere를 구성하려면 다음과 같은 태스크를 수행합니다.

1. JDBC 드라이버를 설치합니다.
2. MDM Hub 마스터 데이터베이스 데이터 소스를 생성합니다.
3. 연산 참조 저장소 데이터 소스를 생성합니다.

1단계. JDBC 드라이버 설치

MDM Hub 마스터 데이터베이스 및 ORS(연산 참조 저장소)에 대한 데이터 소스를 생성하기 전에 JDBC 드라이버를 설치합니다.

지원되는 버전의 JDBC 드라이버를 가져오려면 Oracle에 문의합니다.

지원되는 버전의 JDBC 드라이버를 가져오려면 Microsoft에 문의합니다.

지원되는 버전의 JDBC 드라이버를 가져오려면 IBM에 문의합니다.

- ▶ JDBC 드라이버를 다음 디렉터리에 복사합니다.

<WebSphere installation directory>/lib

2단계. MDM Hub 마스터 데이터베이스 데이터 소스 생성

처리 서버 시스템에 JDBC 드라이버를 설치한 후 MDM Hub 마스터 데이터베이스에 대한 데이터 소스를 생성합니다.

1. WebSphere 응용 프로그램 서버 관리 콘솔을 시작합니다.
2. 드라이버 라이브러리의 위치를 지정합니다.
 - a. 콘솔 탐색 트리에서 **환경**을 확장합니다.
 - b. **WebSphere 변수** 링크를 클릭합니다.
 - c. 다음과 같은 JDBC 드라이버 디렉터리를 가리키도록 JDBC 변수를 업데이트합니다.
<WebSphere 설치 디렉터리>/lib
3. MDM Hub 마스터 데이터베이스 데이터 소스가 사용할 보안 계정을 생성합니다.
 - a. 콘솔 탐색 트리에서 **보안**을 확장합니다.
 - b. **관리, 응용 프로그램 및 인프라 보안** 링크를 클릭합니다.
 - c. **인증**에서 **Java Authentication and Authorization Service**를 확장하고 **J2C 인증 데이터**를 클릭합니다.
 - d. **새로 만들기**를 클릭하고 다음과 같은 속성을 지정합니다.

속성	설명
별칭	MDM Hub 마스터 데이터베이스의 이름입니다.
사용자 ID	MDM Hub 마스터 데이터베이스에 연결하는 데 필요한 사용자 이름입니다.
암호	MDM Hub 마스터 데이터베이스에 액세스하는 데 필요한 암호입니다.

- e. **확인**을 클릭합니다.
4. JDBC 공급자를 생성합니다.
 - a. **리소스 > JDBC**를 확장한 다음 **JDBC 공급자**를 클릭합니다.
JDBC 공급자 페이지가 표시됩니다.
 - b. 응용 프로그램이 데이터 소스를 사용할 범위를 선택합니다.
 - c. **새로 만들기**를 클릭하고 다음과 같은 속성을 지정합니다.

속성	설명
데이터베이스 유형	데이터베이스의 유형입니다. Oracle을 선택합니다.
공급자 유형	JDBC 공급자의 유형입니다. Oracle JDBC 드라이버 를 선택합니다.

속성	설명
구현 유형	데이터 소스 구현 유형입니다. XA 데이터 소스 를 선택합니다.
이름	JDBC 공급자의 이름입니다. 이름을 Informatica MDM Oracle JDBC Provider (XA)로 변경합니다.

d. 다음을 클릭한 다음 **마침**을 클릭합니다.

5. MDM Hub 마스터 데이터베이스 데이터 소스를 생성합니다.

- a. 생성한 JDBC 공급자를 클릭합니다.
구성 페이지가 표시됩니다.
- b. **추가 속성** 아래에서 **데이터 소스**를 클릭합니다.
데이터 소스 페이지가 표시됩니다.
- c. **새로 만들기**를 클릭합니다.
- d. 다음 데이터 소스 속성을 지정합니다.

속성	설명
이름	데이터 소스 이름입니다. MDM Master Data Source를 지정합니다.
JNDI 이름	JDBC 데이터 소스를 바인딩할 JNDI 경로입니다. jdbc/siperian-cmx_system-ds를 지정합니다. 참고: JNDI 이름은 소문자로 지정해야 합니다.
구성 요소 관리 인증 별칭	마스터 데이터베이스 데이터 소스의 인증 별칭입니다. <호스트 이름>/cmx_system을 선택합니다.

e. 다음을 클릭한 다음 **마침**을 클릭합니다.

3단계. 연산 참조 저장소 데이터 소스 생성

처리 서버 시스템에 JDBC 드라이버를 설치한 후 각 연산 참조 저장소에 대한 데이터 소스를 생성합니다.

1. WebSphere 응용 프로그램 서버 관리 콘솔을 시작합니다.
2. 드라이버 라이브러리의 위치를 지정합니다.
 - a. 콘솔 탐색 트리에서 **환경**을 확장합니다.
 - b. **WebSphere 변수** 링크를 클릭합니다.
 - c. 다음과 같은 JDBC 드라이버 디렉토리를 가리키도록 JDBC 변수를 업데이트합니다.
<WebSphere 설치 디렉터리>/lib
3. 연산 참조 저장소가 사용할 보안 계정을 생성합니다.
 - a. 콘솔 탐색 트리에서 **보안**을 확장합니다.
 - b. **관리, 응용 프로그램 및 인프라 보안** 링크를 클릭합니다.
 - c. 인증에서 **Java Authentication and Authorization Service**를 확장하고 **J2C 인증 데이터**를 클릭합니다.

- d. **새로 만들기**를 클릭하고 다음과 같은 속성을 설정합니다.

속성	설명
별칭	연산 참조 저장소의 이름입니다.
사용자 ID	연산 참조 저장소에 연결하는 데 필요한 사용자 이름입니다.
암호	연산 참조 저장소에 액세스하는 데 필요한 암호입니다.

- e. **확인**을 클릭합니다.

4. JDBC 공급자를 생성합니다.

- a. **리소스 > JDBC**를 확장한 다음 **JDBC 공급자**를 클릭합니다.
JDBC 공급자 페이지가 표시됩니다.
- b. 응용 프로그램이 데이터 소스를 사용할 범위를 선택합니다.
- c. **새로 만들기**를 클릭하고 다음과 같은 속성을 지정합니다.

속성	설명
데이터베이스 유형	데이터베이스의 유형입니다. Oracle을 선택합니다.
공급자 유형	JDBC 공급자의 유형입니다. Oracle JDBC 드라이버 를 선택합니다.
구현 유형	데이터 소스 구현 유형입니다. XA 데이터 소스 를 선택합니다.
이름	JDBC 공급자의 이름입니다. 이름을 Informatica MDM Oracle JDBC Provider (XA)로 변경합니다.

- d. **다음**을 클릭한 다음 **마침**을 클릭합니다.

5. 연산 참조 저장소 데이터 소스를 생성합니다.

- a. 생성한 JDBC 공급자를 클릭합니다.
구성 페이지가 표시됩니다.
- b. **추가 속성** 아래에서 **데이터 소스**를 클릭합니다.
데이터 소스 페이지가 표시됩니다.
- c. **새로 만들기**를 클릭합니다.

d. 다음 데이터 소스 속성을 지정합니다.

속성	설명
이름	데이터 소스 이름입니다. MDM ORS Data Source를 지정합니다.
JNDI 이름	JDBC 데이터 소스를 바인딩할 JNDI 경로입니다. jdbc/siperian-<Oracle 호스트 이름>-<Oracle SID>-<연산 참조 저장소 이름>-dsjdbc/siperian-<IBM Db2 호스트 이름>-<IBM Db2 데이터베이스 이름>-<연산 참조 저장소 이름>-dsjdbc/siperian-<Microsoft SQL Server 호스트 이름>-<연산 참조 저장소 이름>-ds를 지정합니다. 참고: JNDI 이름은 소문자로 지정해야 합니다.
구성 요소 관리 인증 별칭	마스터 데이터베이스 데이터 소스의 인증 별칭입니다. <호스트 이름>/<연산 참조 저장소 이름>을 선택합니다.
드라이버 클래스 이름	JDBC 드라이버 클래스입니다. com.microsoft.sqlserver.jdbc.SQLServerXADataSource로 설정합니다.

e. 다음을 클릭한 다음 **마침**을 클릭합니다.

여러 MDM Hub 마스터 데이터베이스에 대한 WebSphere 구성

여러 MDM Hub 마스터 데이터베이스 인스턴스를 구성하려는 경우 MDM Hub 마스터 데이터베이스 인스턴스 수 만큼 많은 WebSphere 프로필을 구성합니다. 각 MDM Hub 마스터 데이터베이스 인스턴스에는 고유한 MDM Hub 인스턴스가 있어야 합니다. 따라서 별도의 WebSphere 프로필에 각 MDM Hub 인스턴스를 배포할 만큼 많은 WebSphere 프로필을 생성합니다.

HTTPS 프로토콜 구성

Hub 서버, 처리 서버, ActiveVOS와 같은 MDM Hub 구성 요소 간 통신에 대해 HTTPS 프로토콜을 사용하려면 WebSphere 응용 프로그램 서버에서 HTTPS 프로토콜을 구성합니다.

1. SSL 활성화 WebSphere 포트를 생성합니다.
2. 자체 서명된 인증서를 허용하도록 WebSphere를 구성합니다.
3. 다음과 같은 사용자 지정 JVM 속성을 구성합니다.

사용자 지정 JVM 속성	설명
javax.net.ssl.keyStore	키 저장소의 위치입니다.
javax.net.ssl.keyStorePassword	키 저장소의 암호입니다.
javax.net.ssl.keyStoreType	키 저장소의 유형입니다.
javax.net.ssl.trustStore	트러스트 저장소의 위치입니다.
javax.net.ssl.trustStorePassword	트러스트 저장소의 암호입니다.
javax.net.ssl.trustStoreType	트러스트 저장소의 유형입니다.

HTTPS 프로토콜 구성에 대한 자세한 내용은 WebSphere 설명서를 참조하십시오.

Informatica Data Director에 대한 WebSphere 구성

Data Director를 사용하려는 경우, WebSphere를 구성한 다음 변경 내용이 적용될 수 있도록 WebSphere를 다시 시작합니다.

다음 구성을 수행해야 합니다.

- 웹 컨테이너 사용자 지정 속성을 설정합니다.
WebSphere 콘솔을 사용하여 `com.ibm.ws.webcontainer.invokerequestlistenerforfilter`를 `true`로 설정합니다.
웹 컨테이너 사용자 지정 속성 설정에 대한 지침은 WebSphere 설명서를 참조하십시오.
- 태스크에 대한 관리 기능을 지원하려면 제한 시간 속성의 값을 2배로 늘립니다.
WebSphere 콘솔을 사용하여 다음과 같은 태스크를 수행합니다.
 1. **WebSphere 콘솔 서버 > 서버 유형 > WebSphere 응용 프로그램 서버 > <대상 서버 이름>**으로 이동합니다.
 2. **컨테이너 서비스** 범주에서 **트랜잭션 서비스**를 클릭하고 제한 시간 속성의 값을 늘립니다.

제 5 장

Hub 저장소 업그레이드

이 장에 포함된 항목:

- [Hub 저장소 업그레이드 개요, 73](#)
- [Hub 저장소 복제\(클린 업그레이드\), 73](#)
- [영어가 아닌 로캘로 설정된 데이터베이스, 74](#)
- [자세한 정보 표시 모드에서 MDM Hub 마스터 데이터베이스 업그레이드, 74](#)
- [자동 모드에서 MDM Hub 마스터 데이터베이스 업그레이드, 75](#)
- [자세한 정보 표시 모드에서 연산 참조 저장소 데이터베이스 업그레이드, 76](#)
- [자동 모드에서 연산 참조 저장소 데이터베이스 업그레이드, 79](#)
- [업그레이드 스크립트가 성공적으로 실행되었는지 확인, 80](#)

Hub 저장소 업그레이드 개요

Hub 저장소는 MDM Hub 마스터 데이터베이스와 하나 이상의 **ORS**(연산 참조 저장소) 데이터베이스가 포함된 데이터베이스입니다. 배포 시 제공된 스크립트를 사용하여 데이터베이스를 업그레이드합니다.

데이터베이스 환경이 영어가 아닌 로캘로 설정된 경우 Hub 저장소를 업그레이드하기 전에 문자 집합이 유니코드 문자 집합인지 확인합니다. 업그레이드를 성공적으로 완료하고 나면 기본 설정 로캘을 선택할 수 있습니다. 이 로캘은 데이터베이스 수준이 아닌, 사용자 계정 기본 설정으로 저장됩니다.

참고: 데이터베이스 디렉터리 또는 폴더 이름에 대한 경로에 공백이 포함되지 않아야 합니다. 디렉터리 또는 폴더 이름에 공백이 있는 경로를 지정하면 업그레이드가 실패합니다.

Hub 저장소 복제(클린 업그레이드)

클린 업그레이드를 수행하는 경우 DBA에게 마스터 데이터베이스 및 연산 참조 저장소의 백업 및 복제를 요청하십시오. 복제된 데이터베이스를 새 환경에 복사합니다. 새 환경에서 마스터 데이터베이스 및 연산 참조 저장소를 업그레이드합니다.

영어가 아닌 로캘로 설정된 데이터베이스

Hub 저장소 데이터베이스 환경이 영어 외 언어 로캘로 설정된 경우 업그레이드 스크립트를 실행하기 전에 데이터베이스 환경에서 유니코드 문자 집합을 사용하는지 확인하십시오. 데이터베이스 환경 변수를 사용하여 문자 집합을 설정합니다.

참고: Microsoft SQL Server 환경에서는 이 작업이 필요하지 않습니다.

업그레이드 스크립트에서 메타데이터를 영어로 변환하고 변환 키를 메타데이터와 연결합니다. 업그레이드가 성공하면 각 MDM Hub 콘솔 사용자는 사용자 인터페이스 및 데이터베이스가 지원하는 로캘을 모두 선택할 수 있습니다. 각 사용자의 로캘 선택 사항은 모든 사용자 데이터가 있는 마스터 데이터베이스에 저장됩니다.

예를 들어 한국어 로캘로 된 Oracle 데이터베이스 환경에 있는 MDM Hub 저장소를 생각해 볼 수 있습니다. 업그레이드하기 전에 데이터베이스 환경 변수인 NLS_LANG가 KOREAN_KOREA.AL32UTF8(한국어 유니코드)로 설정되어 있는지 확인합니다. 업그레이드를 하면 사용자는 로캘을 한국어로 설정할 수 있고 다른 사용자는 지원되는 또 다른 로캘을 선택할 수 있습니다.

다음 테이블에는 문자 집합을 설정하는 데 사용할 수 있는 데이터베이스 환경 변수가 나열되어 있습니다.

데이터베이스	환경 변수 이름
Oracle	NLS_LANG
IBM Db2	DB2CODEPAGE

데이터베이스 환경 변수에 대한 자세한 내용은 사용 중인 데이터베이스의 설명서를 참조합니다.

자세한 정보 표시 모드에서 MDM Hub 마스터 데이터베이스 업그레이드

MDM Hub 마스터 데이터베이스를 업그레이드하려면 업그레이드 스크립트를 실행합니다.

참고: 메타데이터를 변경하는 데 Hub 콘솔을 사용하지 않으면 데이터베이스 업그레이드 스크립트 오류가 발생할 수도 있습니다. 스크립트 오류가 발생하면 Informatica 글로벌 고객 지원 센터에 문의하십시오.

1. 명령 프롬프트를 엽니다.
2. 다음 디렉터리로 이동합니다.
 - UNIX: <MDM Hub 배포 디렉터리>/database/bin
 - Windows의 경우. <MDM Hub 배포 디렉터리>\database\bin
3. 다음 명령을 사용하여 MDM Hub 마스터 데이터베이스 업그레이드 스크립트를 실행합니다.
 - UNIX: sip_ant.sh updatemasterdatabase
 - Windows의 경우. sip_ant.bat updatemasterdatabase
4. 프롬프트에 응답합니다.
5. 마스터 데이터베이스 업그레이드 중에 프롬프트가 나타나면 다음 프롬프트에 응답합니다.

This upgrade should be performed by a DBA to grant 'create sequence' privileges for the master database. The master database does not have 'create sequence' privileges, you can either grant it now (manually) and then move forward or re-start the upgrade, or direct this process to do so for you

now, and continue the current upgrade.
Do you want the process to create this privilege? Yes/No

- **No**를 입력하면 업그레이드 프로세스가 사용자에게 권한이 부여되었는지 다시 확인한 다음 **TNS** 이름 프롬프트로 돌아옵니다.
- **Yes**를 입력하면 업그레이드 프로세스를 계속 진행하기 전에 다음 프롬프트에 응답해야 합니다.

DBA 사용자 이름 입력:

DBA 암호 입력:

6. JBoss 환경에서 응용 프로그램 서버를 다시 시작합니다.
7. **CMX_SYSTEM** 업그레이드 로그 파일의 사본을 업그레이드 설머서 디렉터리에 저장합니다. 변경 스크립트 마다 하나의 로그 파일이 있습니다.

업그레이드 프로세스가 파일을 다음 위치에 저장합니다.

- UNIX의 경우.

<MDM Hub 설치 디렉터리>/server/resources/database/db_changes/<데이터베이스 이름>/Master

- Windows의 경우.

<MDM Hub 설치 디렉터리>\server\resources\database\db_changes\<데이터베이스 이름>\Master

자동 모드에서 MDM Hub 마스터 데이터베이스 업그레이드

자동 모드에서 **MDM Hub** 마스터 데이터베이스를 업그레이드하려면 환경에 적합한 명령을 사용하여 업그레이드 스크립트를 실행합니다.

참고: 명령줄에 자동 업그레이드 프로세스를 표시하려면 **-Dnoprompt**를 **true**로 설정합니다. 명령줄에 시작 시간과 종료 시간만 표시하려면 **-Dnoprompt**를 **true -silent**로 설정합니다.

1. 명령 프롬프트를 엽니다.
2. 다음 디렉터리로 이동합니다.
 - UNIX. <MDM Hub 배포 디렉터리>/database/bin
 - Windows. <MDM Hub 배포 디렉터리>\database\bin
3. Oracle 환경에서 **MDM Hub** 마스터 데이터베이스를 업그레이드하려면 다음 명령을 실행합니다.

UNIX.

```
sip_ant.sh updatemasterdatabase -Dmaster.tnsname=<TNS name> -Dmaster.connectiontype=<SID or SERVICE> -Dmaster.server=<host name> -Dmaster.port=<port> -Dmaster.sid=<SID name> -Dmaster.username=<MDM Hub Master Database username> -Dmaster.password=<MDM Hub Master Database password> -Dcmx.username=<administrator username> -Dcmx.password=<administrator password> -Dcmx.server.masterdatabase.type=ORACLE -Dnoprompt=true
```

Windows.

```
sip_ant.bat updatemasterdatabase -Dmaster.tnsname=<TNS name> -Dmaster.connectiontype=<SID or SERVICE> -Dmaster.server=<host name> -Dmaster.port=<port> -Dmaster.sid=<SID name> -Dmaster.username=<MDM Hub Master Database username> -Dmaster.password=<MDM Hub Master Database password> -Dcmx.username=<administrator username> -Dcmx.password=<administrator password> -Dcmx.server.masterdatabase.type=ORACLE -Dnoprompt=true
```

4. IBM Db2 환경에서 **MDM Hub** 마스터 데이터베이스를 업그레이드하려면 다음 명령을 실행합니다.

UNIX.

```
sip_ant.sh updatemasterdatabase -Dcmx.server.masterdatabase.type=DB2 -Dmaster.hostname=<host name>  
-Dmaster.port=<port> -Dmaster.username=<MDM Hub Master Database username> -Dmaster.password=<MDM Hub  
Master Database password> -Ddba.username=<DBA username>  
-Ddba.password=<DBA password> -Dnoprompt=true
```

Windows.

```
sip_ant.bat updatemasterdatabase -Dcmx.server.masterdatabase.type=DB2 -Dmaster.hostname=<host name>  
-Dmaster.port=<port> -Dmaster.username=<MDM Hub Master Database username> -Dmaster.password=<MDM Hub  
Master Database password> -Ddba.username=<DBA username>  
-Ddba.password=<DBA password> -Dnoprompt=true
```

5. Microsoft SQL Server 환경에서 MDM Hub 마스터 데이터베이스를 업그레이드하려면 다음 명령을 실행합니다.

UNIX.

```
sip_ant.sh updatemasterdatabase -Dmaster.hostname=<hostname> -Dmaster.port=<port> -  
Dmaster.username=<MDM Hub Master Database username>  
-Dmaster.password=<MDM Hub Master Database password>  
-Dcmx.username=<administrator username> -Dcmx.password=<administrator password> -  
Dcmx.server.masterdatabase.type=MSSQL  
-Dmaster.database=<MDM Hub Master Database name> -Dnoprompt=true
```

Windows.

```
sip_ant.bat updatemasterdatabase -Dmaster.hostname=<hostname> -Dmaster.port=<port> -  
Dmaster.username=<MDM Hub Master Database username>  
-Dmaster.password=<MDM Hub Master Database password>  
-Dcmx.username=<administrator username> -Dcmx.password=<administrator password> -  
Dcmx.server.masterdatabase.type=MSSQL  
-Dmaster.database=<MDM Hub Master Database name> -Dnoprompt=true
```

자세한 정보 표시 모드에서 연산 참조 저장소 데이터베이스 업그레이드

각 ORS(연산 참조 저장소) 데이터베이스를 업그레이드하려면 업그레이드 스크립트를 실행합니다. ORS 데이터베이스를 업그레이드하기 전에 MDM Hub 마스터 데이터베이스를 업그레이드합니다.

참고: 메타데이터를 변경하는 데 Hub 콘솔을 사용하지 않으면 데이터베이스 업그레이드 스크립트 오류가 발생할 수도 있습니다. 스크립트 오류가 발생하면 Informatica 글로벌 고객 지원 센터에 문의하십시오.

1. 응용 프로그램 서버를 중지합니다.
2. 명령 프롬프트를 엽니다.
3. 다음 디렉터리로 이동합니다.
 - UNIX의 경우. <MDM Hub 배포 디렉터리>/database/bin
 - Windows의 경우. <MDM Hub 배포 디렉터리>\database\bin
4. 다음 명령을 사용하여 연산 참조 저장소 업그레이드 스크립트를 실행합니다.
 - UNIX. ./sip_ant.sh updateorsdatabase
 - Windows. sip_ant.bat updateorsdatabase
5. 프롬프트에 응답합니다.

Oracle 환경의 경우 다음 정보를 제공합니다.

프롬프트	설명
데이터베이스 유형(ORACLE, MSSQL, DB2)을 입력하십시오.	데이터베이스 유형입니다. Oracle을 지정합니다.
Oracle 연결 유형(서비스, SID)을 입력하십시오. [service]	연결 유형입니다. 다음 값 중 하나를 사용합니다. SERVICE 서비스 이름을 사용하여 Oracle에 연결합니다. SID Oracle 시스템 ID를 사용하여 Oracle에 연결합니다. 기본값은 SERVICE입니다.
연산 참조 저장소 데이터베이스 호스트 이름 입력 [localhost]	데이터베이스를 실행 중인 호스트의 이름입니다.
연산 참조 저장소 데이터베이스 포트 번호를 입력합니다. [1521]	데이터베이스 수신기에서 사용하는 포트 번호입니다. 기본값은 1521입니다.
연산 참조 저장소 데이터베이스 서비스 이름 입력 [orcl]	Oracle 서비스의 이름입니다. 선택한 Oracle 연결 유형이 SERVICE이면 이 프롬프트가 표시됩니다.
Oracle Net 연결 식별자(TNS 이름) 입력 [orcl]	Oracle TNS 이름입니다. 기본값은 orcl입니다.
ORS DB 연결 URL: "jdbc:oracle:thin:@//<host_name>:<port>/<service_name>". 연결 URL 변경 여부 (y/n) [n]	Oracle 연결 유형 SERVICE의 연결 URL입니다. y를 입력하면 기본 연결 URL을 변경할 수 있습니다. 기본 연결 URL을 사용하려면 n을 입력합니다.
데이터베이스 SID 입력 [orcl]	Oracle 시스템 ID의 이름입니다. 이 프롬프트는 선택한 Oracle 연결 유형이 SID이면 표시됩니다.
연산 참조 저장소 데이터베이스 이름 입력 [cmx_ors]	연산 참조 저장소 데이터베이스의 이름입니다. 기본값은 cmx_ors입니다.
연산 참조 저장소 데이터베이스 사용자 암호 입력	연산 참조 저장소에 액세스하는 데 필요한 암호입니다.
목록(de, en_US, fr, ja, ko, zh_CN)에서 로캘 이름 입력 [en_US]	운영 체제 로캘입니다.
DBA 사용자 이름 입력 [sys]	DBA 수준 사용 권한이 있는 사용자의 이름입니다.
DBA 암호 입력	DBA 수준 사용 권한이 있는 사용자의 암호입니다.
ORS 시간 표시 막대 세분화 정수 코드[3](예: 5년, 4월, 3일, 2시, 1분, 0초)를 입력합니다.	사용할 시간 표시 막대 단위를 지정합니다.

IBM Db2 환경의 경우 다음 정보를 제공합니다.

프롬프트	설명
데이터베이스 유형(ORACLE, MSSQL, DB2)을 입력하십시오.	데이터베이스 유형입니다. DB2를 지정합니다.
연산 참조 저장소 데이터베이스 호스트 이름 입력 [localhost]	데이터베이스를 실행 중인 호스트의 이름입니다.
연산 참조 저장소 데이터베이스 포트 번호[50000]를 입력합니다.	데이터베이스 수신기에서 사용하는 포트 번호입니다. 기본값은 50000입니다.
연산 참조 저장소 데이터베이스 이름[SIP97]을 입력합니다.	데이터베이스 이름입니다. 기본값은 SIP97입니다.
연산 참조 저장소 데이터베이스 이름 입력 [cmx_ors]	연산 참조 저장소 데이터베이스의 이름입니다. 기본값은 cmx_ors입니다.
연산 참조 저장소 데이터베이스 사용자 암호 입력	연산 참조 저장소에 액세스하는 데 필요한 암호입니다.
목록(de, en_US, fr, ja, ko, zh_CN)에서 로캘 이름을 입력합니다. [en_US]	운영 체제 로캘입니다. 기본값은 en_US입니다.
DBA 사용자 이름 입력 [sys]	DBA 수준 사용 권한이 있는 사용자의 이름입니다.
DBA 암호 입력	DBA 수준 사용 권한이 있는 사용자의 암호입니다.
ORS 시간 표시 막대 세분화 정수 코드[3](예: 5년, 4월, 3일, 2시, 1분, 0초)를 입력합니다.	사용할 시간 표시 막대 단위를 지정합니다.

Microsoft SQL Server 환경의 경우 다음 정보를 제공합니다.

프롬프트	설명
데이터베이스 유형(ORACLE, MSSQL, DB2)을 입력하십시오.	데이터베이스 유형입니다. MSSQL을 지정합니다.
연산 참조 저장소 데이터베이스 호스트 이름 입력 [localhost]	데이터베이스를 실행 중인 호스트의 이름입니다.
연산 참조 저장소 데이터베이스 포트 번호[1433]를 입력합니다.	데이터베이스 수신기에서 사용하는 포트 번호입니다. 기본값은 1433입니다.
연산 참조 저장소 데이터베이스 이름 입력 [cmx_ors]	연산 참조 저장소 데이터베이스의 이름입니다. 기본값은 cmx_ors입니다.
연산 참조 저장소 데이터베이스 사용자 암호 입력	연산 참조 저장소에 액세스하는 데 필요한 암호입니다.
목록(de, en_US, fr, ja, ko, zh_CN)에서 로캘 이름을 입력합니다. [en_US]	운영 체제 로캘입니다. 기본값은 en_US입니다.

프롬프트	설명
DBA 사용자 이름 입력 [sys]	DBA 수준 사용 권한이 있는 사용자의 이름입니다.
DBA 암호 입력	DBA 수준 사용 권한이 있는 사용자의 암호입니다.
ORS 시간 표시 막대 세분화 정수 코드[3](예: 5년, 4월, 3일, 2시, 1분, 0초)를 입력합니다.	사용할 시간 표시 막대 단위를 지정합니다.

6. CMX_ORS 업그레이드 로그 파일의 사본을 업그레이드 설명서 디렉터리에 저장합니다. 변경 스크립트마다 하나의 로그 파일이 있습니다.

업그레이드 프로세스가 파일을 다음 위치에 저장합니다.

- UNIX의 경우.

<MDM Hub 설치 디렉터리>/server/resources/database/db_changes/<데이터베이스 이름>/ORS

- Windows의 경우.

<MDM Hub 설치 디렉터리>\server\resources\database\db_changes\<데이터베이스 이름>\ORS

중요: 명령줄에서 sip_ant를 실행할 때마다 sip_ant 로그 파일이 덮어씁니다. 다른 ORS를 업그레이드하기 위해 sip_ant 스크립트를 실행하기 전에 백업 사본을 저장해야 합니다.

관련 항목:

- [“MDM Hub 환경 보고서 저장” 페이지 128](#)

자동 모드에서 연산 참조 저장소 데이터베이스 업그레이드

자동 모드에서 연산 참조 저장소 데이터베이스를 업그레이드하려면 환경에 적합한 명령을 사용하여 업그레이드 스크립트를 실행합니다.

참고: 명령줄에 자동 업그레이드 프로세스를 표시하려면 -Dnoprompt를 true로 설정합니다. 명령줄에 자동 업그레이드 프로세스의 시작 시간과 종료 시간만 표시하려면 -Dnoprompt를 true -silent로 설정합니다.

1. 명령 프롬프트를 엽니다.

2. 다음 디렉터리로 이동합니다.

- UNIX. <MDM Hub 배포 디렉터리>/database/bin
- Windows. <MDM Hub 배포 디렉터리>\database\bin

3. Oracle 환경에서 연산 참조 저장소를 업그레이드하려면 다음 명령을 실행합니다.

UNIX.

```
sip_ant.sh updateorsdatabase -Dors.tnsname=<tns name> -Dors.connectiontype=sid -
Dors.hostname=<hostname> -Dors.port=<port> -Dors.sid=<Oracle SID> -Dors.username=<ORS username> -
Dors.password=<ORS password> -Dcmx.server.masterdatabase.type=oracle -Dnoprompt=true
```

Windows.

```
sip_ant.bat updateorsdatabase -Dors.tnsname=<tns name> -Dors.connectiontype=sid -
Dors.hostname=<hostname> -Dors.port=<port> -Dors.sid=<Oracle SID> -Dors.username=<ORS username> -
Dors.password=<ORS password> -Dcmx.server.masterdatabase.type=oracle -Dnoprompt=true
```

4. IBM Db2 환경에서 연산 참조 저장소를 업그레이드하려면 다음 명령을 실행합니다.

UNIX.

```
sip_ant.sh updateorsdatabase -Dors.hostname=<host name> -Dors.database=%db2database% -  
Dors.port=<port>  
-Dors.username=<ORS username> -Dors.password=<ORS password> Dcmx.server.masterdatabase.type=db2 -  
Dnoprompt=true
```

Windows.

```
sip_ant.bat updateorsdatabase -Dors.hostname=<host name> -Dors.database=%db2database% -  
Dors.port=<port>  
-Dors.username=<ORS username> -Dors.password=<ORS password> Dcmx.server.masterdatabase.type=db2 -  
Dnoprompt=true
```

5. Microsoft SQL Server 환경에서 연산 참조 저장소를 업그레이드하려면 다음 명령을 실행합니다.

UNIX.

```
sip_ant.sh updateorsdatabase -Dors.hostname=<hostname> -Dors.port=<port> -Dors.username=<ORS  
username>  
-Dors.password=<ORS password> -Dors.database=<database name> -Dcmx.server.masterdatabase.type=MSSQL  
-Dnoprompt=true
```

Windows.

```
sip_ant.bat updateorsdatabase -Dors.hostname=<hostname> -Dors.port=<port> -Dors.username=<ORS  
username>  
-Dors.password=<ORS password> -Dors.database=<database name> -Dcmx.server.masterdatabase.type=MSSQL  
-Dnoprompt=true
```

업그레이드 스크립트가 성공적으로 실행되었는지 확인

C_REPOS_DB_CHANGE 테이블을 검사하여 Hub 저장소 업그레이드 스크립트가 성공적으로 실행되었는지 확인합니다.

스크립트가 업그레이드 프로세스 중에 실행됩니다(이전 업그레이드 중에 실행되지 않은 스크립트의 경우).

C_REPOS_DB_CHANGE 테이블이 스크립트의 오류 발생을 표시하는 경우 Informatica 글로벌 고객 지원 센터에 문의하십시오.

제 6 장

Hub 서버 업그레이드(인플레이스 업그레이드)

이 장에 포함된 항목:

- [Hub 서버 업그레이드 개요, 81](#)
- [그래픽 모드에서 Hub 서버 업그레이드, 82](#)
- [콘솔 모드에서 Hub 서버 업그레이드, 85](#)
- [자동 모드에서 Hub 서버 업그레이드, 87](#)
- [patchInstallSetup 스크립트 실행, 89](#)
- [업그레이드 기록 폴더에 Hub 서버 로그 파일 복사, 90](#)
- [Hub 서버 업그레이드 재적용\(선택 사항\), 91](#)

Hub 서버 업그레이드 개요

Hub 서버는 MDM의 핵심 및 일반 서비스(액세스, 보안 및 세션 관리 등)를 실행합니다. Hub 서버는 응용 프로그램 서버 환경에 배포됩니다.

참고: 이 장은 인플레이스 업그레이드에만 적용됩니다. 클린 업그레이드의 경우 응용 프로그램 서버 및 데이터베이스 환경에 대한 *Multidomain MDM 설치 가이드*의 지침에 따라 Hub 서버를 설치합니다.

그래픽 모드, 콘솔 모드 또는 자동 모드에서 Hub 서버를 업그레이드할 수 있습니다. 그래픽 모드 또는 콘솔 모드에서 Hub 서버를 업그레이드하려면 배포에서 제공된 Hub 서버 설치 프로그램을 실행합니다. 자동 모드에서 Hub 서버를 업그레이드하려면 자동 설치 프로그램 속성 파일을 구성합니다.

지원되지 않는 ActiveVOS 서버 버전을 사용하는 경우에는 Hub 서버 업그레이드 프로세스 중에 ActiveVOS 서버를 설치하라는 메시지가 표시됩니다.

기존 MDM Hub 설치를 대상으로 가리킬 경우 Hub 서버 설치 프로그램이 설치하는 동안 패치 설치를 전체 설치와 구분합니다. MDM Hub 설치 프로그램은 기존 Hub 서버 설치를 덮어쓰기 전에 중요 파일의 백업을 생성합니다.

그래픽 모드에서 Hub 서버 업그레이드

그래픽 모드에서 Hub 서버를 업그레이드하려면 Hub 서버 설치 프로그램을 실행합니다.

팁: 임베디드 ActiveVOS를 사용하는 경우 업그레이드 프로세스에서 업데이트된 버전의 ActiveVOS를 새 디렉터리에 설치해야 합니다. 기존 ActiveVOS 설치를 덮어쓰려는 경우 업그레이드를 시작하기 전에 `../hub/server/bin/build.properties` 파일을 열고 다음 명령을 주석 처리합니다 `activevos.install.dir = <AVOS 설치 경로>`. 기타 업그레이드 전 태스크는 [“BPM 업그레이드 준비” 페이지 23](#) 항목을 참조하십시오.

1. Hub 서버를 설치할 때 사용한 사용자 이름으로 로그인합니다.
2. Hub 서버가 배포된 응용 프로그램 서버를 시작합니다.
3. 명령 프롬프트를 열고 배포 디렉터리의 Hub 서버 설치 프로그램으로 이동합니다. 기본적으로 설치 프로그램은 다음 디렉터리에 있습니다.
 - UNIX의 경우. `<MDM Hub 배포 디렉터리>/<운영 체제 이름>/mrmsserver`
 - Windows의 경우. `<MDM Hub 배포 디렉터리>\windows\mrmsserver`
4. 다음 명령을 실행합니다.
 - UNIX의 경우. `hub_install.bin`
 - Windows의 경우. `hub_install.exe`
5. 소개 페이지에서 **다음**을 클릭합니다.
사용권 계약 검토 페이지가 표시됩니다.
6. 라이선스 계약의 조건을 검토 및 수락합니다. **다음**을 클릭합니다.
7. Hub 서버 설치 위치를 선택합니다. Hub 서버 설치 폴더에는 `siperian-mrm.ear` 파일이 포함되어 있습니다. **다음**을 클릭합니다.
버전 경고 메시지가 표시됩니다.
8. **확인**을 클릭하여 기존 Hub 서버 설치를 업그레이드합니다.
9. **Hub 콘솔의 서버 세부 정보 구성** 페이지에서 다음 세부 정보를 입력합니다.
 - 공개적으로 액세스할 수 있는 호스트 이름. 응용 프로그램 서버가 바인딩되는 서버의 IP 주소 또는 공개적으로 액세스할 수 있는 호스트 이름(FQDN).
 - HTTP 포트 Hub 콘솔에서 사용해야 하는 서버의 HTTP 포트.
응용 프로그램 서버에 대해 HTTPS가 활성화된 경우 업그레이드 후 `build.properties` 파일의 속성을 편집하여 Hub 콘솔 클라이언트를 구성합니다.
10. 이전 설치에서 응용 프로그램 서버로 WebLogic을 사용하는 경우 Hub 서버 설치 프로그램에 WebLogic 관리 암호를 입력하라는 메시지가 표시됩니다. WebLogic 암호를 입력하십시오.
11. ActiveVOS를 비즈니스 프로세스 관리에 사용하고 있거나 사용하려는 경우 이 버전의 Multidomain MDM에 지원되는 포함된 ActiveVOS 버전을 설치합니다.
중요: 독립 실행형 ActiveVOS는 지원되지 않습니다. 지원되는 버전의 포함된 ActiveVOS가 이미 설치된 경우 이 단계의 일부 하위 단계를 건너뛰어도 됩니다. 하위 단계 **c.**를 수행하여 데이터베이스 세부 정보를 입력하고 하위 단계 **12f.**를 수행하여 ActiveVOS 서버 관리 콘솔에 대한 사용자 이름과 암호를 지정합니다.
 - a. **예**를 선택합니다.
 - b. **Select an ActiveVOS installation directory(ActiveVOS 설치 디렉터리 선택)** 페이지에서 기본 경로를 수락하거나 다른 위치를 선택합니다. **다음**을 클릭합니다.

- c. **Configure the ActiveVOS J2EE data source(ActiveVOS J2EE 데이터 소스 구성)** 페이지에서 ActiveVOS 데이터베이스 스키마 생성 시 지정한 데이터베이스 세부 정보를 입력하고 **다음**을 클릭합니다.

참고: 독립 실행형 ActiveVOS를 포함한 ActiveVOS로 이동하는 경우에는 독립 실행형 ActiveVOS 스키마에 대한 세부 정보를 입력합니다.

- d. **Specify the ActiveVOS web service URL(ActiveVOS 웹 서비스 URL 지정)** 페이지에서 기본 URL을 그대로 사용하거나, ActiveVOS 웹 서비스 호출 시 사용할 URL을 지정합니다. 응용 프로그램 서버에 대한 연결에 사용할 올바른 포트 번호가 URL에 포함되어 있어야 합니다. **다음**을 클릭합니다.

설치 후 설정 스크립트는 URL을 사용하여 ActiveVOS 웹 서비스를 호출하고, 미리 정의된 MDM 워크플로우를 ActiveVOS에 배포하고, URN 매핑을 작성합니다.

- e. **Select the ActiveVOS installer(ActiveVOS 설치 프로그램 선택)** 페이지에서 **선택**을 클릭합니다. 배포 패키지에서 ActiveVOS_Server 설치 파일을 찾습니다. **다음**을 클릭합니다.

- f. 관리자 이름과 암호를 입력하여 ActiveVOS 콘솔의 관리자를 생성합니다.

중요: 사용자 이름과 암호는 응용 프로그램 서버에서 생성한 ActiveVOS 콘솔 사용자 이름 및 암호와 같아야 합니다.

- g. **다음**을 클릭합니다.

12. **MDM 제품 사용 툴킷 구성** 창에서 조직이 속하는 산업 및 환경 유형을 선택합니다.

13. 프록시 서버를 사용하려면 **예**를 선택하고 프록시 서버 세부 정보를 입력합니다. 설치하지 않으려면 **아니요**를 선택합니다.

다음 프록시 서버 세부 정보를 입력할 수 있습니다.

- 프록시 서버 이름/IP
- 프록시 서버 포트
- 프록시 서버 도메인 이름(해당되지 않는 경우 공백 유지)
- 프록시 서버 사용자 이름(해당되지 않는 경우 공백 유지)
- 프록시 서버 암호입니다. 해당되지 않는 경우 비워 둡니다.

14. **다음**을 클릭합니다.

Hub 서버 배포 페이지가 표시됩니다.

15. Hub 서버 EAR 파일을 자동으로 또는 수동으로 배포할지 선택하고 다음을 클릭합니다.

옵션	조건
예, 설치 중에 스크립트를 실행합니다.	<p>설치 완료 후 Hub 서버 EAR 파일을 자동으로 배포합니다.</p> <p>ActiveVOS 시나리오 중 하나에서 다음 응용 프로그램 서버 환경 중 하나를 사용하는 경우 이 옵션을 선택합니다.</p> <p>응용 프로그램 서버 환경:</p> <ul style="list-style-type: none"> • JBoss 독립 실행형 환경 • WebSphere 독립 실행형 환경 <p>ActiveVOS 시나리오:</p> <ul style="list-style-type: none"> • ActiveVOS를 설치하도록 선택했고 이 환경에 다른 버전의 ActiveVOS가 설치되지 않았습니 다. • ActiveVOS를 설치하도록 선택했고 환경에 지원되는 버전의 ActiveVOS가 포함되어 있습니 다. PAM(Product Availability Matrix)에서 지원되는 버전을 확인합니다. • ActiveVOS를 설치하지 않도록 선택했습니다. <p>중요: ActiveVOS를 설치하도록 선택했지만 환경에 지원되지 않는 버전의 ActiveVOS가 있는 경</p> <p>우 아니요, 나중에 실행하겠습니다를 선택합니다.</p>
아니요, 나중에 실행하겠습니다.	<p>Hub 서버 EAR 파일을 수동으로 패키징하고 배포합니다.</p> <p>ActiveVOS를 설치하거나 설치하지 않고 다음 응용 프로그램 서버 환경 중 하나를 사용하는 경</p> <p>우 이 옵션을 선택합니다.</p> <ul style="list-style-type: none"> • WebLogic 독립 실행형 환경 • WebLogic 클러스터 환경 • JBoss 클러스터 환경 • WebSphere 클러스터 환경 <p>지원되지 않는 버전의 ActiveVOS가 포함되는 응용 프로그램 서버 환경을 사용하는 경우 이 옵션</p> <p>을 선택합니다. ActiveVOS 스키마를 지원되는 버전으로 업그레이드한 다음 수동으로 배포해</p> <p>야 합니다.</p>

요약 창이 표시됩니다.

16. 옵션을 변경하려면 **이전** 단추를 클릭하여 이전 선택 항목을 변경합니다.

17. 요약 창에 원하는 옵션이 표시되고 나면 **설치**를 클릭하여 설치 프로세스를 시작합니다.

Hub 서버 설치 프로그램이 시스템을 구성하는 동안 이 프로그램에 **잠시 기다려 주십시오**. 화면이 표시됩니

다. Hub 서버 설치 프로그램은 MDM Hub 설치 디렉터리의 **backup** 폴더에 저장된 보관에 중요 파일을 백업

합니다. 보관의 파일 이름은 다음 예에 나와 있는 형식을 사용합니다.

Informatica MDM Hub Server-2010-09-27_12-13.jar

설치가 완료되면 **설치 완료** 창이 표시됩니다.

18. **완료**를 클릭하여 Hub 서버 설치 프로그램을 종료합니다.

참고: 업그레이드가 성공적으로 완료되지 않은 경우 업그레이드가 실패했음을 나타내는 창이 표시되고 실패

메시지가 포함된 로그 파일의 위치가 표시됩니다.

19. **아니요, 나중에 실행하겠습니다**를 선택한 경우 Hub 서버 EAR 파일을 다시 패키징하고 배포합니다.

- a. 환경에 지원되지 않는 버전의 ActiveVOS가 있는 경우 ActiveVOS 스키마를 지원되는 버전으로 업데이
- 트합니다.
- b. 다음 명령을 실행하여 EAR 파일을 다시 패키징합니다.

UNIX의 경우.

```
cd <MDM Hub 설치 디렉터리>/hub/server/bin
./sip_ant.sh repack
```

Windows의 경우.

```
cd <MDM Hub 설치 디렉터리>\hub\server\bin  
sip_ant.bat repack
```

- c. 응용 프로그램 서버 관리 콘솔에서 Hub 서버 EAR 파일을 수동으로 배포합니다. 응용 프로그램 서버 설명서를 참조하십시오.

20. 응용 프로그램 서버를 다시 시작합니다.

콘솔 모드에서 Hub 서버 업그레이드

UNIX에서 콘솔 모드로 Hub 서버를 업그레이드할 수 있습니다.

팁: 임베디드 ActiveVOS를 비즈니스 프로세스 관리에 사용하는 경우 업그레이드 프로세스에서 업데이트된 버전의 ActiveVOS를 새 디렉터리에 설치해야 합니다. 기존 ActiveVOS 설치를 덮어쓰려는 경우 업그레이드를 시작하기 전에 ../hub/server/bin/build.properties 파일을 열고 다음 명령을 주석 처리합니다 activevos.install.dir = <AVOS 설치 경로>

1. 응용 프로그램 서버를 시작합니다.

2. MDM Hub 배포의 다음 디렉터리로 이동합니다.

UNIX의 경우. <MDM Hub 배포 디렉터리>/<운영 체제 이름>/mrmsserver

3. 명령 프롬프트에서 다음 명령을 실행합니다.

```
./hub_install.bin -i console
```

4. 업그레이드를 위해 선택할 로컬의 번호를 입력하고 **Enter** 키를 누릅니다.

업그레이드에 관한 소개 정보가 표시됩니다.

5. **Enter** 키를 누릅니다.

라이선스 계약이 표시됩니다.

6. 라이선스 계약의 내용을 확인합니다. 라이선스 계약의 조항에 동의하면 **Y**를 입력하고, 라이선스 계약에 동의하지 않고 설치 프로그램을 종료하려면 **N**을 입력합니다.

7. **Enter** 키를 누릅니다.

이전 단계에서 **Y**를 입력한 경우 설치 폴더에 대한 정보가 표시됩니다.

8. Hub 서버를 설치한 디렉터리를 지정합니다.

- 기본 폴더를 선택하려면 **Enter** 키를 누릅니다.

- 경로를 변경하려면 설치 폴더의 절대 경로를 입력하고 **Enter** 키를 누릅니다.

9. 설치 폴더의 위치를 확인합니다. 설치 폴더를 확인하려면 **Y**를 입력하고, 설치 폴더를 변경하려면 **N**을 입력합니다.

버전 경고 메시지가 표시됩니다.

10. **Enter** 키를 눌러 계속할지 확인합니다.

11. Hub 콘솔의 서버 세부 정보를 구성합니다.

- 응용 프로그램 서버가 바인딩되는 서버의 IP 주소 또는 정규화된 호스트 이름(FQDN).

- Hub 콘솔에서 사용해야 하는 서버의 HTTP 포트.

응용 프로그램 서버에 대해 HTTPS가 활성화된 경우 업그레이드 후 build.properties 파일의 속성을 편집하여 Hub 콘솔 클라이언트를 구성합니다.

12. WebLogic 환경에서 WebLogic 암호를 입력하고 **Enter** 키를 누릅니다.

13. 임베디드 ActiveVOS를 비즈니스 프로세스 관리에 사용하고 있거나 사용하려는 경우 이 버전의 Multidomain MDM에 권장되는 임베디드 ActiveVOS 버전을 설치합니다.
중요: 독립 실행형 ActiveVOS는 지원되지 않습니다. 지원되는 버전의 포함된 ActiveVOS가 이미 설치된 경우 이 단계의 일부 하위 단계를 건너뛰어도 됩니다. 하위 단계 c.를 수행하여 데이터베이스 세부 정보를 입력하고 하위 단계 12f.를 수행하여 ActiveVOS 서버 관리 콘솔에 대한 사용자 이름과 암호를 지정합니다.
 - a. y=Yes에 대해 **Enter** 키를 누릅니다.
 - b. ActiveVOS 서버를 설치할 위치를 지정합니다.
 - c. ActiveVOS 데이터베이스 스키마 작성 시 지정한 데이터베이스 세부 정보를 입력하고 **다음**을 클릭합니다.
참고: 독립 실행형 ActiveVOS를 포함한 ActiveVOS로 이동하는 경우에는 독립 실행형 ActiveVOS 스키마에 대한 세부 정보를 입력합니다.
 - d. MDM 및 ActiveVOS 웹 서비스를 호출하는 데 사용할 URL을 지정합니다. 응용 프로그램 서버에 대한 연결에 사용할 올바른 포트 번호가 URL에 포함되어 있어야 합니다.
 - e. ActiveVOS 설치 프로그램 페이지에서 **선택**을 선택한 후 배포 패키지에서 ActiveVOS_Server 설치 파일을 찾습니다.
 - f. 사용자 이름과 암호를 입력하여 ActiveVOS 서버 관리 콘솔에 대한 관리자를 생성합니다.
중요: 사용자 이름과 암호는 응용 프로그램 서버에서 생성한 ActiveVOS 콘솔 사용자 이름 및 암호와 같아야 합니다.
14. **Enter** 키를 누릅니다.
 Informatica 플랫폼 설치 프롬프트가 나타납니다.
15. Informatica 플랫폼을 설치하려면 예에 대해 **Enter** 키를 누릅니다. 그렇지 않을 경우 아니요에 대해 2를 입력하고 **Enter** 키를 누릅니다.
 Informatica 플랫폼 설치 응답 파일 및 보관 파일 위치를 묻는 메시지가 표시됩니다.
16. Informatica 플랫폼 설치 응답 파일 및 보관 파일의 위치를 입력하고 **Enter** 키를 누릅니다.
17. 제품 사용 톨킷 옵션을 지정합니다.
 - a. 조직이 속하는 산업을 선택한 다음 **Enter** 키를 누릅니다.
 - b. 환경 유형을 입력합니다. 프로덕션의 경우 1을 입력하고, 테스트/QA의 경우 2를 입력하며, 개발의 경우 3을 입력한 다음 **Enter** 키를 누릅니다.
18. 프록시 서버가 있는지 여부를 선택합니다. 예에 대해 **Enter** 키를 누릅니다. 그렇지 않을 경우 아니요에 대해 2를 입력하고 **Enter** 키를 누릅니다.
 다음 프록시 서버 세부 정보를 입력할 수 있습니다.
 - 프록시 서버 이름/IP
 - 프록시 서버 포트
 - 프록시 서버 도메인 이름(해당되지 않는 경우 공백 유지)
 - 프록시 서버 사용자 이름(해당되지 않는 경우 공백 유지)
 - 프록시 서버 암호입니다. 해당되지 않는 경우 비워 둡니다.
 설치 선택 항목의 요약이 표시됩니다.
19. postInstallSetup 스크립트를 설치의 일부분으로 실행할지 아니면 나중에 수동으로 실행할지를 선택합니다.

20. `postInstallSetup` 스크립트를 설치의 일부분으로 실행할지 아니면 나중에 수동으로 실행할지를 선택합니다.

옵션	조건
예, 설치 중에 스크립트를 실행합니다.	<p>설치 완료 후 Hub 서버 EAR 파일을 자동으로 배포합니다.</p> <p>ActiveVOS 시나리오 중 하나에서 다음 응용 프로그램 서버 환경 중 하나를 사용하는 경우 이 옵션을 선택합니다.</p> <p>응용 프로그램 서버 환경:</p> <ul style="list-style-type: none"> • JBoss 독립 실행형 환경 • WebSphere 독립 실행형 환경 <p>ActiveVOS 시나리오:</p> <ul style="list-style-type: none"> • ActiveVOS를 설치하도록 선택했고 이 환경에 다른 버전의 ActiveVOS가 설치되지 않았습다. • ActiveVOS를 설치하도록 선택했고 환경에 지원되는 버전의 ActiveVOS가 포함되어 있습니다. PAM(Product Availability Matrix)에서 지원되는 버전을 확인합니다. • ActiveVOS를 설치하지 않도록 선택했습니다. <p>중요: ActiveVOS를 설치하도록 선택했지만 환경에 지원되지 않는 버전의 ActiveVOS가 있는 경우 아니요, 나중에 실행하겠습니다를 선택합니다.</p>
아니요, 나중에 실행하겠습니다.	<p>Hub 서버 EAR 파일을 수동으로 패키징하고 배포합니다.</p> <p>ActiveVOS를 설치하거나 설치하지 않고 다음 응용 프로그램 서버 환경 중 하나를 사용하는 경우 이 옵션을 선택합니다.</p> <ul style="list-style-type: none"> • WebLogic 독립 실행형 환경 • WebLogic 클러스터 환경 • JBoss 클러스터 환경 • WebSphere 클러스터 환경 <p>지원되지 않는 버전의 ActiveVOS가 포함되는 응용 프로그램 서버 환경을 사용하는 경우 이 옵션을 선택합니다. ActiveVOS 스키마를 지원되는 버전으로 업그레이드한 다음 수동으로 배포해야 합니다.</p>

21. **Enter** 키를 누릅니다.

업그레이드 선택 항목의 요약이 표시됩니다.

22. 업그레이드 전 요약에서 정보를 확인합니다. 정보가 올바르면 **Enter** 키를 눌러 업그레이드를 시작합니다. 정보를 변경해야 하는 경우 특정 정보에 대해 **BACK**을 입력하고 정보를 변경합니다.

프로세스가 완료되면 업그레이드 완료 정보가 표시됩니다.

23. **Enter** 키를 눌러 설치 프로그램을 종료합니다.

자동 모드에서 Hub 서버 업그레이드

자동 모드에서 사용자 상호 작용 없이 Hub 서버를 업그레이드할 수 있습니다. 다중 설치가 있는 경우나 시스템 클러스터에서 업그레이드해야 하는 경우 자동 업그레이드를 수행하고자 할 수 있습니다. 자동 업그레이드에서는 어떠한 진행률 또는 실패 메시지도 표시하지 않습니다.

Hub 서버에 대한 자동 업그레이드를 실행하기 전에 자동 업그레이드에 대한 속성 파일을 구성해야 합니다. 설치 프로그램은 이 파일을 읽어 업그레이드 옵션을 확인합니다. 자동 업그레이드 프로세스는 잘못된 설정(예: 잘못된 응용 프로그램 서버 경로 또는 포트)을 입력한 경우에도 완료될 수 있습니다. 속성 파일에 올바른 설정을 입력했는지 확인해야 합니다.

Hub 서버를 설치할 시스템의 하드 디스크에 Hub 서버 업그레이드 파일을 복사합니다. 자동 모드에서 업그레이드하려면 다음 태스크를 완료합니다.

1. 설치 속성 파일을 구성하고 속성 파일에 설치 옵션을 지정합니다.
2. 설치 속성 파일을 사용하여 업그레이드를 실행합니다.

속성 파일 구성

자동 업그레이드 프로세스에 영향을 미치는 속성 파일에서 매개 변수 값을 확인합니다.

1. Hub 서버를 설치할 때 구성한 속성 파일을 찾습니다.
2. ActiveVOS 서버의 번들로 제공되는 라이선스가 허가된 버전을 설치하는 경우 ActiveVOS 속성을 사용자 속성 파일에 추가합니다.
 - a. 이 릴리스와 함께 제공된 `silentInstallServer_sample.properties` 파일을 엽니다.
 - b. ActiveVOS를 검색합니다.
 - c. ActiveVOS 설치 섹션을 속성 파일에 복사합니다.

```
#####
##### ActiveVOS Server installation #####
#####

## Do you want to install ActiveVOS (Yes/No)
AVOS.INSTALL=Yes
## Path to ActiveVOS Installer (ActiveVOS_Server_windows_9.2.4.3.exe for Windows or
ActiveVOS_Server_unix_9.2.4.3.sh for Linux/UNIX)
AVOS_INSTALLER_PATH=c:\ActiveVOS_Server_windows_9.2.4.3.exe
## ActiveVOS server install directory
AVOS_INSTALL_DIR=C:\infamdm\avos\server

## Database type is the same as for HUB (There is no ability to set a different database type
for ActiveVOS)
## Oracle connection data
## Connection Type SID or Service Name
AVOS.ORACLE.CONNECTION.TYPE="Service Name"
AVOS.DB.SERVER=localhost
AVOS.DB.PORT=1521
## Oracle SID name or service name
AVOS.DB.SID=orcl
AVOS.DB.SCHEMA_NAME=avos
AVOS.DB.PASSWORD=!!cmx!!

## DB2 connection data
AVOS.DB.SERVER=localhost
AVOS.DB.PORT=50000
AVOS.DB.DBNAME=AVOS
AVOS.DB.SCHEMA_NAME=AVOS
AVOS.DB.USER=avos
AVOS.DB.PASSWORD=!!cmx!!

## MSSQL connection data
AVOS.DB.SERVER=localhost
AVOS.DB.PORT=1433
AVOS.DB.DBNAME=avos
AVOS.DB.USER=avos
AVOS.DB.PASSWORD=!!cmx!!

##If you are moving from standalone ActiveVOS to embedded ActiveVOS, enter the details for the
standalone ActiveVOS schema.
```

- d. 속성 파일에서 ActiveVOS 데이터베이스 정보를 지정하고 지원되는 다른 데이터베이스에 대한 속성은 제거합니다. 속성과 관련하여 도움이 필요한 경우 환경에 대한 *Multidomain MDM 설치 가이드*를 참조하십시오.

3. ActiveVOS 서버의 라이선스가 허가된 버전을 사용하는 경우 자동 설치 속성 파일에서 다음 속성을 추가 및 구성합니다.

```
SIP.APPSERVER.WEB.URL=http://localhost:8080
## Avos console's administrator username
AVOS.CONSOLE.USER=aadmin
## Avos console's administrator password
AVOS.CONSOLE.PASSWORD=admin
##The user name and password must be the same as the ActiveVOS Console user name and password
that was created in the application server during the pre-installation process.
```

자동 업그레이드 실행

속성 파일을 구성한 후 자동 업그레이드를 시작할 수 있습니다.

1. 응용 프로그램 서버가 실행 중인지 확인합니다.
2. 명령 창을 엽니다.
3. 다음 명령을 실행합니다.

UNIX의 경우. `./hub_install.bin -f <Hub 서버에 대한 자동 속성 파일 위치>`

Windows의 경우. `hub_install.exe -f <Hub 서버에 대한 자동 속성 파일 위치>`

자동 업그레이드는 백그라운드에서 실행됩니다. 이 프로세스에는 다소 시간이 걸릴 수 있습니다. 자동 설치의 일부로 Hub 서버에 대해 `postInstallSetup` 스크립트를 실행한 경우 업그레이드에 성공했는지 `postinstallSetup.log` 파일을 확인하십시오.

로그 파일은 다음 디렉터리에서 사용할 수 있습니다.

UNIX의 경우. `<MDM Hub 설치 디렉터리>/hub/server/logs/`

Windows의 경우. `<MDM Hub 설치 디렉터리>\hub\server\logs\`

patchInstallSetup 스크립트 실행

Hub 서버 설치 중에 수동으로 배포하도록 선택한 경우에는 `patchInstallSetup` 스크립트를 실행해야 합니다.

1. 다음 디렉터리로 이동합니다. `<MDM Hub 설치 디렉터리> /hub/server`
2. 다음 명령을 실행하여 Hub 서버 응용 프로그램을 배포하고 응용 프로그램 서버 구성에 변경 사항을 적용합니다.

UNIX의 경우

참고: UNIX의 경우 암호에 느낌표(!) 문자를 넣는 경우 느낌표(!) 문자 앞에 백슬래시를 넣어야 합니다. 예를 들어 암호가 `!!cmx!!`이면 `!\!cmx\!!`를 입력합니다.

WebLogic

```
patchInstallSetup.sh -Dweblogic.password=<WebLogic password> -Ddatabase.password=<MDM Hub Master
database password>
-Davos.username=<ActiveVOS Console username> -Davos.password=<ActiveVOS Console password> -
Davos.jdbc.database.password=<ActiveVOS database password>
```

중요: WebLogic 12.2.1.3 이상 환경에서 ActiveVOS를 설치하거나 WebLogic T3S 프로토콜을 사용하기로 결정하는 경우 결정에 해당하는 옵션을 추가합니다.

- ActiveVOS 설치됨. `-Dinstall.avos.patch=true`
- T3S 프로토콜 사용됨. `-Dweblogic.naming.protocol=t3s`

보안이 활성화된 WebSphere

```
patchInstallSetup.sh -Dwebsphere.password=<WebSphere password> -Ddatabase.password=<MDM Hub Master database password> -Davos.username=<ActiveVOS Console username> -Davos.password=<ActiveVOS Console password> -Davos.jdbc.database.password=<ActiveVOS database password>
```

보안이 비활성화된 WebSphere

```
patchInstallSetup.sh -Ddatabase.password=<MDM Hub Master database password> -Davos.username=<ActiveVOS Console username> -Davos.password=<ActiveVOS Console password> -Davos.jdbc.database.password=<ActiveVOS database password>
```

JBoss

```
patchInstallSetup.sh -Ddatabase.password=<MDM Hub Master database password> -Davos.username=<ActiveVOS Console username> -Davos.password=<ActiveVOS Console password> -Davos.jdbc.database.password=<ActiveVOS database password>
```

Windows의 경우

WebLogic

```
patchInstallSetup.bat -Dweblogic.password=<WebLogic password> -Ddatabase.password=<MDM Hub Master database password> -Davos.username=<ActiveVOS Console username> -Davos.password=<ActiveVOS Console password> -Davos.jdbc.database.password=<ActiveVOS database password>
```

중요: WebLogic 12.2.1.3 이상 환경에서 ActiveVOS를 설치하거나 WebLogic T3S 프로토콜을 사용하기로 결정하는 경우 결정에 해당하는 옵션을 추가합니다.

- ActiveVOS 설치됨. -Dinstall.avos.patch=true
- T3S 프로토콜 사용됨. -Dweblogic.naming.protocol=t3s

보안이 활성화된 WebSphere

```
patchInstallSetup.bat -Dwebsphere.password=<WebSphere password> -Ddatabase.password=<MDM Hub Master database password> -Davos.username=<ActiveVOS Console username> -Davos.password=<ActiveVOS Console password> -Davos.jdbc.database.password=<ActiveVOS database password>
```

보안이 비활성화된 WebSphere

```
patchInstallSetup.bat -Ddatabase.password=<MDM Hub Master database password> -Davos.username=<ActiveVOS Console username> -Davos.password=<ActiveVOS Console password> -Davos.jdbc.database.password=<ActiveVOS database password>
```

JBoss

```
patchInstallSetup.bat -Ddatabase.password=<MDM Hub Master database password> -Davos.username=<ActiveVOS Console username> -Davos.password=<ActiveVOS Console password> -Davos.jdbc.database.password=<ActiveVOS database password>
```

업그레이드 기록 폴더에 Hub 서버 로그 파일 복사

Hub 서버 로그 파일 사본을 저장합니다. 업그레이드 문제를 해결해야 하는 경우 이 로그 파일을 사용하여 지원합니다.

upgradedoc 업그레이드 설명서 폴더에 Hub 서버 로그 파일을 복사합니다. 이러한 파일을 hub_server_upgrade와 같은 별도의 하위 폴더에 저장합니다. 클러스터에서 여러 Hub 서버를 업그레이드한 경우, 각 Hub 서버 인스턴스에 대한 파일을 별도의 폴더에 저장합니다.

다음 테이블은 복사할 로그 파일을 설명합니다.

파일	설명
<MDM Hub 설치 디렉터리>/hub/server/Infamdm_Hub_Server_InstallLog.xml	Hub 서버 설치에 대한 로그 메시지가 포함됩니다.
<MDM Hub 설치 디렉터리>/hub/server/infamdm_installer_debug.txt	업그레이드 프로세스 실행 시 선택한 모든 옵션과 디버그 메시지가 포함되어 있습니다.
<MDM Hub 설치 디렉터리>/hub/server/logs/patchInstallSetup.log	patchInstallSetup 스크립트 결과가 포함되어 있습니다.
<MDM Hub 설치 디렉터리>/hub/server/logs/cmserver.log	Hub 서버 로그가 포함되어 있습니다. Hub 서버를 시작할 때 Hub 서버에서 이 파일을 생성합니다.
응용 프로그램 서버 로그 파일입니다.	응용 프로그램 서버에 대한 설치 디렉터리 아래 트리에 있습니다.

Hub 서버 업그레이드 재적용(선택 사항)

Hub 서버 업그레이드를 완료하면 업그레이드 프로세스에서 Hub 서버 업그레이드를 재적용할 수 없습니다. Hub 서버 업그레이드 재적용은 업그레이드 프로세스 중에 하드웨어에서 오류가 발생하는 것과 같은 경우에 합니다. 또한 이 프로시저는 업그레이드를 테스트했다가 소프트웨어의 이전 버전로 되돌리려고 하는 경우에도 수행할 수 있습니다.

- 다음 디렉터리에서 siperian-mrm.ear 파일을 백업합니다.
 - UNIX의 경우. <MDM Hub 설치 디렉터리>/hub/server
 - Windows의 경우. <MDM Hub 설치 디렉터리>\hub\server
- 업그레이드 단계를 반복합니다. -DSIPERIAN_FORCED_PATCH_INSTALL=true 매개 변수를 설치 명령에 추가합니다. 예를 들어 UNIX의 그래픽 모드에서 업그레이드를 재적용하는 경우에는 다음 명령을 실행합니다.


```
hub_install.bin -DSIPERIAN_FORCED_PATCH_INSTALL=true
```

제 7 장

처리 서버 업그레이드(인플레이스 업그레이드)

이 장에 포함된 항목:

- [처리 서버 업그레이드 개요, 92](#)
- [그래픽 모드에서 처리 서버 업그레이드, 92](#)
- [콘솔 모드에서 처리 서버 업그레이드, 94](#)
- [자동 모드에서 처리 서버 업그레이드, 95](#)
- [Informatica 주소 확인 5 통합으로의 업그레이드 단계, 97](#)
- [일치 인구집단 구성, 99](#)
- [업그레이드 설명서 디렉터리에 처리 서버 로그 파일 복사, 101](#)
- [처리 서버 업그레이드 재적용\(선택 사항\), 101](#)

처리 서버 업그레이드 개요

처리 서버는 데이터 정리 작업, 일치 작업 및 일괄 작업을 처리하는 서블릿입니다. 처리 서버를 업그레이드하려면 배포 시 제공된 처리 서버 설치 프로그램을 실행합니다. 업그레이드 프로세스 시 기존 MDM Hub 설치 위치를 대상 위치로 선택하면 처리 서버 설치 프로그램은 업그레이드를 전체 설치와 구분합니다. 처리 서버 설치 프로그램은 기존 처리 서버 설치를 덮어쓰기 전에 중요 파일의 백업을 생성합니다.

참고: 이 장은 인플레이스 업그레이드에만 적용됩니다. 클린 업그레이드의 경우 응용 프로그램 서버 및 데이터베이스 환경에 대한 *Multidomain MDM 설치 가이드*의 지침에 따라 처리 서버를 설치합니다.

그래픽 모드에서 처리 서버 업그레이드

그래픽 모드에서 처리 서버를 업그레이드하려면 처리 서버 설치 프로그램을 실행합니다.

1. 처리 서버를 설치할 때 사용한 사용자 이름으로 로그인합니다.
2. 처리 서버가 배포된 응용 프로그램 서버를 시작합니다.
3. 명령 프롬프트를 열고 배포 디렉터리의 처리 서버 설치 프로그램으로 이동합니다. 기본적으로 설치 프로그램은 다음 디렉터리에 있습니다.

- UNIX의 경우. <MDM Hub 배포 디렉터리><운영 체제 이름>/mrmlcleanse
 - Windows의 경우. <MDM Hub 배포 디렉터리>\windows\mrmlcleanse
4. 다음 명령을 실행합니다.
 - UNIX의 경우. hub_cleanse_install.bin
 - Windows의 경우. hub_cleanse_install.exe
 5. 소개 창에서 **다음**을 클릭합니다.
라이선스 계약 창이 표시됩니다.
 6. 라이선스 계약에 동의함 옵션을 선택한 다음 **다음**을 클릭합니다.
설치 폴더 선택 창이 표시됩니다.
 7. 처리 서버 설치 위치를 선택합니다. 처리 서버 설치 폴더에는 siperian-mrm-cleanse.ear 파일이 포함되어 있습니다.
 - 기본 위치를 선택하려면 **다음**을 클릭합니다.
 - 다른 위치를 선택하려면 **선택**을 클릭한 다음 **다음**을 클릭합니다.
 버전 경고 메시지가 표시됩니다.
 8. **확인**을 클릭하여 계속할지 확인합니다.
라이선스 파일 위치 입력 창이 표시됩니다.
 9. 라이선스 파일의 위치를 선택하고 **다음**을 클릭합니다.
 10. 이전 설치에서 응용 프로그램 서버로 WebLogic을 사용하는 경우 처리 서버 설치 프로그램에 WebLogic 관리 암호를 입력하라는 메시지가 표시됩니다. WebLogic 암호를 입력하십시오.
 11. 제품 사용 툴킷 페이지에서 **환경 유형**을 선택합니다.
 12. 프록시 서버가 있는 경우 **예**를 선택하고 프록시 서버 세부 정보를 입력합니다. 그렇지 않으면 **아니요**를 선택하고 **다음**을 클릭합니다.
다음 프록시 서버 세부 정보를 입력할 수 있습니다.
 - 프록시 서버 이름/IP
 - 프록시 서버 포트
 - 프록시 서버 도메인 이름(해당되지 않는 경우 공백 유지)
 - 프록시 서버 사용자 이름(해당되지 않는 경우 공백 유지)
 - 프록시 서버 암호입니다. 해당되지 않는 경우 비워 둡니다.
 13. **다음**을 클릭합니다.
배포 페이지가 표시됩니다.
 14. 자동 또는 수동으로 배포할지 선택합니다. **예**를 클릭하여 자동으로 배포하거나 **아니요**를 클릭하여 수동으로 배포한 후 **다음**을 클릭합니다.
 - WebSphere 독립 실행형 환경 또는 JBoss 독립 실행형 환경에서 **예**를 클릭하여 자동으로 배포한 후 **다음**을 클릭합니다.
 - WebLogic 또는 클러스터된 환경에서 **아니요**를 클릭하여 수동으로 배포한 후 **다음**을 클릭합니다.
 설치 전 요약 창이 표시됩니다.
 15. 옵션을 변경하려면 **이전** 단추를 클릭하여 이전 선택 항목을 변경합니다.
 16. 요약 창에 원하는 옵션이 표시되고 나면 **설치**를 클릭하여 설치 프로세스를 시작합니다.

처리 서버 설치 프로그램이 시스템을 구성하는 동안 이 프로그램에 **잠시 기다려 주십시오**. 화면이 표시됩니다. 처리 서버 설치 프로그램은 MDM Hub 설치 디렉터리의 backup 폴더에 저장된 보관에 중요 파일을 백업합니다. 보관의 파일 이름은 다음 예에 나와 있는 형식을 사용합니다.

Siperian Hub Cleanse Match Server-2010-05-12_18-09.jar

설치가 완료되면 **설치 완료** 창이 표시됩니다.

17. **완료**를 클릭하여 처리 서버 설치 프로그램을 종료합니다.

참고: 업그레이드가 성공적으로 완료되지 않은 경우 업그레이드가 실패했음을 나타내는 창이 표시되고 실패 메시지가 포함된 로그 파일의 위치가 표시됩니다.

18. 14단계에서 **아니요**를 선택한 경우 EAR 파일을 다시 패키징하고 수동으로 배포합니다.

- a. 다음 명령을 실행하여 EAR 파일을 다시 패키징합니다.

UNIX의 경우.

```
cd <MDM Hub 설치 디렉터리>/hub/cleanse/bin
./sip_ant.sh repack
```

Windows의 경우.

```
cd <MDM Hub 설치 디렉터리>\hub\cleanse\bin
sip_ant.bat repack
```

- b. 응용 프로그램 서버 관리 콘솔에서 처리 서버 EAR 파일을 수동으로 배포합니다. 응용 프로그램 서버 설 명서를 참조하십시오.

19. SSA-Name3 라이브러리 파일을 <MDM Hub 설치 디렉터리>/hub/cleanse/lib/upgrade/SSA에서 <MDM Hub 설치 디렉 터리>/hub/cleanse/lib로 복사합니다.

20. 응용 프로그램 서버를 다시 시작합니다.

콘솔 모드에서 처리 서버 업그레이드

UNIX에서 콘솔 모드로 처리 서버를 업그레이드할 수 있습니다.

참고: RedHat Linux에서 처리 서버를 업그레이드할 때는 루트 사용자를 사용하지 마십시오. 루트 사용자에게는 InstallAnywhere에 필요한 .profile이 없습니다. 대신 별도의 사용자 프로필을 생성하여 처리 서버를 업그레이 드하는 데 사용하십시오.

1. 응용 프로그램 서버를 시작합니다.

2. MDM Hub 배포의 다음 디렉터리로 이동합니다.

Solaris의 경우. <MDM Hub 배포 디렉터리>/solaris/mrmcleanse

HP-UX의 경우. <MDM Hub 배포 디렉터리>/hpux/mrmcleanse

Linux의 경우. <MDM Hub 배포 디렉터리>/linux/mrmcleanse

AIX의 경우. <MDM Hub 배포 디렉터리>/aix/mrmcleanse

3. 명령 프롬프트에서 다음 명령을 실행합니다.

```
./hub_cleanse_install.bin -i console
```

4. 설치에 사용할 로컬의 번호를 입력하고 **Enter** 키를 누릅니다.

설치에 관한 소개 정보가 표시됩니다.

5. **Enter** 키를 누릅니다.

라이선스 계약이 표시됩니다.

6. 라이선스 계약의 내용을 확인합니다. 사용권 계약에 동의하려면 **Y**를 입력하고, 사용권 계약에 동의하지 않고 설치 프로그램을 종료하려면 **N**을 입력합니다.
7. **Enter** 키를 누릅니다.
이전 단계에서 **Y**를 입력한 경우 설치 폴더에 대한 정보가 표시됩니다.
8. 처리 서버를 설치한 디렉터리를 지정합니다.
 - 기본 위치를 선택하려면 **Enter** 키를 누릅니다.
 - 경로를 변경하려면 설치 폴더의 절대 경로를 입력하고 **Enter** 키를 누릅니다.
9. 설치 폴더의 위치를 확인합니다. 설치 폴더를 확인하려면 **Y**를 입력하고, 설치 폴더를 변경하려면 **N**을 입력합니다.
버전 경고 메시지가 표시됩니다.
10. **Enter** 키를 눌러 계속할지 확인합니다.
라이선스 파일 위치를 묻는 메시지가 표시됩니다.
11. 라이선스 파일의 절대 경로를 입력하고 **Enter** 키를 누릅니다.
12. WebLogic 환경에서 WebLogic 암호를 입력하고 **Enter** 키를 누릅니다.
13. 제품 사용 툴킷 옵션에서 환경 유형을 선택합니다. 프로덕션의 경우 1을 입력하고, 테스트/QA의 경우 2를 입력하며, 개발의 경우 3을 입력한 다음 **Enter** 키를 누릅니다.
14. 프록시 서버가 있는지 여부를 선택합니다. 예에 대해 **Enter** 키를 누릅니다. 그렇지 않을 경우 아니요에 대해 2를 입력하고 **Enter** 키를 누릅니다.
다음 프록시 서버 세부 정보를 입력할 수 있습니다.
 - 프록시 서버 이름/IP
 - 프록시 서버 포트
 - 프록시 서버 도메인 이름(해당되지 않는 경우 공백 유지)
 - 프록시 서버 사용자 이름(해당되지 않는 경우 공백 유지)
 - 프록시 서버 암호입니다. 해당되지 않는 경우 비워 둡니다.
 설치 선택 항목의 요약이 표시됩니다.
15. postInstallSetup 스크립트를 설치의 일부분으로 실행할지 아니면 나중에 수동으로 실행할지를 선택합니다.
16. **Enter** 키를 누릅니다.
업그레이드 선택 항목의 요약이 표시됩니다.
17. 업그레이드 전 요약에서 정보를 확인합니다. 정보가 올바르면 **Enter** 키를 눌러 업그레이드를 시작합니다.
정보를 변경해야 하는 경우 특정 정보에 대해 **BACK**을 입력하고 정보를 변경합니다.
프로세스가 완료되면 업그레이드 완료 정보가 표시됩니다.
18. **Enter** 키를 눌러 설치 프로그램을 종료합니다.

자동 모드에서 처리 서버 업그레이드

자동 모드에서 사용자 상호 작용 없이 처리 서버를 업그레이드할 수 있습니다. 다중 설치가 있는 경우나 시스템 클러스터에서 업그레이드해야 하는 경우 자동 업그레이드를 수행하고자 할 수 있습니다. 자동 업그레이드에서는 어떠한 진행률 또는 실패 메시지도 표시하지 않습니다.

처리 서버에 대한 자동 업그레이드를 실행하기 전에 자동 업그레이드에 대한 속성 파일을 구성해야 합니다. 설치 프로그램은 이 파일을 읽어 업그레이드 옵션을 확인합니다. 자동 업그레이드 프로세스는 잘못된 설정(예: 잘못된

응용 프로그램 서버 경로 또는 포트 설정)을 입력한 경우에도 완료될 수 있습니다. 속성 파일에 올바른 설정을 입력했는지 확인해야 합니다.

처리 서버를 업그레이드할 시스템의 하드 디스크에 처리 서버 업그레이드 파일을 복사합니다. 자동 모드에서 업그레이드하려면 다음 태스크를 완료합니다.

1. 설치 속성 파일을 구성하고 속성 파일에 설치 옵션을 지정합니다.
2. 설치 속성 파일을 사용하여 업그레이드를 실행합니다.

속성 파일 구성

자동 업그레이드 프로세스에 영향을 미치는 속성 파일에서 매개 변수 값을 확인합니다.

1. 처리 서버를 설치할 때 구성한 속성 파일을 찾습니다.
2. 텍스트 편집기를 사용하여 파일을 열고 자동 업그레이드 프로세스에 영향을 미치는 매개 변수의 값을 확인합니다.

다음 테이블에서는 확인할 업그레이드 매개 변수에 대해 설명합니다.

속성 이름	설명
USER_INSTALL_DIR	처리 서버를 설치한 디렉터리입니다. 예를 들면 C:\<MDM Hub 설치 디렉터리>\cleanse입니다. 속성 파일의 백슬래시 문자를 이스케이프해야 합니다. 설치 디렉터리 경로를 지정할 때 이중 백슬래시를 사용합니다.
SIP.APPSERVER.PASSWORD	WebLogic에 액세스하는 데 필요한 암호입니다. WebLogic 환경용입니다.
RUN_DEPLOYMENT_FLAG	자동 업그레이드의 일부로 postInstallSetup 스크립트를 실행합니다. 자동 업그레이드 종료 시 postInstallSetup을 실행하려면 1로 설정합니다. postInstallSetup을 실행하지 않으려면 0으로 설정합니다.

3. 다음 제품 사용 툴킷 속성을 자동 설치 속성 파일에 추가 및 구성합니다.

```
#Product Usage Toolkit Installation
#CSM_TYPE is the type of Product Usage Toolkit installation.
# valid values are:Production,Test,Development. Should not be blank.
CSM_TYPE=Production

# If the network has a proxy server, fill in the following parameters (leave empty if no proxy):
# proxy server host
CSM_HOST=
# proxy server port
CSM_PORT=
# Proxy server domain name (leave blank, if not applicable)
CSM_DOMAIN=
# Proxy server user name (leave blank, if not applicable)
CSM_PROXY_USER_NAME=
#Proxy server password (leave blank, if not applicable)
CSM_PROXY_PASSWORD=
```

처리 서버 자동 업그레이드 실행

속성 파일을 구성한 후 자동 업그레이드를 시작할 수 있습니다.

1. 응용 프로그램 서버가 실행 중인지 확인합니다.
2. 명령 창을 엽니다.

3. 다음 명령을 실행합니다.

UNIX의 경우. `./hub_cleansetup_install.bin -f <처리 서버에 대한 자동 속성 파일 위치>`

Windows의 경우. `hub_cleansetup_install.exe -f <처리 서버에 대한 자동 속성 파일 위치>`

자동 업그레이드는 백그라운드에서 실행됩니다. 이 프로세스에는 다소 시간이 걸릴 수 있습니다. 자동 설치의 일부로 처리 서버에 대해 설치 후 스크립트를 실행한 경우 업그레이드에 성공했는지 `postinstallSetup.log` 파일을 확인하십시오.

로그 파일은 다음 디렉터리에서 사용할 수 있습니다.

UNIX의 경우. <MDM Hub 설치 디렉터리>/hub/cleanse/logs/

Windows의 경우. <MDM Hub 설치 디렉터리>\hub\cleanse\logs\

Informatica 주소 확인 5 통합으로의 업그레이드 단계

이 섹션에서는 Informatica 주소 확인 5를 사용하기 위해 MDM Hub 구현에 필요한 업그레이드 프로세스에 대해 설명합니다.

참고: 이 섹션은 Informatica 주소 확인 사용 관련 라이선스를 가진 사용자에게 적용됩니다.

Informatica 주소 확인 5 통합으로 업그레이드하려면 다음 단계를 수행해야 합니다.

1. `cmxcleanse.properties` 파일을 엽니다. 이 파일의 위치는 다음과 같습니다.

Windows: `<infamdm_install_directory>\hub\cleanse\resources`

UNIX: `<infamdm_install_directory>/hub/cleanse/resources`

2. 다음 Informatica 주소 확인 5 속성이 `cmxcleanse.properties` 파일에 설정되어 있는지 확인합니다.

Windows:

```
cleanse.library.addressDoctor.property.SetConfigFile=C:\infamdm\hub\cleanse\resources\AddressDoctor\5\SetConfig.xml
cleanse.library.addressDoctor.property.ParametersFile=C:\infamdm\hub\cleanse\resources\AddressDoctor\5\Parameters.xml
cleanse.library.addressDoctor.property.DefaultCorrectionType=PARAMETERS_DEFAULT
```

UNIX:

```
cleanse.library.addressDoctor.property.SetConfigFile=/u1/infamdm/hub/cleanse/resources/AddressDoctor/5/SetConfig.xml
cleanse.library.addressDoctor.property.ParametersFile=/u1/infamdm/hub/cleanse/resources/AddressDoctor/5/Parameters.xml
cleanse.library.addressDoctor.property.DefaultCorrectionType=PARAMETERS_DEFAULT
```

3. 속성 파일을 저장하고 닫습니다.
4. `SetConfig.xml` 및 `Parameters.xml`을 `cmxcleanse.properties` 파일에서 지정한 위치에 복사합니다.

다음은 샘플 `SetConfig.xml` 파일입니다.

```
<!DOCTYPE SetConfig SYSTEM 'SetConfig.dtd'>
<SetConfig>
  <General WriteXMLEncoding="UTF-16LE" WriteXMLBOM="NEVER"
    MaxMemoryUsageMB="600" MaxAddressObjectCount="10" MaxThreadCount="10" />

  <UnlockCode>79FYUAXAVSR0KLV1TDC6PAQVVC3KM14FZC</UnlockCode>

  <DataBase CountryISO3="ALL" Type="BATCH_INTERACTIVE" Path="c:\addressdoctor\5"
    PreloadingType="NONE" />

  <DataBase CountryISO3="ALL" Type="FASTCOMPLETION" Path="c:\addressdoctor\5"
    PreloadingType="NONE" />
```

```

<DataBase CountryISO3="ALL" Type="CERTIFIED" Path="c:\addressdoctor\5"
  PreloadingType="NONE" />

<DataBase CountryISO3="ALL" Type="GEOCODING" Path="c:\addressdoctor\5"
  PreloadingType="NONE" />

  <DataBase CountryISO3="ALL" Type="SUPPLEMENTARY" Path="c:\addressdoctor\5"
    PreloadingType="NONE" />
</SetConfig>

```

다음은 샘플 Parameters.xml 파일입니다.

```

<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE Parameters SYSTEM 'Parameters.dtd'>
<Parameters
  WriteXMLEncoding="UTF-16LE"
  WriteXMLBOM="NEVER">
  <Process
    Mode="BATCH"
    EnrichmentGeoCoding="ON"
    EnrichmentCASS="ON"
    EnrichmentSERP="ON"
    EnrichmentSNA="ON"
    EnrichmentSupplementaryGB="ON"
    EnrichmentSupplementaryUS="ON" />
  <Input
    Encoding="UTF-16LE"
    FormatType="ALL"
    FormatWithCountry="ON"
    FormatDelimiter="PIPE" />
  <Result
    AddressElements="STANDARD"
    Encoding="UTF-16LE"
    CountryType="NAME_EN"
    FormatDelimiter="PIPE" />
</Parameters>

```

5. 구성 파일 SetConfig.xml에서 Informatica 주소 확인 5 잠금 해제 코드를 지정합니다.

SetConfig.xml 파일 및 Parameters.xml 파일에 대한 자세한 내용은 Informatica 주소 확인 5 설명서를 참조하십시오.

6. 다음 위치에서 Informatica 주소 확인 5 라이브러리를 복사합니다.

Windows: <infadm_install_directory>\hub\cleanse\lib\upgrade\AddressDoctor

UNIX: <infadm_install_directory>/hub/cleanse/lib/upgrade/AddressDoctor

7. JADE.dll(또는 이와 동등한 Informatica 주소 확인 4 라이브러리)을 다음 위치의 Informatica 주소 확인 5 라이브러리로 대체합니다.

Windows: <infadm_install_directory>\hub\cleanse\lib

UNIX: <infadm_install_directory>/hub/cleanse/lib

자세한 내용은 libupdate_readme.txt 문서를 참조하십시오. 이 문서의 위치는 다음과 같습니다.

Windows: <infadm_install_directory>\hub\cleanse\lib\upgrade

UNIX: <infadm_install_directory>/hub/cleanse/lib/upgrade

8. 응용 프로그램 서버를 다시 시작합니다.

현재 응용 프로그램을 실행 중인 사용자 이름과 같은 사용자 이름으로 로그인했는지 확인하고 응용 프로그램 서버를 시작하는 동안 예외가 발생하지 않았는지 확인합니다.

9. 처리 서버를 다시 시작합니다.

처리 서버를 초기화하는 동안 터미널 콘솔에 다음과 비슷한 메시지가 표시됩니다.

```
[INFO ] com.siperian.mrm.cleanse.addressDoctor.Library: Initializing AddressDoctor5
```

10. 정리 함수 도구를 시작합니다.

11. 쓰기 잠금을 획득합니다(쓰기 잠금 > 잠금 획득).

12. Informatica 주소 확인 정리 함수를 선택합니다.

13. 새로 고침 단추를 클릭합니다.

Informatica 주소 확인 5 정리 함수가 Informatica 주소 확인 정리 함수 노드에 추가되었습니다.

일치 인구집단 구성

일치 인구집단은 일치 프로세스에 사용할 표준 인구집단 설정을 포함합니다. 지원되는 국가, 언어 또는 인구집단마다 표준 인구집단 설정이 있습니다. 일치 규칙에 사용할 일치 인구집단을 활성화해야 합니다.

일치 인구집단은 Informatica MDM Hub 설치 시 *population.ysp* 파일로 사용할 수 있습니다. 인구집단 이름은 *ysp* 파일 이름과 같습니다. 일본 인구집단을 추가하고 **Person_Name_Kanji** 일치 필드를 사용하려면 인구집단 이름에 **_Kanji**를 추가합니다(예: **Japan_Kanji** 또는 **Japan_i_Kanji**). 이 작업을 수행한 경우 표준 **Person_Name** 일치 필드는 사용할 수 없습니다.

사용하는 인구집단은 MDM Hub의 **SSA-Name3** 버전과 호환되어야 합니다. 추가 채우기 파일이 필요하거나 업데이트된 채우기 파일을 사용하여 이후 버전으로 업그레이드해야 하는 경우 Informatica 글로벌 고객 지원 센터에 문의하십시오. 제품에서 요청한 첫 번째 인구집단 파일은 무료로 제공됩니다. 다른 국가용 인구집단 파일이 필요하거나 최신 버전의 MDM Hub로 업그레이드하기 위해 업데이트된 인구집단 파일이 필요할 수 있습니다.

사용자 지정 채우기 파일 업그레이드

업그레이드 프로세스 중에 채우기 파일이 업그레이드됩니다. 채우기 파일을 사용자 지정한 경우 Informatica 글로벌 고객 지원 센터에 업그레이드된 속성 파일에 사용자 지정을 적용해 줄 것을 요청하십시오.

일치 인구집단 활성화

일치 규칙에 사용할 일치 인구집단을 활성화해야 합니다.

1. <인구집단>.ysp 파일을 다음 위치에 복사합니다.

UNIX의 경우. <MDM Hub 설치 디렉터리>/hub/cleanse/resources/match

Windows의 경우. <MDM Hub 설치 디렉터리>\hub\cleanse\resources\match

2. **C_REPOS_SSA_POPULATION** 메타데이터 테이블에서 인구집단이 등록되어 있는지 확인합니다.

MDM Hub 설치에 대한 시드 데이터베이스의 **C_REPOS_SSA_POPULATION** 테이블에 몇 개의 인구집단이 등록되어 있지만 활성화되어 있지는 않습니다.

3. **C_REPOS_SSA_POPULATION** 테이블에 인구집단이 포함되어 있지 않으면 인구집단을 추가하고 활성화합니다.

인구집단 이름은 *ysp* 파일 이름과 같습니다. 예를 들어 *ysp* 파일 이름이 **US.ysp**인 경우 인구집단 이름은 **US**입니다.

연산 참조 저장소에 인구집단을 추가하려면 다음 단계를 따릅니다.

IBM Db2 또는 Oracle의 경우.

a. 인구집단을 추가할 연산 참조 저장소 스키마에 연결합니다.

b. **SQL*Plus**에서 다음 디렉터리에 있는 **add_std_ssa_population.sql** 스크립트를 실행합니다.

UNIX의 경우. <MDM Hub 설치 디렉터리>/server/resources/database/custom_scripts/oracle

Windows의 경우. <MDM Hub 설치 디렉터리>\server\resources\database\custom_scripts\oracle

- c. 다음 테이블에 설명되어 있는 프롬프트에 응답합니다.

프롬프트	설명
추가할 인구집단을 입력합니다.	인구집단의 이름입니다.
ROWID_SSA_POP 값(예: INFA.0001) 입력, 기본값[INFA.0001]	C_REPOS_SSA_POPULATION 메타데이터 테이블의 ROWID_SSA_POP 열에 대한 고유한 값입니다. 기본값은 INFA.0001입니다.

인구집단이 C_REPOS_SSA_POPULATION 테이블에 등록됩니다.

- d. 다음 명령을 실행하여 인구집단을 활성화합니다.

```
UPDATE c_repos_ssa_population SET enabled_ind = 1 WHERE population_name = '<Your Population>';
COMMIT;
```

Microsoft SQL Server의 경우.

- a. 다음 디렉터리에 있는 add_std_ssa_population.bat 스크립트를 실행합니다.

<MDM Hub 설치 디렉터리>\server\resources\database\custom_scripts\MSSQL

- b. 다음 테이블에 설명되어 있는 프롬프트에 응답합니다.

프롬프트	설명
CMX_ORS DB에 대한 MSSQL 인스턴스가 있는 호스트 이름("localhost")	Microsoft SQL Server 인스턴스의 호스트 이름입니다.
cmx_ors 사용자 이름("cmx_ors")	연산 참조 저장소의 사용자 이름입니다.
cmx_ors 사용자 암호	연산 참조 저장소의 암호입니다.
인구집단 이름(참고: 일본 인구집단 또는 Japan_i population에 Person_Name_Kanji를 사용하는 경우 이름 끝에 접미사 '_Kanji'를 추가) 입력, 기본값("")	인구집단의 이름입니다.
ROWID_SSA_POP 값(예: INFA.0001) 입력, 기본값(INFA.0001)	C_REPOS_SSA_POPULATION 메타데이터 테이블의 ROWID_SSA_POP 열에 대한 고유한 값입니다.

인구집단이 C_REPOS_SSA_POPULATION 테이블에 등록됩니다.

- c. 다음 명령을 실행하여 인구집단을 활성화합니다.

```
USE <연산 참조 저장소 사용자>
GO
UPDATE [dbo].[C_REPOS_SSA_POPULATION] SET ENABLED_IND = 1 WHERE POPULATION_NAME = '<인구집단>'
```

4. 처리 서버를 다시 시작합니다.

5. Hub 콘솔에 로그인하여 인구집단이 활성화되었는지 확인합니다.

인구집단이 기본 개체의 **일치/병합 설정** 사용자 인터페이스에 표시됩니다.

업그레이드 설명서 디렉터리에 처리 서버 로그 파일 복사

Hub 서버 로그 파일 사본을 저장합니다. 업그레이드 문제를 해결해야 하는 경우 이 로그 파일을 사용하여 지원합니다.

업그레이드 기록 폴더에 처리 서버 로그 파일을 복사합니다. 이러한 파일을 `cleanse_match_server_upgrade`와 같은 별도의 하위 폴더에 저장합니다. 클러스터에서 여러 처리 서버를 업그레이드한 경우 각 처리 서버 인스턴스의 파일을 별도의 폴더에 저장합니다.

다음 테이블은 복사할 로그 파일을 설명합니다.

파일	설명
<정리 설치 디렉터리>/hub/cleanse/Infamdm_Hub_Cleanse_Match_Server_InstallLog.xml	처리 서버 설치에 대한 로그 파일이 포함되어 있습니다.
<정리 설치 디렉터리>/hub/cleanse/infamdm_installer_debug.txt	업그레이드 프로세스 실행 시 선택한 모든 옵션과 디버그 메시지가 포함되어 있습니다.
<정리 설치 디렉터리>/hub/cleanse/logs/patchInstallSetup.log	patchInstallSetup 스크립트 결과가 포함되어 있습니다.
<정리 설치 디렉터리>/hub/cleanse/logs/cmserver.log	처리 서버 로그가 포함되어 있습니다.
응용 프로그램 서버 로그 파일입니다.	응용 프로그램 서버에 대한 설치 디렉터리 아래 트리에 있습니다.

처리 서버 업그레이드 재적용(선택 사항)

처리 서버 업그레이드를 완료하면 업그레이드 프로세스에서 처리 서버 업그레이드를 재적용할 수 없습니다. 처리 서버 업그레이드는 업그레이드 프로세스 중에 하드웨어에서 오류가 발생하는 것과 같은 경우에 재적용을 합니다. 또한 이 프로시저는 업그레이드를 테스트했다가 소프트웨어의 이전 버전으로 되돌리려고 하는 경우에도 수행할 수 있습니다.

- 다음 디렉터리에서 `siperian-mrm.ear` 파일을 백업합니다.
 - UNIX의 경우. <MDM Hub 설치 디렉터리>/hub/cleanse
 - Windows의 경우. <MDM Hub 설치 디렉터리>\hub\cleanse
- 업그레이드 단계를 반복합니다. `-DSIPERIAN_FORCED_PATCH_INSTALL=true` 매개 변수를 설치 명령에 추가합니다. 예를 들어 UNIX의 그래픽 모드에서 업그레이드를 재적용하는 경우에는 다음 명령을 실행합니다.

```
hub_cleanse_install.bin -DSIPERIAN_FORCED_PATCH_INSTALL=true
```

제 8 장

리소스 키트 업그레이드 (인플레이스 업그레이드)

이 장에 포함된 항목:

- [리소스 키트 업그레이드 개요, 102](#)
- [리소스 키트 제거\(인플레이스 업그레이드\), 102](#)
- [리소스 키트 설치, 103](#)

리소스 키트 업그레이드 개요

리소스 키트 현재 버전으로 업그레이드하려면 리소스 키트 전체 설치를 수행합니다. 리소스 키트 패치 설치를 수행할 수 없습니다.

1. 인플레이스 업그레이드를 수행하는 경우 리소스 키트를 제거합니다.
2. 리소스 키트 현재 버전을 설치합니다.

리소스 키트 제거(인플레이스 업그레이드)

리소스 키트의 설치된 버전을 제거하려면 설치 제거 프로그램을 실행해야 합니다. 설치 제거 프로그램은 리소스 키트 디렉터리에 있습니다.

1. 응용 프로그램 서버를 시작합니다.
2. 명령 프롬프트를 열고 다음 디렉터리로 이동합니다.
UNIX의 경우. <리소스 키트 설치 디렉터리>/deploy/UninstallerData
Windows의 경우. <리소스 키트 설치 디렉터리>\deploy\UninstallerData
3. 다음 명령을 실행합니다.
UNIX의 경우. Uninstall_Informatica_MDM_Hub_Resource_Kit.bin
Windows의 경우. Uninstall Informatica MDM Hub Resource Kit.exe
4. **제거**를 클릭합니다.
제거 완료 창이 표시됩니다.

5. 리소스 키트 이전 버전을 제거합니다.
6. **완료**를 클릭합니다.

리소스 키트 설치

리소스 키트를 제거한 후 현재 버전의 리소스 키트를 설치합니다.

1. 응용 프로그램 서버를 시작합니다.
2. 배포 디렉터리의 리소스 키트 설치 프로그램으로 이동합니다. 기본적으로 설치 프로그램은 다음 디렉터리에 있습니다.

UNIX의 경우. <MDM Hub 배포 디렉터리>/<운영 체제 이름>/mrmresourcekit

Windows의 경우. <MDM Hub 배포 디렉터리>\windows\mrmresourcekit

3. 다음 명령을 실행합니다.

UNIX의 경우. hub_resourcekit_install.bin

Windows의 경우. hub_resourcekit_install.exe

4. 설치에 사용할 언어를 선택한 다음 **확인**을 클릭합니다.

소개 창이 표시됩니다.

5. **다음**을 클릭합니다.

라이선스 계약 창이 표시됩니다.

6. **라이선스 계약에 동의함** 옵션을 선택한 다음 **다음**을 클릭합니다.

설치 기능 창이 표시됩니다.

7. 설치할 리소스 키트 기능을 선택하고 **다음**을 클릭합니다.

다음 리소스 키트 기능을 선택할 수 있습니다.

샘플 스키마

샘플 스키마를 설치하려면 이 옵션을 선택합니다. 샘플 응용 프로그램을 설치하려면 먼저 샘플 스키마를 생성하여 Hub 서버에 등록해야 합니다.

샘플 및 유틸리티

샘플 응용 프로그램과 유틸리티를 설치하려면 이 옵션을 선택합니다.

배포한 샘플 응용 프로그램의 목록은 build.properties 파일에 저장되어 있습니다.

build.properties 파일은 다음 디렉터리에 있습니다. <MDM Hub 설치 디렉터리>\hub\resourcekit\samples

SIF SDK 및 Javadocs

SIF SDK와 연관된 Javadocs, 라이브러리 및 리소스를 설치하려면 이 옵션을 선택합니다.

BPM SDK

BPM SDK와 연관된 리소스를 설치하려면 이 옵션을 선택합니다.

Jaspersoft

Jaspersoft 설치 프로그램을 리소스 키트 홈에 복사하려면 이 옵션을 선택합니다.

SSA-NAME3

SSA-NAME3 설치 프로그램을 리소스 키트 홈에 복사하려면 이 옵션을 선택합니다.

설치 폴더 선택 창이 표시됩니다.

8. 리소스 키트 설치 위치를 선택합니다.

- 기본 위치를 선택하려면 **다음**을 클릭합니다.
- 경로를 입력하려면 설치 폴더 경로를 입력하고 **다음**을 클릭합니다.
참고: 폴더 이름에 공백이 들어가도록 지정하면 설치 오류가 발생합니다.
- 기본 설치 위치로 돌아가려면 **기본 폴더 복원**을 클릭합니다.
- 다른 위치를 선택하려면 **선택**을 클릭한 다음 **다음**을 클릭합니다.

UNIX의 경우 **링크 폴더 선택** 창이 표시됩니다. Windows의 경우 **바로 가기 폴더 선택** 창이 표시됩니다.

9. 제품 아이콘 또는 링크를 생성할 위치를 선택하거나 제품 아이콘 또는 링크를 생성하지 않는 옵션을 선택한 후 **다음**을 클릭합니다.

구성 선택 창이 표시됩니다.

10. 구성 옵션을 선택하고 **다음**을 클릭합니다.

다음 구성 옵션을 선택할 수 있습니다.

샘플 구성

샘플을 구성합니다.

소스만

샘플의 소스를 설치합니다.

샘플 구성을 선택하면 **리소스 키트 응용 프로그램 서버** 창이 표시됩니다. **소스만**을 선택하면 **설치 전 요약** 창이 표시됩니다.

11. **리소스 키트 응용 프로그램 서버** 창에서 리소스 키트를 설치할 응용 프로그램 서버를 선택하고 **다음**을 클릭합니다.

선택한 응용 프로그램 서버의 **응용 프로그램 서버 홈** 창이 표시됩니다.

12. 응용 프로그램 서버 설정을 구성합니다.

- JBoss 설정을 다음과 같이 구성합니다.
 1. 응용 프로그램 서버 설치 디렉터리를 지정한 다음 **다음**을 클릭합니다.
JBoss 응용 프로그램 서버 구성 이름 창이 표시됩니다.
 2. 구성 이름을 지정한 다음 **다음**을 클릭합니다.
기본 값은 독립 실행형입니다.
 3. 원격 포트를 지정합니다.
- WebLogic 설정을 다음과 같이 구성합니다.
 1. MDM Hub에 사용할 도메인의 WebLogic 도메인 설치 디렉터리 경로를 선택하고 **다음**을 클릭합니다.
WebLogic 응용 프로그램 서버 로그인 창이 표시됩니다.

2. WebLogic 서버 로그인 정보를 입력합니다.

다음 로그인 매개 변수를 구성할 수 있습니다.

호스트

WebLogic이 설치되어 있는 호스트 컴퓨터의 이름입니다.

서버

WebLogic이 배포되어 있는 도메인의 WebLogic 서버 인스턴스의 이름입니다.

사용자 이름

WebLogic을 설치하는 사용자의 이름입니다.

암호

WebLogic 사용자에게 해당하는 암호입니다.

포트 번호

WebLogic 서버가 수신하는 포트 번호입니다.

- WebSphere 설정을 다음과 같이 구성합니다.

1. WebSphere 응용 프로그램 서버의 경로를 선택하고 **다음**을 클릭합니다.

미리 알림 창이 표시됩니다.

2. 선행 조건을 수행했는지 확인하고 **확인**을 클릭합니다.

WebSphere 보안 선택 창이 표시됩니다.

3. WebSphere 보안 활성화 여부를 선택하고 **다음**을 클릭합니다.

아니요를 선택하고 **다음**을 클릭하면 **WebSphere 응용 프로그램 서버 포트** 창이 표시됩니다. 기본 값은 **아니요**입니다.

WebSphere 응용 프로그램 서버의 서버 이름과 RMI 및 SOAP 포트를 설정합니다. 클러스터된 환경에서는 클러스터 서버 이름 중 하나와 해당 SOAP 및 RMI 포트 정보를 입력합니다.

예를 선택하고 **다음**을 클릭하면 **WebSphere 응용 프로그램 서버 포트 및 사용자 자격 증명** 창이 표시됩니다. WebSphere 사용자 이름 및 WebSphere 암호를 지정합니다.

Informatica MDM Hub 서버 창이 표시됩니다.

13. Hub 서버 설치에 대한 정보를 입력하고 **다음**을 클릭합니다.

다음 Hub 서버 정보를 지정할 수 있습니다.

서버 이름

Hub 서버를 호스팅하는 서버의 이름.

서버 HTTP 포트

Hub 서버의 포트 번호.

Informatica MDM 관리 암호

MDM Hub에 액세스하는 데 필요한 암호.

리소스 키트 ORS ID 창이 표시됩니다.

14. 목록에서 리소스 키트 ORS ID를 선택하고 **다음**을 클릭합니다.

이 목록에는 생성한 연산 참조 저장소 ID가 포함되어 있습니다. 샘플 스키마와 관련된 연산 참조 저장소 ID를 선택합니다.

샘플 스키마를 등록하지 않은 경우에는 샘플 스키마의 연산 참조 저장소 ID가 표시되지 않습니다. 샘플 연산 참조 저장소를 등록한 다음 설치를 다시 시작합니다.

배포 선택 창이 표시됩니다.

15. 배포 옵션을 선택한 후 **다음**을 클릭합니다.

다음 배포 옵션 중 하나를 선택할 수 있습니다.

예, 설치 중에 실행합니다.

설치 중에 리소스 키트를 배포하고 구성합니다.

아니요, 나중에 배포합니다.

이후에 수동으로 배포하고 구성하려면 이 옵션을 선택합니다.

샘플 및 유틸리티 기능을 설치하도록 선택하면 이 설치 단계에서 리소스 키트를 배포하고 구성합니다. 이 설치 단계에서 리소스 키트를 배포하고 구성하지 않는 경우에는 `postInstallSetup` 스크립트를 사용하여 변경 사항을 적용할 수도 없고 샘플을 재배포할 수도 없습니다.

설치 후 설정을 수동으로 실행하도록 선택하면 이후에 `postInstallSetup` 스크립트를 사용하여 `EAR` 파일을 배포할 수 없습니다. 설치에 어떤 변경 사항이든 적용하려면 `EAR` 파일을 수동으로 편집하고 배포해야 합니다.

설치 전 요약 창이 표시됩니다.

16. 설치 전 요약을 검토하여 설치 선택 사항을 확인한 다음 **설치**를 클릭합니다.

설치가 완료되면 **설치 완료** 창이 표시됩니다.

17. **완료**를 클릭하여 리소스 키트 설치 프로그램을 종료합니다.

제 9 장

업그레이드 후 태스크

이 장에 포함된 항목:

- [업그레이드 후 태스크, 107](#)
- [Microsoft SQL Server 2017용 JDBC 드라이버 구성, 108](#)
- [속성 업데이트, 108](#)
- [JBoss 업그레이드 후 태스크, 109](#)
- [인플레이스 업그레이드의 업그레이드 후 태스크 수행, 109](#)
- [개체, 열 및 사용되지 않는 개체에 대한 참조 삭제, 109](#)
- [Hub 서버 배포를 위한 PostInstall 스크립트 실행\(조건부\), 110](#)
- [Hub 콘솔 클라이언트 구성, 111](#)
- [WebSphere 관리 보안 구성, 111](#)
- [클래스 로더 구성 WebSphere에서, 118](#)
- [연산 참조 저장소 등록, 119](#)
- [업그레이드된 메타데이터 유효성 검사, 123](#)
- [클린 업그레이드의 업그레이드 후 태스크 수행, 125](#)
- [플랫폼 변환을 위한 정리 함수 구성, 127](#)
- [MDM Hub 환경 보고서 검토, 128](#)
- [외부 호출 및 응용 프로그램 업그레이드, 129](#)
- [EJB 프로토콜용 SiperianClient 라이브러리 클래스 업그레이드, 130](#)
- [MDM Hub 메타데이터 준비, 130](#)
- [업그레이드 테스트, 131](#)
- [일반 Hub 서버 속성 구성, 132](#)
- [Data Director 및 Hub 서버 속성, 133](#)
- [Data Director 글로벌 속성, 134](#)
- [비즈니스 항목 스키마 생성, 134](#)

업그레이드 후 태스크

클린 업그레이드 또는 인플레이스 업그레이드를 수행하는 경우 업그레이드 후 태스크를 수행하여 환경이 올바르게 구성되었는지 확인해야 합니다.

Microsoft SQL Server 2017용 JDBC 드라이버 구성

Microsoft SQL Server 2017을 사용하는 환경에 Hub 서버를 설치한 경우 올바른 버전의 JDBC 드라이버 파일을 Hub 서버 lib 디렉터리에 복사합니다.

1. Microsoft SQL Server가 설치되어 있는 시스템의 Binn 디렉터리에서 Microsoft JDBC 드라이버의 지원되는 최신 버전을 찾습니다.
설치 전 태스크를 수행할 때 드라이버 파일을 Microsoft 웹 사이트에서 Binn 디렉터리로 다운로드 및 복사합니다.
2. Microsoft JDBC JAR 파일을 다음 디렉터리에 복사합니다.
<MDM Hub 설치 디렉터리>/hub/server/lib
3. 응용 프로그램 서버를 다시 시작합니다.

속성 업데이트

업그레이드 태스크	세부 정보
속성 파일에서 응용 프로그램 서버 설정 업데이트	<p>응용 프로그램 서버를 업그레이드할 경우 속성 파일에서 응용 프로그램 서버 설정을 수동으로 업데이트해야 합니다.</p> <p><MDM Hub 설치 디렉터리>\hub\server\bin에 있는 build.properties 파일에서 다음 설정을 업데이트합니다.</p> <ul style="list-style-type: none"> - SIP.AS.HOME - SIP.AS.SERVER_FOLDER - SIP.AS.DEPLOY_FOLDER <p><MDM Hub 설치 디렉터리>\hub\server에 있는 setSiperianEnv.bat 파일에서 다음 설정을 업데이트합니다.</p> <ul style="list-style-type: none"> - SET JBS_HOME - SET JBS_SERVER_DIR - SET JBS_DEPLOY_DIR - SET JBS_CLIENT_CLASSPATH <p><MDM Hub 설치 디렉터리>\hub\server\resources에 있는 cmxserver.properties 파일에서 다음 설정을 업데이트합니다.</p> <ul style="list-style-type: none"> - cmx.appserver.version <p>다른 응용 프로그램 서버 구성 설정(예: 포트 번호)을 변경한 경우 cmxserver.properties에서도 설정을 업데이트해야 합니다.</p>

JBoss 업그레이드 후 태스크

JBoss 환경에서 다음 태스크를 수행합니다.

태스크	작업
odjbc6.jar 파일 제거	다음 위치에서 JAR 파일을 제거합니다. <code><JBoss install location>\modules\com\activevos\main</code> <code><JBoss install location>\modules\com\informatica\mdm\jdbc\main</code> <code><MDM Hub 설치 디렉터리>\hub\server\lib</code>
필요한 경우 JDBC 드라이버를 sqljdbc42.jar로 업데이트합니다.	1. JDBC 드라이버를 코어 모듈로 설치합니다. 지침은 JBoss 설명서를 참조하십시오. 2. JBoss standalone-full.xml 파일에서 datasource 정의를 업데이트하여 모듈을 사용하도록 합니다.

인플레이스 업그레이드의 업그레이드 후 태스크 수행

인플레이스 업그레이드 후 몇 가지 업그레이드 후 태스크를 수행해야 합니다.

다음 테이블은 업그레이드 후 태스크에 대한 세부 정보를 제공합니다.

업그레이드 태스크	세부 정보
Java 캐시를 지우기	1. Java 캐시를 지웁니다. 자세한 내용은 Java 설명서를 참조하십시오. 2. Hub 콘솔을 실행합니다.
로깅 구성	Data Director의 성능을 최적화하려면 log4j.xml 파일을 편집하여 로깅을 구성합니다. 1. 다음 디렉터리에서 log4j.xml을 엽니다. <code><MDM Hub 설치 디렉터리>/hub/server/conf</code> 2. 다음 로깅 구성을 추가합니다. <code><category name="org.eclipse.persistence.sdo"> <priority value="WARN"/> </category></code> <code><category name="org.eclipse.persistence.default"> <priority value="WARN"/> </category></code> 3. log4j.xml 파일을 저장하고 닫습니다.

개체, 열 및 사용되지 않는 개체에 대한 참조 삭제

업그레이드 태스크	세부 정보
REL_START_DATE, REL_END_DATE 및 HUID 테이블에 대한 참조 업데이트	REL_START_DATE 및 REL_END_DATE 시스템 열과 HUID 테이블에 대한 참조를 업데이트합니다. 업그레이드 프로세스에서는 계층 관리자 기본 개체에서 사용된 패키지 및 보기에서 REL_START_DATE 및 REL_END_DATE 열에 대한 참조를 제거합니다. REL_START_DATE의 참조를 PERIOD_START_DATE로 변경합니다. REL_END_DATE의 참조를 PERIOD_END_DATE로 변경합니다.

Hub 서버 배포를 위한 PostInstall 스크립트 실행(조건부)

설치 중에 `postInstallSetup` 스크립트를 건너뛸 경우 스크립트를 실행합니다. 설치 후 프로세스에서는 Hub 서버 응용 프로그램을 배포하고, 데이터 소스를 생성하고, JMS 메시지 대기열을 구성합니다.

WebLogic 서버 환경에서 관리 서버와 관리되는 서버가 다른 시스템에 있는 상태의 관리되는 서버를 사용하는 경우 모든 배포 파일을 관리 서버의 MDM Hub 설치 디렉터리에 복사합니다. 자세한 내용은 *Multidomain MDM 설치 가이드*를 참조하십시오.

- ▶ 명령 프롬프트를 열고 다음 디렉터리에서 `postInstallSetup` 스크립트를 실행합니다. <MDM Hub 설치 디렉터리>/hub/server

참고: MDM Hub 설치 프로그램과 함께 번들로 제공되는 ActiveVOS 버전을 설치하지 않은 경우 ActiveVOS 사용자 이름과 암호를 명령에 포함시키지 마십시오. UNIX의 경우 암호에 느낌표(!) 문자를 넣는 경우 느낌표 앞에 백슬래시를 포함시켜야 합니다. 예를 들어, 암호가 `!!cmx!!`, 다음 암호를 입력하십시오. `\!\\!cmx\\!\\!`

보안이 비활성화된 WebSphere의 경우:

```
./postInstallSetup.sh -Ddatabase.password=<MDM Hub Master database password>
-Davos.username=<ActiveVOS Console username>
-Davos.password=<ActiveVOS Console password>
-Davos.jdbc.database.username=<ActiveVOS database username>
-Davos.jdbc.database.password=<ActiveVOS database password>
```

보안이 활성화된 WebSphere의 경우:

```
./postInstallSetup.sh -Dwebsphere.password=<WebSphere password>
-Ddatabase.password=<MDM Hub Master database password>
-Davos.username=<ActiveVOS Console username>
-Davos.password=<ActiveVOS Console password>
-Davos.jdbc.database.username=<ActiveVOS database username>
-Davos.jdbc.database.password=<ActiveVOS database password>
```

JBoss의 경우

```
./postInstallSetup.sh -Ddatabase.password=<MDM Hub Master database password>
-Davos.username=<ActiveVOS Console username>
-Davos.password=<ActiveVOS Console password>
-Davos.jdbc.database.username=<ActiveVOS database username>
-Davos.jdbc.database.password=<ActiveVOS database password>
```

WebLogic의 경우:

```
./postInstallSetup.sh -Dweblogic.password=<WebLogic password>
-Ddatabase.password=<MDM Hub Master database password>
-Davos.username=<ActiveVOS Console username>
-Davos.password=<ActiveVOS Console password>
-Davos.jdbc.database.username=<ActiveVOS database username>
-Davos.jdbc.database.password=<ActiveVOS database password>
```

중요: WebLogic 12.2.1.3 이상 환경에서 ActiveVOS를 설치하거나 WebLogic T3S 프로토콜을 사용하기로 결정하는 경우 결정에 해당하는 옵션을 추가합니다.

- ActiveVOS 설치됨. `-Dinstall.avos.patch=true`
- T3S 프로토콜 사용됨. `-Dweblogic.naming.protocol=t3s`

ActiveVOS 콘솔 자격 증명은 응용 프로그램 서버의 관리자 자격 증명과 같습니다.

ActiveVOS 데이터베이스 자격 증명은 `create_bpm` 스크립트를 실행하는 데 사용된 자격 증명과 같습니다.

관리되는 서버가 포함된 WebLogic 환경에서 배포하는 경우 WebLogic 서버 관리 콘솔에서 모든 관리되는 서버를 배포 대상으로 지정해야 합니다.

자세한 내용은 *Multidomain MDM 설치 가이드* 및 WebLogic 설명서를 참조하십시오.

Hub 콘솔 클라이언트 구성

Hub 서버 시스템에 연결되도록 Hub 콘솔 클라이언트를 구성하려면 `cmxserver.properties` 파일을 편집하고 Hub 서버 응용 프로그램을 다시 패키징합니다.

참고: Hub 콘솔을 실행할 때 호스트 이름 및 포트 번호를 재정의할 수 있습니다.

1. 다음 디렉터리의 `cmxserver.properties` 파일에서 `cmx.appserver.console.mode`에 대한 관련 값을 설정합니다.

<MDM Hub 설치 디렉터리>/hub/server/resources
사용하는 통신 프로토콜(HTTP 또는 HTTPS)로 값을 설정합니다.

2. 다음 디렉터리로 이동합니다.

<MDM Hub 설치 디렉터리>/hub/server/bin

3. 다음 명령을 실행합니다.

UNIX의 경우.

```
./sip_ant.sh repack -Dconsole.hostname=<host name> -Dconsole.webport=<port>
```

Windows의 경우.

```
sip_ant.bat repack -Dconsole.hostname=<host name> -Dconsole.webport=<port>
```

여기서 호스트 이름은 응용 프로그램 서버가 바인딩되어 있는 서버의 IP 주소 또는 공개적으로 액세스할 수 있는 호스트 이름이며, 포트는 Hub 콘솔에서 사용해야 하는 현재 노드의 HTTP 또는 HTTPS 포트입니다.

WebSphere 관리 보안 구성

WebSphere 관리 보안을 구성하여 WebSphere 관리 콘솔에 대한 MDM Hub 액세스를 제어할 수 있습니다.

WebSphere 관리 보안을 구성하려면 다음 단계를 수행합니다.

1. ORS(연산 참조 저장소)를 등록 해제합니다.
2. EAR 파일을 제거하고 WebSphere에서 데이터 소스를 제거합니다.
3. WebSphere 관리 콘솔에서 WebSphere 관리 보안을 활성화합니다.
4. Hub 서버 및 처리 서버 속성을 구성합니다.
5. Hub 서버 및 처리 서버 PostInstallSetup 스크립트를 실행합니다.
6. ORS를 등록합니다.

ORS를 등록할 때 WebSphere 자격 증명을 입력합니다. ORS를 확인한 후 자격 증명을 입력할 필요가 없습니다.

연산 참조 저장소 등록 해제

ORS(연산 참조 저장소)를 등록 해제하려면 MDM Hub 콘솔의 데이터베이스 도구를 사용합니다.

1. MDM Hub 콘솔에서 **쓰기 잠금 > 잠금 획득**을 클릭합니다.
2. 구성 작업 영역에서 **데이터베이스** 도구를 선택합니다.
데이터베이스 정보 페이지가 표시됩니다.
3. 데이터베이스 목록에서 등록을 해제할 ORS를 선택합니다.
4. 데이터베이스 등록 해제를 클릭합니다.
데이터베이스 도구에 ORS 등록 해제를 확인하는 메시지가 표시됩니다.

5. 예를 클릭합니다.

EAR 파일 제거 및 데이터 소스 제거

EAR 파일을 제거하고 데이터 소스를 제거하려면 WebSphere 관리 콘솔을 사용합니다.

1. WebSphere 관리 콘솔을 사용하여 다음 배포 파일의 배포를 해제합니다.

배포 파일 이름	설명
siperian-mrm.ear	필수 사항입니다. Hub 서버 응용 프로그램입니다.
provisioning-ear.ear	필수 사항입니다. 프로비저닝 도구 응용 프로그램입니다.
entity360view-ear.ear	선택 사항입니다. Entity 360 프레임워크입니다.

2. WebSphere 관리 콘솔을 사용하여 MDM Hub 마스터 데이터베이스 및 연산 참조 저장소에 대한 모든 데이터 소스를 제거합니다.
3. 응용 프로그램 서버를 다시 시작합니다.

자세한 내용은 WebSphere 설명서를 참조하십시오.

WebSphere 관리 콘솔에서 WebSphere 관리 보안 활성화

WebSphere 관리 콘솔에서 WebSphere 관리 보안을 활성화해야 합니다. WebSphere 관리 보안을 활성화할 때 WebSphere 응용 프로그램 보안을 비활성화합니다.

자세한 내용은 WebSphere 설명서를 참조하십시오.

Hub 서버 및 처리 서버 속성 구성

WebSphere 관리 보안을 활성화하려면 Hub 서버 및 처리 서버 속성 파일을 구성해야 합니다.

1. 응용 프로그램 서버를 중지합니다.
2. Hub 서버에서 WebSphere 보안을 활성화합니다.
 - a. 다음 디렉터리에서 `cmxserver.properties`를 엽니다.
 UNIX의 경우. <MDM Hub 설치 디렉터리>/hub/server/resources
 Windows의 경우. <MDM Hub 설치 디렉터리>\hub\server\resources
 - b. `cmx.websphere.security.enabled`를 true로 설정합니다.
3. 처리 서버에서 WebSphere 보안을 활성화합니다.
 - a. 다음 디렉터리에서 `cmxcleanse.properties`를 엽니다.
 UNIX의 경우. <MDM Hub 설치 디렉터리>/hub/cleanse/resources
 Windows의 경우. <MDM Hub 설치 디렉터리>\hub\cleanse\resources
 - b. `cmx.websphere.security.enabled`를 true로 설정합니다.

4. Hub 서버에서 WebSphere 사용자 이름을 구성합니다.
 - a. 다음 디렉터리에서 build.properties를 엽니다.
 UNIX의 경우. <MDM Hub 설치 디렉터리>/hub/server/bin
 Windows의 경우. <MDM Hub 설치 디렉터리>\hub\server\bin
 - b. websphere.username을 WebSphere 관리자 이름으로 설정합니다.
5. 처리 서버에서 WebSphere 사용자 이름을 구성합니다.
 - a. 다음 디렉터리에서 build.properties를 엽니다.
 UNIX의 경우. <MDM Hub 설치 디렉터리>/hub/cleanse/bin
 Windows의 경우. <MDM Hub 설치 디렉터리>\hub\cleanse\bin
 - b. websphere.username을 WebSphere 관리자 이름으로 설정합니다.
6. SQL* Plus에서 다음 명령을 실행하여 c_repos_cleanse_match_server.is_secured를 1로 설정합니다.

```
UPDATE c_repos_cleanse_match_server set is_secured = 1 where rowid_cleanse_match_server='<Insert value here>';
COMMIT;
```
7. 응용 프로그램 서버를 시작합니다.

Hub 서버 PostInstallSetup 스크립트 수동 실행

Hub 서버 PostInstallSetup 스크립트를 실행해야 합니다.

1. 명령 프롬프트를 엽니다.
2. 다음 디렉터리의 PostInstallSetup 스크립트로 이동합니다.
 UNIX의 경우. <MDM Hub 설치 디렉터리>/hub/server
 Windows의 경우. <MDM Hub 설치 디렉터리>\hub\server
3. 다음 명령을 실행합니다.
 UNIX의 경우. postinstallsetup.sh -Ddatabase.password=<MDM Hub 마스터 데이터베이스> -Dwebsphere.password=<WebSphere administrative user password>
 Windows의 경우. postinstallsetup.bat -Ddatabase.password=<MDM Hub 마스터 데이터베이스> -Dwebsphere.password=<WebSphere administrative user password>
4. 응용 프로그램 서버를 다시 시작합니다.

처리 서버 PostInstallSetup 스크립트 실행

처리 서버 PostInstallSetup 스크립트를 실행해야 합니다.

1. 명령 프롬프트를 엽니다.
2. 다음 디렉터리의 PostInstallSetup 스크립트로 이동합니다.
 UNIX의 경우. <MDM Hub 설치 디렉터리>/hub/cleanse
 Windows의 경우. <MDM Hub 설치 디렉터리>\hub\cleanse
3. 다음 명령을 실행합니다.
 UNIX의 경우. postinstallsetup.sh -Dwebsphere.password=<websphere administrative user password>
 Windows의 경우. postinstallsetup.bat -Dwebsphere.password=<websphere administrative user password>
4. 응용 프로그램 서버를 다시 시작합니다.

연산 참조 저장소 등록

연산 참조 저장소를 등록하려면 Hub 콘솔을 사용합니다.

1. 구성 작업 영역에서 **데이터베이스** 도구를 시작합니다.
2. 쓰기 잠금 > 잠금 획득을 클릭합니다.
3. 데이터베이스 등록을 클릭합니다.

Informatica MDM Hub 연결 마법사가 표시되어 데이터베이스 유형을 선택하라는 메시지를 표시합니다.

4. Microsoft SQL Server, Oracle 또는 IBM Db2를 선택하고 **다음**을 클릭합니다.
5. Microsoft SQL Server에서 데이터베이스에 대한 연결 속성을 구성합니다.
 - a. 연결 속성 페이지에서 연결 속성을 지정한 다음 **다음**을 클릭합니다.

다음 테이블에는 연결 속성이 나열 및 설명되어 있습니다.

속성	설명
데이터베이스 표시 이름	Hub 콘솔에 표시되어야 하는 연산 참조 저장소의 이름입니다.
시스템 식별자	Hub 저장소 인스턴스에서 레코드를 고유하게 식별하기 위해 키에 지정되는 접두사입니다.
데이터베이스 호스트 이름	Microsoft SQL Server 데이터베이스를 호스팅하는 서버의 IP 주소 또는 이름입니다.
포트	Microsoft SQL Server 데이터베이스의 포트입니다. 기본값은 1433입니다.
스키마 이름	연산 참조 저장소의 이름입니다.
암호	연산 참조 저장소의 사용자 이름과 연결된 암호입니다.
Dynamic Data Masking 호스트	Dynamic Data Masking을 호스팅하는 서버의 이름 또는 IP 주소입니다. Dynamic Data Masking을 사용하지 않는 경우 빈 상태로 둡니다.
DDM 연결 URL	선택 사항입니다. Dynamic Data Masking 서버의 URL입니다. Dynamic Data Masking을 사용하지 않는 경우 빈 상태로 둡니다.

- b. 연결 속성 페이지에서 연결 속성을 지정한 다음 **다음**을 클릭합니다.
- c. 요약을 검토하고, 추가 연결 속성을 지정합니다.

다음 테이블에는 구성할 수 있는 추가 연결 속성이 나열되어 있습니다.

속성	설명
연결 URL	연결 URL입니다. 연결 마법사가 기본적으로 연결 URL을 생성합니다.
등록 후 데이터 소스 작성	등록 후 응용 프로그램 서버에 데이터 소스를 작성하려면 선택합니다.

6. Oracle 환경에서 데이터베이스에 대한 연결 속성을 구성합니다.
 - a. Oracle 연결 방법을 선택하고 **다음**을 클릭합니다.

다음 테이블에서는 선택할 수 있는 Oracle 연결 방법을 설명합니다.

연결 방법	설명
서비스	서비스 이름을 사용하여 Oracle에 연결합니다.
SID	Oracle 시스템 ID를 사용하여 Oracle에 연결합니다.

SERVICE 및 SID 이름에 대한 자세한 내용은 Oracle 설명서를 참조하십시오.

연결 속성 페이지가 표시됩니다.

- b. 선택한 연결 유형의 연결 속성을 지정하고 **다음**을 클릭합니다.

다음 테이블에는 연결 속성이 나열 및 설명되어 있습니다.

속성	설명
데이터베이스 표시 이름	Hub 콘솔에 표시되어야 하는 연산 참조 저장소의 이름입니다.
시스템 식별자	Hub 저장소 인스턴스에서 레코드를 고유하게 식별하기 위해 키에 지정되는 접두사입니다.
데이터베이스 호스트 이름	Oracle 데이터베이스를 호스팅하는 서버의 IP 주소 또는 이름입니다.
SID	서버에서 실행 중인 Oracle 데이터베이스의 인스턴스를 나타내는 Oracle 시스템 식별자입니다. SID 필드는 SID 연결 유형을 선택한 경우에 표시됩니다.
서비스	Oracle 데이터베이스에 연결하는 데 사용되는 Oracle 서비스의 이름입니다. 서비스 필드는 서비스 연결 유형을 선택한 경우에 표시됩니다.
포트	Oracle 데이터베이스 서버에서 실행 중인 Oracle 수신기의 TCP 포트입니다. 기본값은 1521입니다.
Oracle TNS 이름	응용 프로그램 서버의 TNSNAMES.ORA 파일에 정의된 대로 네트워크에서 데이터베이스로 알려진 이름입니다. 예를 들면 mydatabase.mycompany.com입니다. Oracle 데이터베이스를 설치할 때 Oracle TNS 이름을 설정합니다. Oracle TNS 이름에 대한 자세한 내용은 Oracle 설명서를 참조하십시오.
스키마 이름	연산 참조 저장소의 이름입니다.
사용자 이름	연산 참조 저장소의 사용자 이름입니다. 기본적으로 이 이름은 연산 참조 저장소를 생성하는 데 사용한 스크립트에서 지정한 사용자 이름입니다. 이 사용자는 Hub 저장소의 모든 연산 참조 저장소 데이터베이스 개체를 소유합니다. 연산 참조 저장소에 프록시 사용자가 구성된 경우 대신 프록시 사용자를 지정할 수 있습니다.

속성	설명
암호	연산 참조 저장소의 사용자 이름과 연결된 암호입니다. Oracle의 경우 암호는 대/소문자를 구분하지 않습니다. 기본적으로 이 암호는 연산 참조 저장소를 생성할 때 지정하는 암호입니다. 연산 참조 저장소에 프록시 사용자가 구성된 경우 대신 프록시 사용자의 암호를 지정해야 합니다.
Dynamic Data Masking 호스트	Dynamic Data Masking을 호스팅하는 서버의 이름 또는 IP 주소입니다. Dynamic Data Masking을 사용하지 않는 경우 빈 상태로 둡니다.
DDM 연결 URL	선택 사항입니다. Dynamic Data Masking 서버의 URL입니다. Dynamic Data Masking을 사용하지 않는 경우 빈 상태로 둡니다.

참고: 스키마 이름과 사용자 이름은 둘 다 연산 참조 저장소를 생성할 때 지정한 연산 참조 저장소의 이름입니다. 이 정보가 필요한 경우 데이터베이스 관리자에게 문의하십시오.

요약 페이지가 표시됩니다.

- c. 요약을 검토하고, 추가 연결 속성을 지정합니다.

다음 테이블에는 구성할 수 있는 추가 연결 속성이 나열되어 있습니다.

속성	설명
연결 URL	연결 URL입니다. 연결 마법사가 기본적으로 연결 URL을 생성합니다. 다음 목록에는 Oracle 연결 유형에 대한 연결 URL 형식이 나와 있습니다. 서비스 연결 유형 <code>jdbc:oracle:thin:@//database_host:port/service_name</code> SID 연결 유형 <code>jdbc:oracle:thin:@//database_host:port:sid</code> 서비스 연결 유형에 한해 다른 연결 URL을 사용자 지정하고 나중에 테스트할 수 있습니다.
등록 후 데이터 소스 작성	등록 후 응용 프로그램 서버에 데이터 소스를 작성하려면 선택합니다. 참고: 해당 옵션을 선택하지 않는 경우에는 데이터 소스를 수동으로 구성해야 합니다.

- d. 서비스 연결 유형의 경우 기본 URL을 변경하려면 **편집** 단추를 클릭하고 URL을 지정한 다음 **확인**을 클릭합니다.

7. IBM Db2 환경에서 데이터베이스에 대한 연결 속성을 구성합니다.

- a. 연결 속성을 지정하고 **다음**을 클릭합니다.

다음 테이블에는 연결 속성이 나열 및 설명되어 있습니다.

속성	설명
데이터베이스 표시 이름	Hub 콘솔에 표시되어야 하는 연산 참조 저장소의 이름입니다.
시스템 식별자	Hub 저장소 인스턴스에서 레코드를 고유하게 식별하기 위해 키에 지정되는 접두사입니다.
데이터베이스 서버 이름	IBM Db2 데이터베이스를 호스팅하는 서버의 이름 또는 IP 주소입니다.
데이터베이스 이름	생성한 데이터베이스의 이름입니다.
데이터베이스 호스트 이름	IBM Db2 데이터베이스를 호스팅하는 서버의 이름 또는 IP 주소입니다.
스키마 이름	연산 참조 저장소의 이름입니다.
사용자 이름	연산 참조 저장소의 사용자 이름입니다. 기본적으로 이 이름은 연산 참조 저장소를 생성하는 데 사용한 스크립트에서 지정한 사용자 이름입니다. 이 사용자는 Hub 저장소의 모든 연산 참조 저장소 데이터베이스 개체를 소유합니다. 연산 참조 저장소에 프록시 사용자가 구성된 경우 대신 프록시 사용자를 지정할 수 있습니다.
암호	연산 참조 저장소의 사용자 이름과 연결된 암호입니다. IBM Db2의 경우 암호는 대/소문자를 구분합니다. 기본적으로 이 암호는 연산 참조 저장소를 생성할 때 지정하는 암호입니다. 연산 참조 저장소에 프록시 사용자가 구성된 경우 대신 프록시 사용자의 암호를 지정해야 합니다.
Dynamic Data Masking 호스트	Dynamic Data Masking을 호스팅하는 서버의 이름 또는 IP 주소입니다. Dynamic Data Masking을 사용하지 않는 경우 빈 상태로 둡니다.
DDM 연결 URL	선택 사항입니다. Dynamic Data Masking 서버의 URL입니다. Dynamic Data Masking을 사용하지 않는 경우 빈 상태로 둡니다.

참고: 스키마 이름과 사용자 이름은 둘 다 연산 참조 저장소를 생성할 때 지정한 연산 참조 저장소의 이름입니다. 이 정보가 필요한 경우 데이터베이스 관리자에게 문의하십시오.

요약 페이지가 표시됩니다.

- b. 요약을 검토하고, 추가 연결 속성을 지정합니다.

다음 테이블에는 구성할 수 있는 추가 연결 속성이 나열되어 있습니다.

속성	설명
연결 URL	연결 URL입니다. 연결 마법사가 기본적으로 연결 URL을 생성합니다. 다음 예에서는 연결 URL의 형식을 보여 줍니다. <code>jdbc:db2://database_host:port/db_name</code>
등록 후 데이터 소스 작성	등록 후 응용 프로그램 서버에 데이터 소스를 작성하려면 선택합니다. 참고: 해당 옵션을 선택하지 않는 경우에는 데이터 소스를 수동으로 구성해야 합니다.

8. **마침**을 클릭합니다.
데이터베이스 등록 대화 상자가 표시됩니다.
9. **확인**을 클릭합니다.
응용 프로그램 서버 로그인 대화 상자가 표시됩니다.
10. WebSphere 관리자 이름과 암호를 입력합니다.
11. **확인**을 클릭합니다.
MDM Hub이 ORS를 등록합니다.
12. 응용 프로그램 서버를 다시 시작합니다.
13. 등록한 연산 참조 저장소를 선택하고 **데이터베이스 연결 테스트** 단추를 클릭하여 데이터베이스 설정을 테스트합니다.
데이터베이스 테스트 대화 상자에 데이터베이스 연결 테스트 결과가 표시됩니다.
14. **확인**을 클릭합니다.
ORS가 등록되고 데이터베이스에 대한 연결이 테스트됩니다.

클래스 로더 구성 WebSphere에서

필수 PostInstallSetup 스크립트 중 하나를 실행한 후 WebSphere 배포 관리자를 사용하여 Hub 서버 및 처리 서버 응용 프로그램의 클래스 로더를 구성합니다.

1. siperian-mrm.ear, provisioning-ear.ear, entity360view-ear.ear 및 siperian-mrm-cleanse.ear 응용 프로그램의 클래스 로더를 구성합니다.
 - a. **응용 프로그램 > 응용 프로그램 유형 > WebSphere 엔터프라이즈 응용 프로그램**을 선택합니다.
 - b. **엔터프라이즈 응용 프로그램** 페이지에서 응용 프로그램 중 하나를 클릭합니다.
 - c. 응용 프로그램을 구성하는 페이지에서 **클래스 로드 및 업데이트 검색** 링크를 클릭합니다.
 - d. **클래스 로더** 구성 페이지에서 **로컬 클래스 로더**를 사용하여 로드된 클래스 먼저(상위 클래스는 마지막에) 클래스 로더 순서 옵션을 선택합니다.
 - e. **적용**을 클릭한 다음 **확인**을 클릭합니다.
2. 다음 응용 프로그램 EAR 파일의 웹 모듈에 대한 클래스 로더를 구성합니다.

응용 프로그램 EAR 파일	웹 모듈	클래스 로더 순서
siperian-mrm.ear	zds-gui.war	로컬 클래스 로더를 사용하여 로드된 클래스 먼저(상위 클래스는 마지막에)
provisioning-ear.ear	provisioning.war	로컬 클래스 로더를 사용하여 로드된 클래스 먼저(상위 클래스는 마지막에)
siperian-mrm-cleanse.ear	siperian-mrm-cleanse.war	로컬 클래스 로더를 사용하여 로드된 클래스 먼저(상위 클래스는 마지막에)

- a. **응용 프로그램 > 응용 프로그램 유형 > WebSphere 엔터프라이즈 응용 프로그램**을 선택합니다.

- b. **엔터프라이즈 응용 프로그램** 페이지에서 응용 프로그램 **EAR** 파일의 이름을 클릭합니다.
 - c. 응용 프로그램을 구성하는 페이지에서 **모듈 관리** 링크를 클릭합니다.
 - d. 모듈 목록에서 웹 모듈의 링크를 클릭합니다.
 - e. 웹 모듈 구성 페이지에서 클래스 로더 순서를 선택합니다.
 - f. **적용**을 클릭한 다음 **확인**을 클릭합니다.
3. WebSphere를 다시 시작한 후 **Hub** 서버 및 처리 서버 응용 프로그램을 시작합니다.

연산 참조 저장소 등록

연산 참조 저장소를 등록하려면 **Hub** 콘솔을 사용합니다.

1. 구성 작업 영역에서 **데이터베이스** 도구를 시작합니다.
2. 쓰기 잠금 > 잠금 획득을 클릭합니다.
3. 데이터베이스 등록을 클릭합니다.

Informatica MDM Hub 연결 마법사가 표시되어 데이터베이스 유형을 선택하라는 메시지를 표시합니다.

4. Microsoft SQL Server, Oracle 또는 IBM Db2를 선택하고 **다음**을 클릭합니다.
5. Microsoft SQL Server에서 데이터베이스에 대한 연결 속성을 구성합니다.
 - a. 연결 속성 페이지에서 연결 속성을 지정한 다음 **다음**을 클릭합니다.

다음 테이블에는 연결 속성이 나열 및 설명되어 있습니다.

속성	설명
데이터베이스 표시 이름	Hub 콘솔에 표시되어야 하는 연산 참조 저장소의 이름입니다.
시스템 식별자	Hub 저장소 인스턴스에서 레코드를 고유하게 식별하기 위해 키에 지정되는 접두사입니다.
데이터베이스 호스트 이름	Microsoft SQL Server 데이터베이스를 호스팅하는 서버의 IP 주소 또는 이름입니다.
포트	Microsoft SQL Server 데이터베이스의 포트입니다. 기본값은 1433입니다.
스키마 이름	연산 참조 저장소의 이름입니다.
암호	연산 참조 저장소의 사용자 이름과 연결된 암호입니다.
Dynamic Data Masking 호스트	Dynamic Data Masking을 호스팅하는 서버의 이름 또는 IP 주소입니다. Dynamic Data Masking을 사용하지 않는 경우 빈 상태로 둡니다.
DDM 연결 URL	선택 사항입니다. Dynamic Data Masking 서버의 URL입니다. Dynamic Data Masking을 사용하지 않는 경우 빈 상태로 둡니다.

- b. 연결 속성 페이지에서 연결 속성을 지정한 다음 **다음**을 클릭합니다.

- c. 요약을 검토하고, 추가 연결 속성을 지정합니다.
다음 테이블에는 구성할 수 있는 추가 연결 속성이 나열되어 있습니다.

속성	설명
연결 URL	연결 URL입니다. 연결 마법사가 기본적으로 연결 URL을 생성합니다.
등록 후 데이터 소스 작성	등록 후 응용 프로그램 서버에 데이터 소스를 작성하려면 선택합니다.

6. Oracle 환경에서 데이터베이스에 대한 연결 속성을 구성합니다.

- a. Oracle 연결 방법을 선택하고 **다음**을 클릭합니다.
다음 테이블에서는 선택할 수 있는 Oracle 연결 방법을 설명합니다.

연결 방법	설명
서비스	서비스 이름을 사용하여 Oracle에 연결합니다.
SID	Oracle 시스템 ID를 사용하여 Oracle에 연결합니다.

SERVICE 및 SID 이름에 대한 자세한 내용은 Oracle 설명서를 참조하십시오.

연결 속성 페이지가 표시됩니다.

- b. 선택한 연결 유형의 연결 속성을 지정하고 **다음**을 클릭합니다.
다음 테이블에는 연결 속성이 나열 및 설명되어 있습니다.

속성	설명
데이터베이스 표시 이름	Hub 콘솔에 표시되어야 하는 연산 참조 저장소의 이름입니다.
시스템 식별자	Hub 저장소 인스턴스에서 레코드를 고유하게 식별하기 위해 키에 지정되는 접두사입니다.
데이터베이스 호스트 이름	Oracle 데이터베이스를 호스팅하는 서버의 IP 주소 또는 이름입니다.
SID	서버에서 실행 중인 Oracle 데이터베이스의 인스턴스를 나타내는 Oracle 시스템 식별자입니다. SID 필드는 SID 연결 유형을 선택한 경우에 표시됩니다.
서비스	Oracle 데이터베이스에 연결하는 데 사용되는 Oracle 서비스의 이름입니다. 서비스 필드는 서비스 연결 유형을 선택한 경우에 표시됩니다.
포트	Oracle 데이터베이스 서버에서 실행 중인 Oracle 수신기의 TCP 포트입니다. 기본값은 1521입니다.

속성	설명
Oracle TNS 이름	응용 프로그램 서버의 TNSNAMES.ORA 파일에 정의된 대로 네트워크에서 데이터베이스로 알려진 이름입니다. 예를 들면 mydatabase.mycompany.com입니다. Oracle 데이터베이스를 설치할 때 Oracle TNS 이름을 설정합니다. Oracle TNS 이름에 대한 자세한 내용은 Oracle 설명서를 참조하십시오.
스키마 이름	연산 참조 저장소의 이름입니다.
사용자 이름	연산 참조 저장소의 사용자 이름입니다. 기본적으로 이 이름은 연산 참조 저장소를 생성하는 데 사용한 스크립트에서 지정한 사용자 이름입니다. 이 사용자는 Hub 저장소의 모든 연산 참조 저장소 데이터베이스 개체를 소유합니다. 연산 참조 저장소에 프록시 사용자가 구성된 경우 대신 프록시 사용자를 지정할 수 있습니다.
암호	연산 참조 저장소의 사용자 이름과 연결된 암호입니다. Oracle의 경우 암호는 대/소문자를 구분하지 않습니다. 기본적으로 이 암호는 연산 참조 저장소를 생성할 때 지정하는 암호입니다. 연산 참조 저장소에 프록시 사용자가 구성된 경우 대신 프록시 사용자의 암호를 지정해야 합니다.
Dynamic Data Masking 호스트	Dynamic Data Masking을 호스팅하는 서버의 이름 또는 IP 주소입니다. Dynamic Data Masking을 사용하지 않는 경우 빈 상태로 둡니다.
DDM 연결 URL	선택 사항입니다. Dynamic Data Masking 서버의 URL입니다. Dynamic Data Masking을 사용하지 않는 경우 빈 상태로 둡니다.

참고: 스키마 이름과 사용자 이름은 둘 다 연산 참조 저장소를 생성할 때 지정한 연산 참조 저장소의 이름입니다. 이 정보가 필요한 경우 데이터베이스 관리자에게 문의하십시오.

요약 페이지가 표시됩니다.

- c. 요약을 검토하고, 추가 연결 속성을 지정합니다.

다음 테이블에는 구성할 수 있는 추가 연결 속성이 나열되어 있습니다.

속성	설명
연결 URL	연결 URL입니다. 연결 마법사가 기본적으로 연결 URL을 생성합니다. 다음 목록에는 Oracle 연결 유형에 대한 연결 URL 형식이 나와 있습니다. 서비스 연결 유형 jdbc:oracle:thin:@//database_host:port/service_name SID 연결 유형 jdbc:oracle:thin:@//database_host:port:sid 서비스 연결 유형에 한해 다른 연결 URL을 사용자 지정하고 나중에 테스트할 수 있습니다.
등록 후 데이터 소스 작성	등록 후 응용 프로그램 서버에 데이터 소스를 작성하려면 선택합니다. 참고: 해당 옵션을 선택하지 않는 경우에는 데이터 소스를 수동으로 구성해야 합니다.

- d. 서비스 연결 유형의 경우 기본 URL을 변경하려면 **편집** 단추를 클릭하고 URL을 지정한 다음 **확인**을 클릭합니다.

7. IBM Db2 환경에서 데이터베이스에 대한 연결 속성을 구성합니다.

- a. 연결 속성을 지정하고 **다음**을 클릭합니다.

다음 테이블에는 연결 속성이 나열 및 설명되어 있습니다.

속성	설명
데이터베이스 표시 이름	Hub 콘솔에 표시되어야 하는 연산 참조 저장소의 이름입니다.
시스템 식별자	Hub 저장소 인스턴스에서 레코드를 고유하게 식별하기 위해 키에 지정되는 접두사입니다.
데이터베이스 서버 이름	IBM Db2 데이터베이스를 호스팅하는 서버의 이름 또는 IP 주소입니다.
데이터베이스 이름	생성한 데이터베이스의 이름입니다.
데이터베이스 호스트 이름	IBM Db2 데이터베이스를 호스팅하는 서버의 이름 또는 IP 주소입니다.
스키마 이름	연산 참조 저장소의 이름입니다.
사용자 이름	연산 참조 저장소의 사용자 이름입니다. 기본적으로 이 이름은 연산 참조 저장소를 생성하는 데 사용한 스크립트에서 지정한 사용자 이름입니다. 이 사용자는 Hub 저장소의 모든 연산 참조 저장소 데이터베이스 개체를 소유합니다. 연산 참조 저장소에 프록시 사용자가 구성된 경우 대신 프록시 사용자를 지정할 수 있습니다.
암호	연산 참조 저장소의 사용자 이름과 연결된 암호입니다. IBM Db2의 경우 암호는 대/소문자를 구분합니다. 기본적으로 이 암호는 연산 참조 저장소를 생성할 때 지정하는 암호입니다. 연산 참조 저장소에 프록시 사용자가 구성된 경우 대신 프록시 사용자의 암호를 지정해야 합니다.
Dynamic Data Masking 호스트	Dynamic Data Masking을 호스팅하는 서버의 이름 또는 IP 주소입니다. Dynamic Data Masking을 사용하지 않는 경우 빈 상태로 둡니다.
DDM 연결 URL	선택 사항입니다. Dynamic Data Masking 서버의 URL입니다. Dynamic Data Masking을 사용하지 않는 경우 빈 상태로 둡니다.

참고: 스키마 이름과 사용자 이름은 둘 다 연산 참조 저장소를 생성할 때 지정한 연산 참조 저장소의 이름입니다. 이 정보가 필요한 경우 데이터베이스 관리자에게 문의하십시오.

요약 페이지가 표시됩니다.

- b. 요약을 검토하고, 추가 연결 속성을 지정합니다.

다음 테이블에는 구성할 수 있는 추가 연결 속성이 나열되어 있습니다.

속성	설명
연결 URL	연결 URL입니다. 연결 마법사가 기본적으로 연결 URL을 생성합니다. 다음 예에서는 연결 URL의 형식을 보여 줍니다. <code>jdbc:db2://database_host:port/db_name</code>
등록 후 데이터 소스 작성	등록 후 응용 프로그램 서버에 데이터 소스를 작성하려면 선택합니다. 참고: 해당 옵션을 선택하지 않는 경우에는 데이터 소스를 수동으로 구성해야 합니다.

8. **마침**을 클릭합니다.
데이터베이스 등록 대화 상자가 표시됩니다.
9. **확인**을 클릭합니다.
응용 프로그램 서버 로그인 대화 상자가 표시됩니다.
10. WebSphere 관리자 이름과 암호를 입력합니다.
11. **확인**을 클릭합니다.
MDM Hub이 ORS를 등록합니다.
12. 응용 프로그램 서버를 다시 시작합니다.
13. 등록한 연산 참조 저장소를 선택하고 **데이터베이스 연결 테스트** 단추를 클릭하여 데이터베이스 설정을 테스트합니다.
데이터베이스 테스트 대화 상자에 데이터베이스 연결 테스트 결과가 표시됩니다.
14. **확인**을 클릭합니다.
ORS가 등록되고 데이터베이스에 대한 연결이 테스트됩니다.

업그레이드된 메타데이터 유효성 검사

ORS(연산 참조 저장소)에 유효성 검사 오류가 하나도 없는지 확인합니다. 이 결과를 업그레이드하기 전의 이전 유효성 검사 결과와 비교합니다. **Hub** 콘솔에서 리포지토리 관리자를 사용하여 메타데이터의 유효성을 검사합니다.

참고: MDM Hub 이전 버전에서 업그레이드를 하고 나면 매핑 없이 준비 테이블이 있던 이전 데이터베이스에서 유효성 검사 오류가 발생할 수도 있습니다. **Hub** 콘솔의 리포지토리 관리자 도구에서 **복구** 단추를 클릭하여 이와 같이 복구 가능한 문제를 수정합니다.

메타데이터 유효성 검사

ORS(연산 참조 저장소) 리포지토리 메타데이터의 유효성을 검사하려면 **Hub** 콘솔의 리포지토리 관리자 도구를 사용하십시오.

1. **Hub** 콘솔 작업 영역의 구성에서 **리포지토리 관리자** 도구를 선택하십시오.
2. **리포지토리 관리자** 도구에서 **유효성 검사** 탭을 선택하십시오.
3. **유효성 검사를 수행할 리포지토리 선택** 목록에서 리포지토리를 선택하십시오.
4. **유효성 검사** 단추를 클릭합니다.

5. **유효성 검사 확인 선택** 대화 상자에서 수행할 유효성 검사 확인을 선택합니다. **확인**을 클릭합니다.
리포지토리 관리자 도구가 리포지토리의 유효성을 검사하고 문제가 있으면 **발견된 문제** 창에 모두 표시합니다.
6. **복구** 단추를 눌러서 복구 가능한 문제를 해결하십시오.
7. **ORS가 알 수 없음** 상태로 남아 있으면 응용 프로그램 서버와 데이터베이스 시스템의 시스템 시계를 동기화하십시오.

유효성 검사 결과 저장

유효성 검사를 실행한 다음, 유효성 검사 결과를 HTML 파일로 저장할 수 있습니다.

1. **Hub** 콘솔에 있는 **리포지토리 관리자** 도구에서 **유효성 검사** 탭을 선택합니다.
2. **저장** 단추를 클릭합니다.
3. **저장** 대화 상자에서 유효성 검사 결과를 저장하고자 하는 디렉터리로 이동합니다.
4. HTML 파일에 대한 설명 파일 이름을 지정합니다. **저장**을 클릭합니다.
리포지토리 관리자(가) 유효성 검사 결과를 지정된 위치에 HTML 파일로 저장합니다.

메타데이터 유효성 검사 메시지 확인

유효성 검사 도구를 실행한 후에 유효성 검사 메시지를 수신할 수도 있습니다.

다음은 가장 일반적인 유효성 검사 오류 메시지입니다.

경고: SIP-PV-10703 패키지 'EMPLOYEE_DETAILS_PKG'가 데이터베이스 뷰와 동기화되어 있지 않습니다.

데이터베이스 뷰와 동기화하려면 리포지토리 관리자에서 복구 프로세스를 실행합니다.

'C_EMPLOYEE_DETAILS_MTI' 보기 - 프록시 사용자 역할에 대해 SELECT 사용 권한이 부여되지 않았습니다. 또는 SIP-MV-11410- 루트 MTIP의 SQL이 올바르지 않습니다.

MTIP 보기를 재작성합니다.

1. Hub 콘솔에서 구성 작업 영역을 열고 **엔터프라이즈 관리자**를 클릭합니다.
2. 쓰기 잠금을 획득합니다.
3. **ORS 데이터베이스** 탭을 선택합니다.
4. 데이터베이스를 선택합니다.
5. **속성** 탭을 선택합니다.
6. **MTIP 재생성 필요**라는 속성을 찾고 **MTIP 재생성** 단추를 클릭합니다.

프록시 사용자 역할에 'EMPLOYEE_DETAILS_PKG' - SELECT 보기 권한이 부여되지 않았습니다.

이 권한이 추가되도록 프록시 사용자 역할을 업데이트합니다.

1. Hub 콘솔에서 보안 액세스 관리자 작업 영역을 연 후 **역할**을 클릭합니다.
2. 쓰기 잠금을 획득합니다.
3. 프록시 사용자 역할을 선택합니다.
4. **리소스 권한** 탭을 선택합니다.
5. 메시지에 나온 이름으로 된 패키지 또는 테이블을 찾습니다.
6. **읽기** 확인란을 선택합니다.

프록시 사용자 역할에 SIP-PV-11105 - SELECT 권한에 보기가 부여되지 않았습니다.

프록시 사용자 역할에 'C_REPOS_USER_GROUP_ALL' - SELECT 보기 권한이 부여되지 않았습니다.

데이터베이스 마이그레이션 스크립트에서 프록시 사용자 역할을 생성했지만 프록시 사용자에게 리포지토리 보기에 대한 권한을 부여하지 않았습니다. 데이터베이스에서 프록시 사용자에게 리포지토리 보기에 대한 SELECT 권한을 부여합니다.

지역화된 스키마 업데이트

업그레이드 전 스키마에 지역화된 조회 테이블이 포함되는 경우 업그레이드 후에 유효성 검사 오류가 표시될 수 있습니다.

버전 10.2.x에서 지역화된 조회 테이블의 경우 메타데이터 유효성 검사 프로세스에서 다음 유형의 오류가 생성됩니다.

SIP-MV-22000 The name attribute for DB bundle [<lookup table name>.dbBundleMapping] is not defined in the configuration

이 오류를 수정하려면 다음 단계를 수행합니다.

1. C_REPOS_CO_CS_CONFIG 리포지토리 테이블에서 dbBundleConfig.xml 파일을 다운로드합니다.
2. dbBundleConfig.xml 파일에서 bundle 요소의 name=<lookup table name> 매개 변수를 mapping 요소에 복사합니다. 예를 들어 다음은 편집된 mapping 요소를 보여줍니다.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<dbBundleConfiguration xmlns="http://www.example.com/mdm/db-bundle-configuration">
<bundle name="LUCountry" hubObject="C_LU_COUNTRY_LCL">
<mapping name="LUCountry" keyColumn="COUNTRY_CD" countryColumn="COUNTRY_CODE"
languageColumn="LANGUAGE_CODE" valueColumn="LOCALIZED_STRING"/>
</bundle>
<bundle name="LUState" hubObject="C_LU_STATE_LCL">
<mapping name="LUState" keyColumn="STATE_CD" countryColumn="COUNTRY_CODE"
languageColumn="LANGUAGE_CODE" valueColumn="LOCALIZED_STRING"/>
</bundle>
<bundle name="LUCountry.LUState" hubObject="C_LU_STATE_LCL">
<mapping name="LUCountry.LUState" keyColumn="STATE_CD" countryColumn="COUNTRY_CODE"
languageColumn="LANGUAGE_CODE" valueColumn="LOCALIZED_STRING"/>
</bundle>
</dbBundleConfiguration>
```

3. 편집된 dbBundleConfig.xml 파일을 C_REPOS_CO_CS_CONFIG 리포지토리 테이블에 업로드합니다.
4. 메타데이터 유효성 검사를 실행합니다.

클린 업그레이드의 업그레이드 후 태스크 수행

클린 업그레이드 후 Hub 콘솔을 시작할 때 암호 해독 오류가 발생하면 몇 가지 업그레이드 후 태스크를 수행해야 합니다.

다음 업그레이드 후 태스크를 수행합니다.

1. 스키마 암호를 암호화합니다.
2. 스키마 암호를 업데이트합니다.
3. 연산 참조 저장소 연결을 테스트하고 업데이트합니다.
4. ActiveVOS 연결을 테스트하고 업데이트합니다.
5. 정리 함수를 테스트하고 처리 서버를 추가합니다.

스키마 암호 암호화

암호화를 통해 데이터베이스 스키마 암호를 보호합니다.

- ▶ 데이터베이스 스키마 암호를 암호화하려면 명령 프롬프트에서 다음 명령을 실행합니다.

```
java -classpath siperian-api.jar;siperian-common.jar;siperian-server.jar
com.delos.util.PublicKeyBasedEncryptionHelper <plain text password> <Hub Server installation
directory>
```

결과가 터미널 창에 출력됩니다.

```
Plaintext Password: password
Encrypted Password: encrypted password
```

스키마 암호 업데이트

MDM Hub 마스터 데이터베이스 암호 또는 연산 참조 저장소 암호를 업데이트할 수 있습니다.

1. 마스터 데이터베이스 암호 또는 연산 참조 저장소 암호를 업데이트하려면 **cmx_system** 사용자로 연결한 후 다음 문을 실행합니다.

Oracle 및 IBM Db2의 경우.

```
UPDATE C_REPOS_DATABASE SET PASSWORD = '<new_password>' WHERE USER_NAME = <user_name>;
COMMIT;
```

Microsoft SQL Server의 경우.

```
UPDATE [dbo].[C_REPOS_DATABASE] SET PASSWORD = '<new_password>' WHERE USER_NAME = <user_name>
```

2. 응용 프로그램 서버를 다시 시작합니다.

연산 참조 저장소 연결 테스트 및 업데이트

연산 참조 저장소에 대한 연결을 테스트합니다. 연결이 실패하면 Hub 콘솔에서 연산 참조 저장소 암호를 편집합니다.

1. 구성 작업 영역에서 **데이터베이스**를 클릭합니다.
2. 쓰기 잠금을 획득합니다.
3. 테스트할 연산 참조 저장소를 선택합니다.
4. **데이터베이스 연결 테스트** 단추를 클릭합니다.
데이터베이스 테스트 대화 상자가 표시됩니다.
5. **확인**을 클릭합니다.
6. 연결 테스트가 실패하면 **데이터베이스 연결 속성 편집** 아이콘을 클릭합니다.
데이터베이스 등록 대화 상자가 선택한 연산 참조 저장소에 대해 표시됩니다.
7. 연산 참조 저장소와 연결된 사용자 이름의 암호를 편집하고 **확인**을 클릭합니다.

ActiveVOS 연결 테스트 및 업데이트

MDM Hub에 대한 ActiveVOS 워크플로우 엔진의 연결을 테스트합니다. 연결이 실패하면 Hub 콘솔에서 ActiveVOS 트러스트된 사용자의 암호를 편집합니다.

1. 구성 작업 영역에서 **워크플로우 관리자**를 클릭합니다.
2. 쓰기 잠금을 획득합니다.
3. **워크플로우 엔진** 탭을 클릭하고 ActiveVOS 워크플로우 엔진을 선택합니다.

4. 워크플로우 엔진 연결을 테스트하려면 **테스트**를 클릭합니다.
5. 워크플로우 엔진 연결 테스트가 실패하면 **편집**을 클릭합니다.
6. 워크플로우 **편집** 대화 상자에서 ActiveVOS 트러스트된 사용자의 암호를 편집하고 **확인**을 클릭합니다.

처리 서버 테스트 및 추가

문자열 함수와 같은 기본 제공 정리 함수를 테스트합니다. 테스트가 암호 해독 오류와 함께 실패하는 경우 Hub 콘솔에서 처리 서버를 삭제 및 추가합니다.

1. 기본 제공 정리 함수를 테스트합니다.
 - a. 모델 작업 영역에서 **정리 함수**를 클릭합니다.
 - b. 쓰기 잠금을 획득합니다.
 - c. 테스트할 정리 함수를 선택합니다.
 - d. **테스트** 탭을 클릭합니다.
 - e. 입력 값을 입력하고 **테스트**를 클릭합니다.
2. 테스트가 실패하면 SIP-09131: 일반 암호 해독 오류의 발생 여부를 cmxserver.log 파일에서 확인합니다. cmxserver.log 파일은 다음 디렉터리에 있습니다.
 <MDM Hub 설치 디렉터리>/hub/server/logs
3. 오류 로그에 SIP-09131 오류가 있는 경우 처리 서버를 삭제 및 추가합니다.
 - a. 유틸리티 작업 영역에서 **처리 서버**를 클릭합니다.
 - b. 쓰기 잠금을 획득합니다.
 - c. 삭제할 처리 서버를 선택하고 **처리 서버 삭제** 아이콘을 클릭합니다.
 - d. 삭제를 확인하고 **확인**을 클릭합니다.
 - e. **처리 서버 추가** 아이콘을 클릭합니다.
 - f. **처리 서버 추가/편집** 대화 상자에서 처리 서버 속성을 구성합니다.
 - g. 구성을 저장합니다.

플랫폼 변환을 위한 정리 함수 구성

직접 구성한 플랫폼 변환을 사용하려면 정리 함수 도구에서 IDQ 라이브러리를 추가합니다. 그런 다음 플랫폼 변환 대신 라이브러리의 정리 함수를 사용할 수 있습니다.

1. Hub 콘솔을 실행하고 **정리 함수** 도구를 시작합니다.
2. 쓰기 잠금을 획득합니다.
3. **정리 함수**를 마우스 오른쪽 단추로 클릭한 다음 **IDQ 라이브러리 추가**를 클릭합니다.
IDQ 라이브러리 추가 대화 상자가 나타납니다.

4. 다음 속성을 지정합니다.

속성	설명
라이브러리 이름	IDQ 라이브러리의 이름입니다. 이 이름은 정리 함수 목록에 폴더 이름으로 표시됩니다.
IDQ WSDL URI	플랫폼 변환과 연결된 WSDL의 URI입니다.
IDQ WSDL 서비스	플랫폼 변환과 연결된 WSDL의 서비스입니다.
IDQ WSDL 포트	플랫폼 변환과 연결된 WSDL의 포트입니다.
설명	정리 함수 도구에 표시할 라이브러리에 대한 설명 텍스트입니다.

5. **확인**을 클릭합니다.

IDQ 라이브러리가 정리 함수 탐색기에 나타납니다.

6. **새로 고침**을 클릭하여 IDQ 라이브러리를 생성합니다.

정리 함수 도구가 플랫폼 변환과 연결된 WSDL을 검색하고, IDQ 라이브러리를 생성하고, 정리 함수 목록에 사용 가능한 정리 함수를 표시합니다.

7. 정리 함수를 테스트합니다.

이제 플랫폼 변환 대신 정리 함수를 사용할 수 있습니다. IDQ 라이브러리의 정리 함수는 플랫폼 변환과 연결된 웹 서비스를 호출할 수 있습니다.

MDM Hub 환경 보고서 검토

Hub 콘솔에서 엔터프라이즈 관리자 도구를 사용하여 Hub 서버, 처리 서버, MDM Hub 마스터 데이터베이스 및 연산 참조 저장소 데이터베이스의 현재 MDM Hub 구성을 검토합니다. 구성 요소의 버전 기록을 참고합니다.

upgradedoc 업그레이드 기록 폴더에 환경 보고서의 사본을 저장합니다.

MDM Hub 환경 보고서 저장

MDM Hub 환경 보고서를 저장하려면 Hub 콘솔에서 엔터프라이즈 관리자 도구를 사용합니다.

1. Hub 콘솔의 **구성** 작업 영역에서 **엔터프라이즈 관리자** 도구를 선택합니다.
2. **엔터프라이즈 관리자** 도구에서 **환경 보고서** 탭을 선택합니다.
3. **저장**을 클릭합니다.
4. **Hub 환경 보고서 저장** 대화 상자에서 환경 보고서를 저장할 디렉터리로 이동합니다.
5. **저장**을 클릭합니다.

외부 호출 및 응용 프로그램 업그레이드

버전 10.4부터 Multidomain MDM은 인증서 기반 인증을 사용하여 외부 호출 및 응용 프로그램을 인증합니다. 외부 호출 및 사용자 지정 응용 프로그램을 사용하려면 트러스트된 응용 프로그램 사용자를 구성해야 합니다. 또한 EJB는 외부 호출에 지원되지 않습니다. 대신 HTTP 통신 프로토콜을 사용해야 합니다.

비즈니스 항목 서비스 및 사용자 지정 응용 프로그램에서 10.4 이전 버전의 리소스 키트에 포함된 BESExternalCall 샘플 코드 및 라이브러리를 사용하는 경우 다음 업그레이드 단계를 수행합니다.

1. 사용자 지정 응용 프로그램에 대한 트러스트된 응용 프로그램 사용자를 구성합니다.
2. bes-client.properties 파일에서 다음 연결 속성을 구성합니다.

연결 속성	설명
siperian-client.protocol	사용하려는 통신 프로토콜입니다. 기본값은 HTTP입니다. 기본값을 변경하지 마십시오.
bes-client.http.url	사용자 지정 응용 프로그램에서 MDM에 연결하기 위한 URL입니다. 다음 구문을 사용합니다. http://<MDM 호스트>:<포트 번호>/cmx 기본값은 http://localhost:8080/cmx입니다.

샘플 bes-client.properties 파일은 다음 디렉터리에 있습니다.

<리소스 키트 설치 디렉터리>/samples/BESExternalCall/source/resources

3. BESExternalCall 샘플의 CustomLogicService.java 파일에서 CustomLogicService 클래스를 검토하고 트러스트된 응용 프로그램 사용자의 사용을 구현합니다.

다음 코드 샘플은 CustomLogicService 클래스를 보여 줍니다.

```
public class CustomLogicService implements Provider<Source> {

    @Override
    public Source invoke(Source request) {

        CompositeServiceClient compositeServiceClient = createCompositeServiceClient();
        CustomLogicFactory customLogicFactory = new CustomLogicFactoryImpl(compositeServiceClient);
        String appName = "<trusted application user>";
        ExternalCallProcessor externalCallProcessor =
            new ExternalCallProcessor(compositeServiceClient, appName, customLogicFactory);

        return externalCallProcessor.invoke(request);
    }

    private static CompositeServiceClient createCompositeServiceClient() {
        InputStream resourceAsStream = CustomLogicService.class.getResourceAsStream("/bes-client.properties");
        Properties config = new Properties();
        try {
            config.load(resourceAsStream);
        } catch (IOException e) {
            throw new RuntimeException(e);
        }
        return CompositeServiceClient.newCompositeServiceClient(config);
    }
}
```

4. 업데이트된 다음 JAR 파일을 사용할 사용자 지정 응용 프로그램을 작성합니다.

- siperian-api.jar
- siperian-common.jar

- mdm-spi.jar

응용 프로그램이 작성되면 외부 호출 및 응용 프로그램에서 HTTP 통신 프로토콜 및 인증서 기반 인증을 사용합니다.

5. 사용자 지정 응용 프로그램을 응용 프로그램 서버에 재배포합니다.

EJB 프로토콜용 SiperianClient 라이브러리 클래스 업그레이드

EJB 프로토콜을 사용하여 SIF(서비스 통합 프레임워크) 요청을 통해 MDM Hub과 통신하는 경우 SiperianClient 라이브러리 클래스 최신 버전을 사용해야 합니다. 사용자 지정 JNDI 조회 방법을 사용하는 경우 이 조회 방법이 EJB3 규칙을 준수하도록 조회 방법을 업데이트합니다.

1. 기존의 SiperianClient 라이브러리 클래스를 SiperianClient 라이브러리 클래스 최신 버전으로 대체합니다.
다음 디렉터리에 있는 siperian-api.jar 파일에 SiperianClient 라이브러리 클래스가 포함되어 있습니다.
 - <리소스 키트 설치 디렉터리>\sdk\sifsdk\lib
 - <MDM Hub 설치 디렉터리>\hub\server\lib
2. 사용자 지정 JNDI 조회 방법을 사용하는 경우 이 조회 방법이 EJB3 규칙을 준수하도록 조회 방법을 업데이트합니다.

MDM Hub 메타데이터 준비

업그레이드 태스크	세부 정보
일치 토큰 다시 생성	각 기본 개체에 대한 일치 토큰 생성 일괄 작업을 실행합니다. 일치 토큰 생성 일괄 작업에서는 처리 서버 업그레이드 시 업데이트하는 SSA-Name3 라이브러리 파일을 기반으로 일치 토큰을 생성합니다.
검색 데이터 다시 인덱싱	검색 데이터에 악센트 부호 문자(예: â 및 ñ)가 포함되어 있으면 처음에 스마트 검색 데이터 인덱싱 일괄 작업을 실행하여 데이터를 다시 인덱싱할 수 있습니다. 데이터를 다시 인덱싱한 후에는 검색 요청이 악센트 부호 문자를 포함하는 레코드를 반환할 수 있습니다.
메타데이터 캐시 구성(선택 사항)	버전 10.1 이하에서 MDM Hub는 메타데이터 캐시에 JBoss 캐시를 사용했습니다. 이러한 버전에서 업그레이드한 후에는 MDM Hub 서버가 JBoss 캐시 구성 파일 대신 Infinispan 구성 파일을 사용합니다. JBoss 캐시와 유사한 결과를 얻으려면 Infinispan 캐시를 구성해야 할 수 있습니다. 자세한 내용은 “메타데이터 캐시 구성(선택 사항)” 페이지 182을 참조하십시오.
사용자 지정 인덱스 다시 등록	마이그레이션 후에 사용자 지정 인덱스를 다시 등록해야 합니다. registerCustomIndex SIF API를 사용하여 사용자 지정 인덱스를 다시 등록합니다. RegisterCustomIndex SIF API에 대한 자세한 내용은 <i>Multidomain MDM 서비스 통합 프레임워크 가이드</i> 를 참조하십시오. registerCustomIndex SIF API를 실행하는 SOAP 및 Java 코드 샘플은 KB 500116. https://kb.informatica.com/howto/6/Pages/19/500116.aspx?myk=500116 을 참조하십시오.

업그레이드 테스트

업그레이드된 MDM Hub를 테스트합니다. 각 Multidomain MDM 구현은 고유하며 테스트 요구 사항은 개발, 테스트 및 프로덕션 환경마다 다릅니다. 추천 업그레이드 테스트가 사용자 환경에 적합하지 않은 경우에는 고유의 테스트를 설계할 수 있습니다. 구현에 대한 고유 요구 사항을 충족하도록 테스트 활동을 설계합니다.

MDM Hub 업그레이드 테스트

환경에 적용되는 다음 Hub 콘솔 업그레이드 테스트를 수행합니다.

1. Hub 콘솔을 실행합니다.
2. 구성 작업 영역에서 **사용자** 도구를 선택하여 기존 사용자의 속성을 봅니다.
3. 모델 작업 영역에서 **스키마 뷰어** 도구를 선택한 다음 연산 참조 저장소에 연결합니다. **스키마 뷰어**에서 스키마를 검토합니다.
4. 기본 개체의 **일치/병합 설정**을 보려면 **모델** 작업 영역에서 **스키마** 도구를 선택합니다.
5. **유틸리티** 작업 영역에서 **일괄 처리 뷰어** 도구를 선택합니다. 가능한 경우 단계 일괄 작업, 로드 일괄 작업, 일치 일괄 작업 및 병합 일괄 작업에서 테스트 일괄 작업을 실행합니다.
6. **유틸리티** 작업 영역에서 처리 서버 도구를 선택합니다. 등록된 처리 서버에 대한 연결을 테스트합니다.
7. **모델** 작업 영역에서 **정리 함수** 도구를 선택합니다. 각 외부 정리 엔진에서 정리 함수 테스트를 실행합니다.
8. **데이터 스튜어드** 작업 영역에서 **데이터 관리자** 도구를 선택합니다. 두 개의 일치하는 테스트 레코드를 생성합니다.
9. **데이터 스튜어드** 작업 영역에서 **Merge Manager** 도구를 선택합니다. 두 개의 테스트 레코드를 찾고 테스트 레코드를 병합한 다음 테스트 레코드의 병합을 해제합니다.

사용자 지정 코드 업그레이드 테스트

사용자 지정 클라이언트 응용 프로그램 등의 사용자 지정 코드가 있는 경우 테스트를 실행하여 사용자 지정 코드가 예상대로 실행되는지 확인합니다.

프로비저닝 도구 업그레이드 테스트

프로비저닝 도구에 로그인합니다. 프로비저닝 도구는 비즈니스 항목, 참조 항목, 응용 프로그램, 사용자 지정 보기, 태스크 등에 대한 구성을 포함하는 XML 파일의 유효성을 검사합니다.

유효성 검사 프로세스가 성공하면 다음 업그레이드 테스트로 계속합니다. 필요한 경우 구성의 주요 부분을 검사하여 설정을 확인할 수 있습니다.

유효성 검사 프로세스에서 일부 오류가 감지되는 경우에는 오류 및 제안된 수정 목록을 검토합니다. 수정에는 일부 설정을 제거하는 작업이 포함될 수 있습니다. 모든 수정을 수락하거나 변경 없이 취소하도록 선택할 수 있습니다. 취소를 선택하는 경우 XML 파일의 오류를 직접 수정해야 합니다. XML 파일은 C_REPOS_CO_CS_CONFIG 및 C_REPOS_COMPONENT_INSTANCE 리포지토리 테이블에 저장됩니다.

주의: 오류를 수정하지 않고 종료하면 프로비저닝 도구가 잠길 수 있습니다.

1. 프로비저닝 도구에 로그인합니다.
2. 연산 참조 저장소 데이터베이스를 선택합니다.
3. 유효성 검사 오류가 표시되면 제안된 수정을 검토합니다.
 - 수정을 적용하려면 **수정**을 클릭합니다.

- 수정을 적용하지 않고 종료하려면 **취소**를 클릭합니다.
프로비저닝 도구에서 로그아웃됩니다. XML 파일을 열고 오류를 수정합니다.
4. 필요한 경우 XML 유효성이 검사된 후 구성 설정을 확인할 수 있습니다.

비즈니스 항목이 있는 Data Director 업그레이드 테스트

비즈니스 항목이 있는 Data Director를 사용하는 경우 응용 프로그램을 열고 테스트합니다.
환경에 적용되는 다음 업그레이드 테스트를 수행합니다.

1. Data Director에 로그인합니다.
2. 여러 검색을 실행합니다.
3. 여러 태스크를 생성 및 처리합니다.
4. 테스트 레코드를 삽입합니다.
5. 테스트 레코드를 복사하여 테스트 레코드를 하나 더 생성합니다.
6. 검색을 실행하여 두 개의 테스트 레코드를 찾습니다.
7. 두 개의 테스트 레코드를 병합 및 병합 해제합니다.

제목 영역이 있는 Data Director 업그레이드 테스트

제목 영역이 있는 Data Director를 사용하는 경우 테스트를 시작하기 전에 응용 프로그램을 배포해야 합니다.
환경에 적용되는 다음 업그레이드 테스트를 수행합니다.

1. Data Director 구성 관리자를 시작한 후 Data Director 응용 프로그램 인스턴스를 배포합니다.
2. Data Director에 로그인합니다.
3. 여러 검색을 실행합니다.
4. 여러 태스크를 생성 및 처리합니다.
5. 테스트 레코드를 삽입합니다.
6. 테스트 레코드를 복사하여 테스트 레코드를 하나 더 생성합니다.
7. 검색을 실행하여 두 개의 테스트 레코드를 찾습니다.
8. 두 개의 테스트 레코드를 병합 및 병합 해제합니다.

일반 Hub 서버 속성 구성

업그레이드 프로세스에서는 Hub 서버 속성 값이 유지됩니다. 이전 버전에서 업그레이드하는 경우 `cmxserver.properties` 파일에는 새로운 속성과 해당 릴리스에 추가된 속성 기본값에 대한 변경 내용이 포함되지 않습니다.

JBoss 환경에서 설치한 경우 `cmx.jboss7.management.port` 속성의 값을 9999에서 9990으로 변경합니다.

릴리스에 추가된 새 속성을 구성해야 합니다. 자세한 내용은 *Multidomain MDM 릴리스 가이드* 및 *Multidomain MDM 구성 가이드*를 참조하십시오.

Data Director 및 Hub 서버 속성

업그레이드 프로세스에서는 Data Director에 영향을 주는 Hub 서버 속성 값이 유지됩니다.

이전 버전에서 업그레이드할 경우 업그레이드 전 `cmxserver.properties` 파일에는 버전 10.0.0 이상에 추가된 일부 속성이 포함되지 않습니다. 업그레이드 프로세스에서 속성이 `cmxserver.properties` 파일에 추가되었는지 확인하십시오. 필요한 경우 누락된 속성을 <MDM Hub 설치 디렉터리>/hub/server/resources/cmxserver.properties 파일 끝에 추가합니다.

속성의 기본값은 Data Director 응용 프로그램의 현재 동작을 유지합니다. 속성을 사용자 지정하기 전에 *Multidomain MDM 구성 가이드*의 “Hub 서버 속성” 장에서 속성 설명을 읽어 보십시오.

버전 10.3에 추가된 속성

파일 업로드, 태스크 관리자 및 워크플로우 다이어그램을 지원하는 새로운 속성이 추가되었습니다.

```
# File upload properties
# -----
# Maximum upload size.
cmx.file.max_file_size_mb=20
# Maximum number of concurrent uploads.
cmx.file.max_concurrent_uploads=20
# Type of files that can be uploaded.
cmx.file.allowed_file_extensions=pdf,jpg
# Number of minutes until an uploaded file expires.
# To avoid expiration, set to 0.
cmx.server.attachment.temp.ttl_minutes=60

# Task Manager property
# -----
# Set to true to display the Task Manager tab in applications
# that use subject areas.
cmx.dataview.taskmanager.enabled=true

# Workflow diagram properties
# -----
# Set to true to display the workflow diagram associated
# with the tasks in the Task Manager for the users with
# the ActiveVOS abAdmin role.
cmx.e360.BPMProcess.view.enabled=false
cmx.e360.BPMProcess.view.autologout.seconds=30
```

참고: Data Director 응용 프로그램의 전체 텍스트 검색에 Elasticsearch를 사용하려는 경우 Elasticsearch 속성을 수동으로 추가합니다. 자세한 내용은 [“검색 구성 업그레이드 개요” 페이지 135](#) 항목을 참조하십시오.

버전 10.2 이하에 추가된 속성

데이터 탭, 검색 및 Entity 360 프레임워크를 지원하는 속성이 추가되었습니다. 기본값을 추가하고 필요에 따라 변경합니다.

```
# View properties
# -----
# Show or hide the views for subject areas.
cmx.dataview.enabled=true
# Show or hide the views for business entities.
cmx.e360.view.enabled=false
# Show or hide the Cross-reference view and Match view.
cmx.e360.match_xref.view.enabled=false

# Search with Solr (formerly Smart Search) properties
# -----
# Set to true to use Solr for search.
cmx.ss.enabled=false
```

Data Director 글로벌 속성

Data Director 환경에 표시되거나 숨겨진 열에 대한 사용자 기본 설정이 포함된 경우 이 버전의 암호화 해시 알고리즘 변경으로 인해 업그레이드 시 설정이 손실됩니다. 업그레이드 후 **C_REPOS_DS_PREF_DETAIL** 테이블을 지우고 사용자 기본 설정을 다시 생성하십시오.

사용자 기본 설정을 포함하여 Data Director 글로벌 속성을 업데이트하는 방법은 *Multidomain MDM Data Director 구현 가이드*를 참조하십시오.

비즈니스 항목 스키마 생성

비즈니스 항목 서비스를 사용하는 경우에는 업그레이드 후 비즈니스 항목 스키마를 생성해야 합니다. 비즈니스 항목 스키마는 Informatica Data Director 구성 관리자를 사용하여 생성할 수 있습니다.

비즈니스 항목 스키마를 생성하기 전에 사용자 지정 비즈니스 항목 또는 비즈니스 항목 서비스 구성 파일을 모두 백업하십시오.

Informatica Data Director 구성 관리자에서 비즈니스 항목 스키마를 생성하려면 응용 프로그램 화면에서 **비즈니스 항목 스키마 생성**을 클릭합니다.

제 10 장

검색 구성 업그레이드

이 장에 포함된 항목:

- [검색 구성 업그레이드 개요, 135](#)
- [1단계. Elasticsearch 설치 및 설정, 135](#)
- [2단계. 검색에 대한 MDM Hub 속성 구성, 140](#)
- [3단계. 프로비저닝 도구를 사용하여 검색 구성, 141](#)
- [4단계. 연산 참조 저장소 유효성 검사, 143](#)
- [5단계. 검색 데이터 인덱싱, 144](#)
- [Elasticsearch 버전 6.8.6으로 업그레이드\(선택 사항\), 144](#)

검색 구성 업그레이드 개요

Data Director 응용 프로그램 또는 사용자 지정 응용 프로그램을 사용하여 특정 비즈니스 항목 내의 데이터를 검색할 수 있습니다. 이전에는 검색 작업을 위해 Solr를 구성했지만 이것은 더 이상 지원되지 않습니다. 검색 작업에 Elasticsearch를 구성해야 합니다.

Elasticsearch를 사용하도록 검색 구성을 업그레이드하려면 다음 태스크를 수행합니다.

1. Elasticsearch를 설치 및 설정합니다.
2. 검색에 대한 MDM Hub 속성을 구성합니다.
3. 프로비저닝 도구를 사용하여 검색을 구성합니다.
4. ORS(연산 참조 저장소) 유효성을 검사합니다.
5. 검색 데이터를 인덱싱합니다.

1단계. Elasticsearch 설치 및 설정

검색을 구성하려면 Elasticsearch를 설치 및 설정해야 합니다.

Elasticsearch를 설정하려면 다음 태스크를 수행합니다.

1. 설치 전 태스크를 완료합니다.
2. Elasticsearch를 설치합니다.
3. Elasticsearch JVM(Java Virtual Machine)을 구성합니다.

4. Elasticsearch 속성 파일을 구성합니다.
5. Elasticsearch를 보호합니다.
6. 분석 플러그 인을 설치합니다.
7. 정지 단어, 동의어 및 문자 매핑 구성
8. Elasticsearch를 시작합니다.

설치 전 태스크 완료

Elasticsearch 클러스터를 설치 및 설정하기 전에 환경을 준비하고 고가용성을 구성할지 결정합니다.

모든 환경의 태스크

다음 태스크를 수행하여 설치 환경을 준비합니다.

- 각 시스템이 지원되는 Elasticsearch 버전에 대한 하드웨어 요구 사항을 충족하는지 확인합니다. 하드웨어에 대한 자세한 내용은 Elasticsearch 설명서를 참조하십시오.
- 각 시스템이 지원 운영 체제 및 Java 버전과 같은 지원되는 Elasticsearch 버전에 대한 소프트웨어 요구 사항을 충족하는지 확인합니다. 소프트웨어 요구 사항에 대한 자세한 내용은 *Elasticsearch Support Matrix*를 참조하십시오.
- 스왑, 파일 설명자, 가상 메모리와 같은 중요 시스템 구성을 완료합니다. 중요 시스템 구성에 대한 자세한 내용은 Elasticsearch 설명서를 참조하십시오.

UNIX 환경의 태스크

UNIX 환경에서 다음 태스크를 수행합니다.

- 파일 설명자 수 부족으로 인한 데이터 손실을 방지하도록 파일 설명자의 수를 65536 이상으로 설정합니다.
- 메모리 스왑을 방지하려면 스왑을 방지하도록 시스템을 구성합니다. `mlockall`을 통해 메모리의 힙을 잠그도록 JVM(Java Virtual Machine)을 구성할 수 있습니다.

고가용성 요구 사항

인덱싱 및 검색할 대량의 데이터가 있는 경우에는 고가용성 Elasticsearch 클러스터를 구현하는 것이 좋습니다. 고가용성 클러스터에는 여러 개의 노드가 있으며 클러스터가 노드 간에 워크로드를 분배할 수 있습니다. 프로덕션 환경에서 하나의 노드가 실패하는 경우 클러스터가 워크로드를 다른 노드에 분배합니다.

설치 전 태스크의 하나로, 고가용성 Elasticsearch 클러스터를 구현할지 결정합니다. 구현할 경우 Elasticsearch 클러스터를 평소대로 구성하되 다음과 같은 추가 요구 사항을 충족해야 합니다.

- Elasticsearch 클러스터에 3개 이상의 노드가 있습니다.
 - 팁:** 소형 클러스터를 설정하여 클러스터를 시작하고 필요에 따라 확장할 수 있습니다. 워크로드를 분석하여 노드 실패를 처리할 수 있는 충분한 용량이 있는지 확인해야 합니다.
- 각 노드는 별도의 전용 시스템에 구성되어 있습니다.
- 안정성과 성능을 보장하기 위해 마스터 노드가 3개 이상입니다. Elasticsearch에서는 마스터 노드를 홀수로 구성하는 것을 권장합니다.
 - 클러스터에 노드가 3개만 있다면 모든 노드를 마스터 노드로 구성합니다.
 - 클러스터에 4개 이상의 노드가 있는 경우에는 3개의 노드를 마스터 노드로 구성하고 나머지 노드를 데이터 노드로 구성합니다.
- Elasticsearch 클러스터 크기에 따라 복제본의 수를 결정합니다. 프로비저닝 도구를 사용하여 Elasticsearch 인덱스를 구성할 때 사용할 복제본의 수를 지정할 수 있습니다.

- 각 노드의 `elasticsearch.yml` 구성 파일에서 다음과 같은 추가 속성을 설정합니다.
 - `discovery.zen.minimum_master_nodes`
 - `discovery.zen.ping.unicast.hosts`

하드웨어 요구 사항, 시스템 구성, 속성 값을 포함한 고가용성 클러스터에 대한 자세한 내용은 **Elasticsearch** 설명서를 참조하십시오.

Elasticsearch 설치

Hub 서버 및 처리 서버를 설치한 후 검색을 구성하려면 **Elasticsearch** 클러스터를 설치 및 설정합니다.

Elasticsearch 설치에 지원되는 운영 체제 및 **Java** 버전을 사용하는지 확인하십시오. 자세한 내용은 **Elasticsearch Support Matrix**를 참조하십시오.

Elasticsearch를 설치하고 클러스터를 설정하는 방법에 대한 자세한 내용은 **Elasticsearch** 설명서를 참조하십시오.

1. **Elastic** 웹 사이트에서 지원되는 버전의 **Elasticsearch** 보관 파일을 다운로드합니다.
지원되는 버전에 대한 자세한 내용은 **PAM(Product Availability Matrix)**을 참조하십시오. **PAM**은 <https://network.informatica.com/community/informatica-network/product-availability-matrices>에서 액세스할 수 있습니다.
2. **Elasticsearch** 보관 파일을 추출합니다.

Elasticsearch JVM(Java Virtual Machine) 구성

시스템에서 사용 가능한 **RAM** 양에 따라 힙 크기를 사용하도록 **Elasticsearch JVM(Java Virtual Machine)**을 구성합니다. **JVM**을 구성하려면 `jvm.options` 파일을 편집합니다.

1. 다음 디렉터리에서 `jvm.options` 파일을 찾습니다.
<elasticsearch 설치 디렉터리>/config
2. 텍스트 편집기를 사용하여 파일을 열고 다음 속성을 편집합니다.

속성	설명
-Xms	최소 힙 크기입니다. 기본값은 1GB입니다.
-Xmx	최대 힙 크기입니다. 기본값은 1GB입니다.
-XX:HeapDumpPath	힙 덤프 경로입니다. 기본값은 <code>/var/lib/elasticsearch</code> 입니다. 다중 클러스터 환경에서는 이 속성을 대체 경로로 설정해야 합니다.

참고: 최소 힙 크기(`Xms`) 및 최대 힙 크기(`Xmx`)를 동일한 값으로 설정합니다. 다른 속성의 경우 기본 설정을 사용합니다.

Elasticsearch 속성 파일 구성

Informatica는 샘플 **Elasticsearch** 속성 파일을 제공합니다. **Elasticsearch**를 구성하려면 속성 파일을 편집합니다.

1. 다음 디렉터리에서 `elasticsearch.yml` 파일을 찾습니다.
<elasticsearch 설치 디렉터리>/config

2. 텍스트 편집기를 사용하여 파일을 열고 다음 속성을 편집합니다.

속성	설명
bootstrap.memory_lock	메모리 잠금을 설정합니다. Elasticsearch 메모리 스왑을 방지하려면 true로 설정합니다. 기본값은 true입니다.
cluster.name	Elasticsearch 클러스터의 고유한 이름을 지정합니다. 클러스터가 여러 개인 경우 각 클러스터의 이름은 고유해야 합니다. 클러스터에 다수의 노드가 있는 경우 클러스터의 각 노드에 동일한 클러스터 이름을 지정해야 합니다.
discovery.zen.minimum_master_nodes	다중 노드 클러스터에서 데이터 손실을 방지하고 클러스터 안정성을 유지하는 데 필요합니다. 다음 값으로 설정합니다. (마스터 적격 노드의 수/2) + 1 예를 들어 클러스터에 3개의 노드가 있고 이들 모두 마스터 노드로 사용하고 데이터를 포함할 수 있는 경우 속성을 (3/2) + 1(2로 반올림됨)로 설정합니다.
discovery.zen.ping.unicast.hosts	다중 노드 클러스터에 필요합니다. 이 속성은 클러스터 노드의 IP 주소 및 전송 포트 목록인 검색 설정을 지정하는 데 사용됩니다. 다음 형식을 사용하여 속성을 지정합니다. ["host1:port1", "host2:port2", "host3:port3"]
http.port	HTTP 요청의 포트입니다. 기본값은 9200입니다.
network.host	바인딩 주소로 사용할 호스트의 IP 주소입니다.
node.data	CRUD 및 검색 등 데이터 관련 작업을 수행하는 데이터 노드로 노드를 활성화합니다. 기본값은 true입니다.
node.ingest	인덱싱 전에 데이터를 변환하고 보강하는 수집 노드로 노드를 활성화합니다. 기본값은 true입니다.
node.master	클러스터를 제어하는 마스터 노드로 노드를 활성화합니다. 클러스터에 여러 개의 노드가 있는 경우 하나 이상의 노드를 마스터 노드로 활성화합니다.고가용성의 경우 여러 노드를 마스터 노드로 설정합니다. 기본값은 true입니다.
node.name	노드의 고유한 이름을 지정합니다.
path.data	데이터를 저장하려는 디렉터리의 경로입니다. 여러 데이터 디렉터리를 구성할 수 있습니다. 여러 데이터 디렉터리 구성에 대한 자세한 내용은 Elasticsearch 설명서를 참조하십시오.
path.logs	로그 파일의 경로입니다.
transport.tcp.port	TCP 바인딩 포트입니다. 기본값은 9300입니다.

3. 속성 파일을 동일한 이름인 `elasticsearch.yml`로 저장합니다.

Elasticsearch 보호

Elasticsearch를 설치한 후 MDM Hub와 Elasticsearch 간 통신을 보호합니다. 또한 Elasticsearch 클러스터를 보호합니다.

Elasticsearch 보호에 대한 자세한 내용은 Elasticsearch 보안 설명서를 참조하십시오.

분석 플러그인 설치

새로운 분석기, 토큰라이저, 토큰 필터 및 문자 필터를 추가하여 Elasticsearch를 확장하는 음성 및 일본어(kuromoji) 분석 플러그인을 설치합니다. 음성 분석 플러그인은 토큰을 분석하여 해당하는 음성 값으로 변환합니다. 일본어(kuromoji) 분석 플러그인은 Kuromoji 분석기를 사용하여 일본어를 분석합니다.

1. 음성 및 일본어(kuromoji) 분석 플러그인을 Elastic 웹 사이트에서 다운로드합니다.
2. 다음 명령을 실행하여 각 클러스터 노드에서 음성 분석 플러그인을 설치합니다.
`sudo bin/elasticsearch-plugin install analysis-phonetic`
3. 다음 명령을 실행하여 각 클러스터 노드에서 일본어(kuromoji) 분석 플러그인을 설치합니다.
`sudo bin/elasticsearch-plugin install analysis-kuromoji`
4. 각 클러스터 노드를 다시 시작합니다.

정지 단어, 동의어 및 문자 매핑 구성

검색을 수행할 때 MDM에서 "and", "an" 및 "is" 같은 일반적인 단어가 무시될 수 있습니다. 또한 MDM에서는 검색 문자열의 동의어도 검색될 수 있습니다. 예를 들어 "William"을 검색하는 경우 동의어인 "Will"과 "Willy"가 검색 결과에 포함될 수 있습니다.

일반적인 단어를 무시하거나 검색 결과에 동의어를 포함하도록 구성하려면 정지 단어 및 동의어가 포함된 Informatica의 텍스트 파일을 사용하거나 직접 구성할 수 있습니다.

중국어, 일본어 및 한국어와 같은 언어에 대해 기본 Elasticsearch 분석기를 사용하려면 Informatica가 제공하는 mapping-FoldToASCII.txt 매핑 파일을 사용할 수 있습니다. 이러한 기본 분석기의 문자 필터는 매핑 파일을 사용하여 기본 라틴 유니코드에 없는 알파벳, 숫자 및 기호 문자를 해당하는 ASCII 문자로 변환합니다.

stopwords.txt, synonyms.txt, stopwords_ja.txt 및 mapping-FoldToASCII.txt 파일을 받으려면 Informatica 글로벌 고객 지원 센터에 문의하십시오.

정지 단어, 동의어 및 문자 매핑을 구성하려면 다음 단계를 수행하십시오.

1. 다음 위치에 analysis 디렉터리를 생성합니다.
`<elasticsearch 설치 디렉터리>/config`
2. stopwords.txt 및 synonyms.txt 파일을 analysis 디렉터리에 복사합니다.
3. 일본어와 같은 언어에 대한 정지 단어를 구성하려면 다음 위치에 lang 디렉터리를 생성합니다.
`<elasticsearch 설치 디렉터리>/config/analysis`
4. 다른 언어에 대한 정지 단어 파일(예: stopwords_ja.txt)과 mapping-FoldToASCII.txt 파일을 lang 디렉터리에 복사합니다.

Elasticsearch 시작

Elasticsearch를 설정한 후 변경 내용을 적용하려면 Elasticsearch 클러스터의 각 노드를 시작합니다.

팁: Elasticsearch를 시작할 때 메모리 잠금 문제가 발생하면 soft memlock unlimited 및 hard memlock unlimited를 설정해야 할 수 있습니다.

1. 명령 프롬프트를 열고 다음 디렉터리로 변경합니다.
`<elasticsearch 설치 디렉터리>/bin`
2. 다음 명령을 실행합니다.
UNIX의 경우. `elasticsearch.sh`
Windows의 경우. `elasticsearch.bat`

2단계. 검색에 대한 MDM Hub 속성 구성

MDM Hub 속성을 구성하려면 Hub 콘솔, 처리 서버 속성 파일 및 Hub 서버 속성 파일을 사용합니다.

1. 처리 서버 속성을 구성합니다.
2. Hub 서버 속성을 구성합니다.

검색에 대한 Hub 서버 구성

모든 Hub 서버 인스턴스에서 검색을 활성화하도록 구성해야 합니다. Hub 콘솔의 Hub 서버 도구와 `cmxserver.properties` 파일을 사용하여 검색에 대한 Hub 서버 속성을 구성합니다.

1. 텍스트 편집기를 사용하여 다음 위치의 `cmxserver.properties` 파일을 엽니다. <MDM Hub 설치 디렉터리>\hub\server\resources\cmxserver.properties
2. 검색에 대한 다음 속성을 구성합니다.

`cmx.ss.engine`

Elasticsearch 엔진을 검색에 사용하려는 경우 필요합니다. 속성을 수동으로 추가하고 `es`로 설정합니다.

`ex.max.conn.per.host`

호스트에 연결하려는 Elasticsearch 노드의 최대 수를 설정합니다. 호스트의 Elasticsearch 클러스터 노드 수로 설정합니다.

`ex.max.threads`

Elasticsearch 클러스터의 각 노드에 사용할 Apache 비동기식 비차단 수신기의 최대 스레드 수를 설정합니다. 기본값은 1입니다.

이 값은 Informatica 글로벌 고객 지원 센터에서 제안한 경우에만 변경합니다.

`es.index.refresh.interval`

처음에 스마트 검색 데이터 인덱싱 일괄 작업이 실행된 후 Elasticsearch에서 데이터 변경 내용을 커밋하는 간격(초)을 설정합니다. 이 시간 간격 후에 데이터를 검색에 사용할 수 있습니다. 기본값은 30입니다.

이 속성은 초기 인덱싱 중에 발생하는 높은 인덱싱 볼륨에 영향을 미칩니다. 이 값은 Informatica 글로벌 고객 지원 센터에서 제안한 경우에만 변경합니다.

`ssl.keyStore`

응용 프로그램 서버의 HTTPS 포트를 사용하여 Hub 서버를 구성하는 경우 필수입니다. 속성을 수동으로 추가합니다. 키 저장소 파일의 절대 경로 및 파일 이름입니다.

`ssl.keyStore.password`

응용 프로그램 서버의 HTTPS 포트를 사용하여 Hub 서버를 구성하는 경우 필수입니다. 속성을 수동으로 추가합니다. 키 저장소 파일에 대한 일반 텍스트 암호입니다.

`ssl.trustStore`

응용 프로그램 서버의 HTTPS 포트를 사용하여 Hub 서버를 구성하는 경우 필수입니다. 속성을 수동으로 추가합니다. 트러스트 저장소 파일의 절대 경로 및 파일 이름입니다.

`ssl.trustStore.password`

응용 프로그램 서버의 HTTPS 포트를 사용하여 Hub 서버를 구성하는 경우 필수입니다. 속성을 수동으로 추가합니다. 트러스트 저장소 파일에 대한 일반 텍스트 암호입니다.

Hub 서버 속성을 업데이트한 후 ORS(연산 참조 저장소) 유효성을 검사하고 Hub 콘솔을 다시 시작해야 합니다.

검색에 대한 처리 서버 구성

Elasticsearch로 검색을 구성하는 경우 모든 처리 서버 인스턴스에서 검색을 활성화합니다. Hub 콘솔의 처리 서버 도구와 `cmxcleanse.properties` 파일을 사용하여 검색에 대한 처리 서버 속성을 구성합니다.

1. 노드의 Hub 콘솔에서 처리 서버 도구를 시작합니다.
2. 쓰기 잠금 > 잠금 획득을 클릭합니다.
3. 처리 서버 도구의 오른쪽 창에서 **처리 서버 추가** 단추를 클릭합니다.
처리 서버 추가/편집 대화 상자가 표시됩니다.
4. 검색을 활성화하려면 처리 서버의 **검색 처리 활성화** 속성을 선택합니다.
5. **확인**을 클릭한 다음 **저장**을 클릭합니다.
6. `cmxcleanse.properties` 파일을 편집합니다.

`cmxcleanse.properties` 파일은 다음 위치에 있습니다. <MDM Hub 설치 디렉터리>\hub\cleanse\resources

- a. 검색에 대한 다음 속성을 구성합니다.

`cmx.ss.engine`

Elasticsearch 엔진을 검색에 사용하려는 경우 필요합니다. 속성을 수동으로 추가하고 `es`로 설정합니다.

`ex.max.conn.per.host`

호스트에 연결하려는 Elasticsearch 노드의 최대 수를 설정합니다. 호스트의 Elasticsearch 클러스터 노드 수로 설정합니다.

`ex.max.threads`

Elasticsearch 클러스터의 각 노드에 사용할 Apache 비동기식 비차단 수신기의 최대 스레드 수를 설정합니다. 기본값은 1입니다.

이 값은 Informatica 글로벌 고객 지원 센터에서 제안한 경우에만 변경합니다.

- b. Solr 검색에 대해 구성된 다음 속성을 제거합니다.

- `solr.allowAdminConsole`
- `zookeeper.tickTime`
- `pingSolrOnStartup`

7. `cmxcleanse.properties` 파일을 저장합니다.
8. 응용 프로그램 서버를 다시 시작합니다.

3단계. 프로비저닝 도구를 사용하여 검색 구성

Elasticsearch를 설정하고 MDM Hub 속성을 구성한 후 프로비저닝 도구를 사용하여 검색 환경을 구성합니다.

1. Elasticsearch 클러스터를 구성합니다.
2. 검색 결과 보기를 구성합니다.

Elasticsearch 클러스터 구성

프로비저닝 도구를 사용하여 MDM 응용 프로그램의 Elasticsearch 클러스터를 구성하십시오. 이 구성은 검색 API에 사용됩니다. 검색 API는 Data Director 응용 프로그램 및 모든 사용자 지정 응용 프로그램에서 사용됩니다.

참고: Elasticsearch 클러스터를 구성할 때는 클러스터의 마스터 노드만 지정해야 합니다.

1. 지원되는 브라우저를 열고 다음 URL을 입력합니다.
`https://<MDM Hub Server host name>:<MDM Hub Server port number>/provisioning/`
 로그인 페이지가 나타납니다.
2. 사용자 이름 및 암호를 입력하고 **로그인**을 클릭합니다.
3. 데이터베이스 목록에서 Elasticsearch 클러스터를 구성하려는 데이터베이스를 선택합니다.
4. **구성 > 인프라 설정**을 클릭합니다.
인프라 설정 페이지가 표시됩니다.
5. 목록에서 **Elasticsearch 클러스터**를 선택한 다음 **ESCluster**를 클릭합니다.
ESCluster가 트리 보기 패널에 나타납니다.
6. Elasticsearch 클러스터 노드를 구성하려면 트리 보기 패널에서 **esNode**를 선택하고 **생성**을 클릭합니다.
7. 구성된 Elasticsearch 클러스터의 다음 속성을 지정합니다.

속성	설명
이름	Elasticsearch 클러스터의 마스터 노드 이름입니다.
URL	Elasticsearch 클러스터의 마스터 노드 URL입니다. URL 형식은 <code>https://<호스트 이름>:<포트></code> 입니다.

8. **적용**을 클릭합니다.
9. 추가 마스터 노드를 생성하려면 6~8 단계를 반복합니다.
10. 변경 내용을 MDM Hub에 게시합니다.
 - a. **게시**를 클릭합니다.
 변경 내용을 게시 또는 검토하라는 메시지를 표시하는 확인 대화 상자가 나타납니다.
 - b. 변경 내용을 검토하거나 검토 없이 게시합니다.
 - 검토 없이 게시하려면 **게시**를 클릭합니다.
 - 검토 후 게시하려면 **변경 내용 검토**를 클릭하고 화면에 나타나는 지시 사항을 따릅니다.

검색 또는 쿼리 결과 표시 구성

프로비저닝 도구를 사용하여 검색에 사용할 비즈니스 항목 보기를 구성할 수 있습니다. 검색 결과에는 검색 결과에 구성된 비즈니스 항목 보기의 일부인 필드만 포함됩니다. 검색 필터가 표시되는 순서를 구성할 수도 있습니다.

검색 가능한 보기를 구성하기 전에 검색 결과에 사용하려는 비즈니스 항목 보기를 생성합니다.

참고: 검색 결과에 비즈니스 항목의 하위 레코드 필드를 표시하려면 비즈니스 항목에서 변환된 비즈니스 항목 보기를 사용합니다. 보기가 루트 레코드 수준의 하위 레코드 필드를 포함하는지 확인합니다.

1. 프로비저닝 도구에 로그인합니다.
2. 데이터베이스 목록에서 응용 프로그램에 연결된 데이터베이스를 선택합니다.

3. **구성 > 응용 프로그램 편집기**를 클릭합니다.
응용 프로그램 페이지가 표시됩니다.
4. **응용 프로그램** 목록에서 검색을 구성하려는 응용 프로그램을 선택합니다.
응용 프로그램이 없는 경우 검색을 구성하기 전에 생성합니다.
5. 트리 보기 패널에서 **검색 구성**을 선택한 다음, **생성**을 클릭합니다.
6. 속성 패널에서 비즈니스 항목을 선택하고 검색 또는 쿼리 결과를 표시하는 데 사용할 비즈니스 항목 보기를 선택합니다.
비즈니스 항목 보기를 선택하지 않은 경우 검색 및 쿼리 결과에 모든 비즈니스 항목 필드가 포함됩니다.
7. 필요한 경우, 검색을 구성했다면 필터를 선택하고 검색 필터의 표시 순서를 구성합니다.
 - a. **필터 표시 순서** 옆에 있는 **편집** 아이콘을 클릭합니다.
필터 표시 순서 편집 대화 상자가 나타납니다. 대화 상자에는 필터가 포함되어 있으며, 이러한 필터는 비즈니스 항목 모델에서 필터링 가능으로 구성된 필드입니다.
 - b. **사용 가능한 필터** 섹션에서 **선택한 필터** 섹션으로 필터를 끌어다 놓습니다.
 - c. 순서를 구성하려면 필터를 위 또는 아래로 끌어 이동합니다.
 - d. **확인**을 클릭합니다.
8. **적용**을 클릭합니다.
검색 구성이 임시 작업 공간에 저장됩니다.
9. 변경 내용을 **MDM Hub**에 게시합니다.
 - a. **게시**를 클릭합니다.
변경 내용을 게시 또는 검토하라는 메시지를 표시하는 확인 대화 상자가 나타납니다.
 - b. 변경 내용을 검토하거나 검토 없이 게시합니다.
 - 검토 없이 게시하려면 **게시**를 클릭합니다.
 - 검토 후 게시하려면 **변경 내용 검토**를 클릭하고 화면에 나타나는 지시 사항을 따릅니다.

4단계. 연산 참조 저장소 유효성 검사

Elasticsearch 구성의 영향을 받는 ORS(연산 참조 저장소)의 메타데이터 유효성을 검사하려면 Hub 콘솔에서 리포지토리 관리자 도구를 사용합니다.

1. Hub 콘솔을 시작하고 **MDM Hub** 마스터 데이터베이스에 연결합니다.
2. **구성** 작업 영역을 확장하고 **리포지토리 관리자**를 클릭합니다.
리포지토리 관리자가 나타납니다.
3. **유효성 검사** 탭을 클릭하고 유효성을 검사할 리포지토리를 선택합니다.
4. **유효성 검사**를 클릭합니다.
유효성 검사 선택 대화 상자가 나타납니다.
5. 수행할 유효성 검사를 선택합니다.
6. **확인**을 클릭합니다.
리포지토리 관리자가 리포지토리의 유효성을 검사하고 문제가 있으면 **발견된 문제** 창에 모두 표시합니다.
7. 문제를 복구하려면 **복구**를 클릭합니다.

5단계. 검색 데이터 인덱싱

환경에 데이터가 포함되는 경우 처음에 스마트 검색 데이터 인덱싱 일괄 작업을 수동으로 실행하여 데이터를 인덱싱합니다. 환경에 데이터가 포함되지 않는 경우 처음에 스마트 검색 데이터 인덱싱 작업을 실행하지 않아도 됩니다. 로드 일괄 작업을 시행하여 데이터를 로드하면 로드 일괄 작업이 처음에 스마트 검색 데이터 인덱싱 일괄 작업을 자동으로 실행하고 데이터를 인덱싱합니다. 검색 요청은 인덱스를 사용하여 레코드를 검색합니다.

비즈니스 항목에 기여하는 모든 기본 개체에 대해 처음에 스마트 검색 데이터 인덱싱 일괄 작업을 실행합니다. 처음에 스마트 검색 데이터 인덱싱 일괄 작업을 기본 개체에 실행하면 **Elasticsearch** 서버가 검색 가능한 필드의 데이터를 인덱싱합니다. 인덱싱된 데이터는 검색 가능한 필드가 속하는 비즈니스 항목을 나타내는 모든 컬렉션에 추가됩니다. 컬렉션이 너무 큰 경우 컬렉션을 하나 이상의 **shard**로 분할할 수 있습니다. **shard**는 여러 노드에서 분할된 컬렉션의 논리적 단위입니다. 검색을 수행하면 **Elasticsearch** 서버가 컬렉션을 읽고 일치 필드를 반환합니다.

처음에 스마트 검색 데이터 인덱싱 일괄 작업은 작업에서 모든 레코드에 대한 인덱싱 요청을 대기열에 추가한 후에 레코드를 비동기적으로 인덱싱하고 작업의 성공적인 완료를 보고합니다. 검색 요청에서 인덱싱된 레코드는 인덱스 요청이 완료된 후에만 표시되며 몇 분이 소요될 수 있습니다.

중요: 데이터를 인덱싱한 후 필드의 검색 가능한 속성을 업데이트하는 경우 인덱스가 삭제됩니다. 처음에 스마트 검색 데이터 인덱싱 일괄 작업을 실행하여 데이터를 인덱싱해야 합니다. 또한 인덱싱 프로세스는 리소스 소모가 많은 프로세스이므로 처음에 스마트 검색 데이터 인덱싱 일괄 작업 여러 개를 동시에 실행하지 마십시오.

Elasticsearch 버전 6.8.6으로 업그레이드(선택 사항)

기존 MDM 환경에서 Search Guard가 포함된 Elasticsearch 버전 6.2.3을 사용하는 경우 X-Pack Community Edition이 포함된 Elasticsearch 버전 6.8.6으로 업그레이드하는 것이 좋습니다. Elasticsearch 버전 6.8.6은 이전 버전과 호환됩니다.

Elasticsearch 버전 6.8.6으로 업그레이드하기 전 선행 조건

Elasticsearch 버전 6.8.6으로 업그레이드하기 전에 기존 Elasticsearch 버전 6.2.3의 `elasticsearch.yml` 파일에 다음 속성이 있는지 확인해야 합니다.

- `cluster.name`: <사용자 정의 값>
- `node.name`: <사용자 정의 값>

참고: 이러한 속성을 기존 Elasticsearch 버전 6.2.3 구성 디렉터리에서 Elasticsearch 6.8.6 구성 디렉터리로 마이그레이션해야 합니다. 나열된 정확한 순서로 마이그레이션 단계를 실행합니다.

기존 Elasticsearch 버전 6.2.3에 이러한 구성 속성이 없는 경우 기존 Elasticsearch 버전이 호환되지 않으므로 Elasticsearch 버전 6.8.6으로 업그레이드하지 마십시오.

Elasticsearch 속성 파일 구성

Elasticsearch 6.8.6을 구성하려면 속성 파일을 편집합니다.

1. 다음 디렉터리로 이동합니다.
`<elasticsearch 6.8.6 설치 디렉터리>/config`

2. 텍스트 편집기에서 `elasticsearch.yml` 파일을 열고 다음 속성을 편집합니다.

속성	설명
<code>bootstrap.memory_lock</code>	메모리를 잠그려면 <code>true</code> 로 설정합니다. 기본값은 <code>true</code> 입니다.
<code>cluster.name</code>	Elasticsearch 클러스터의 고유한 이름입니다. 클러스터가 여러 개인 경우 각 클러스터의 이름은 고유해야 합니다. 클러스터에 다수의 노드가 있는 경우 클러스터의 각 노드에 동일한 클러스터 이름을 지정해야 합니다.
<code>http.port</code>	HTTP 요청의 포트입니다. Elasticsearch 6.8.6에 대한 HTTP 포트가 활성화되었고 포트 번호가 Elasticsearch 6.2.3 포트 번호와 충돌하지 않는지 확인하십시오. 기본값은 9200입니다.
<code>network.host</code>	바인딩 주소로 사용할 호스트의 IP 주소입니다.
<code>node.data</code>	노드를 CRUD 및 검색 등 데이터 관련 작업을 수행하는 데이터 노드로 사용할지 여부를 나타냅니다. <code>true</code> 로 설정합니다. 기본값은 <code>true</code> 입니다.
<code>node.ingest</code>	노드를 인덱싱 전에 데이터를 변환하고 보강하는 수집 노드로 사용할지 여부를 나타냅니다. <code>true</code> 로 설정합니다. 기본값은 <code>true</code> 입니다.
<code>node.master</code>	노드를 클러스터를 제어하는 마스터 노드로 사용할지 여부를 나타냅니다. 클러스터에 여러 개의 노드가 있는 경우 하나 이상의 노드를 마스터 노드로 활성화합니다.고가용성의 경우 여러 노드를 마스터 노드로 설정합니다. <code>true</code> 로 설정합니다. 기본값은 <code>true</code> 입니다.
<code>node.name</code>	노드의 고유한 이름을 지정합니다.
<code>transport.tcp.port</code>	TCP 바인딩 포트입니다. Elasticsearch 6.8.6에 대한 TCP 바인딩 포트가 활성화되었고 포트 번호가 Elasticsearch 6.2.3 포트 번호와 충돌하지 않는지 확인하십시오. 기본값은 9300입니다.

3. `elasticsearch.yml` 파일을 저장합니다.
4. 분석 플러그 인을 설치합니다. 분석 플러그 인을 설치하는 방법에 대한 자세한 내용은 [“분석 플러그 인 설치” 페이지 139](#) 항목을 참조하십시오.

Elasticsearch 인덱스 업그레이드

기존 Elasticsearch 버전 6.2.3 인덱스 속성을 Elasticsearch 버전 6.8.6 인덱스 속성으로 업그레이드하십시오. 인덱스를 업그레이드하려면 Elasticsearch 구성 속성 파일을 편집합니다.

1. 다음 디렉터리로 이동합니다.
`<elasticsearch 6.2.3 설치 디렉터리>/config`
2. `analysis` 폴더를 복사합니다.
3. `analysis` 폴더를 다음 디렉터리에 붙여 넣습니다.
`<elasticsearch 6.8.6 설치 디렉터리>/config`
4. Elasticsearch 버전 6.8.6 구성 디렉터리에서 `elasticsearch.yml` 파일의 다음 속성을 편집합니다.

5. 텍스트 편집기를 사용하여 파일을 열고 다음 속성을 편집합니다.

속성	설명
xpack.security.transport.ssl.enabled	REST 계층에서 TLS(Transport Layer Security)를 활성화할지 여부를 지정합니다. true로 설정합니다. 기본값은 true입니다.
xpack.security.transport.ssl.verification_mode	인증서 확인을 위한 모드입니다. 지원되는 모드는 certificate입니다. certificate라는 값을 입력합니다.
xpack.security.transport.ssl.keystore.path	키 저장소 파일의 상대 경로입니다. sip_ant 스크립트를 사용하여 키 저장소를 생성한 경우 기본 경로를 사용합니다. 기본값은 MDM_ESKEYSTORE_FILE_JKS.keystore입니다.
xpack.security.transport.ssl.truststore.path	트러스트 저장소 파일의 상대 경로입니다. sip_ant 스크립트를 사용하여 트러스트 저장소를 생성한 경우 기본 경로를 사용합니다. 기본값은 MDM_TRUST_JKS.keystore입니다.
xpack.security.transport.ssl.keystore.password	키 저장소의 암호입니다. 기본값은 infa@es입니다.
xpack.security.transport.ssl.truststore.password	트러스트 저장소의 암호입니다. 기본값은 infa@es입니다.
xpack.security.http.ssl.enabled	X-Pack을 활성화할지 여부를 지정합니다. true로 설정합니다.
xpack.security.http.ssl.verification_mode	인증서 확인을 위한 모드입니다. 지원되는 모드는 certificate입니다. certificate라는 값을 입력합니다.
xpack.security.http.ssl.client_authentication	REST 계층에서 TLS 클라이언트 인증 모드를 사용할지 여부를 지정합니다. REQUIRED로 설정합니다.
xpack.security.http.ssl.keystore.path	키 저장소 파일의 상대 경로입니다. sip_ant 스크립트를 사용하여 키 저장소를 생성한 경우 기본 경로를 사용합니다. 기본값은 MDM_ESCLIENT_FILE_JKS.keystore입니다.
xpack.security.http.ssl.truststore.path	트러스트 저장소 파일의 상대 경로입니다. sip_ant 스크립트를 사용하여 트러스트 저장소를 생성한 경우 기본 경로를 사용합니다. 기본값은 MDM_TRUST_JKS.keystore입니다.
xpack.security.http.ssl.truststore.password	트러스트 저장소의 암호입니다. 기본값은 infa@es입니다.
xpack.security.http.ssl.keystore.password	키 저장소의 암호입니다. 기본값은 infa@es입니다.
xpack.security.authc.token.enabled	인증 토큰을 활성화할지 여부를 지정합니다. 값은 항상 false로 설정되어야 합니다.
xpack.security.authc.api_key.enabled	API 키 토큰을 활성화할지 여부를 지정합니다. 값은 항상 false로 설정되어야 합니다.

6. Elasticsearch 버전 6.8.6 구성 디렉터리에서 roles.yml 파일을 열고 다음 코드를 입력하여 사용자 역할을 정의합니다.

```
user:
  cluster: [ 'monitoring_user','transport_client','kibana_user','snapshot_user' ]
```

```
indices:
  - names: [ '*' ]
    privileges: [ 'all' ]
```

7. Elasticsearch 버전 6.8.6 구성 디렉터리에서 `elasticsearch.yml` 파일을 열고 다음 코드를 입력합니다.

```
xpack.security.authc:
  anonymous:
    username:
    roles:
    authz_exception:
```

8. Elasticsearch를 시작합니다. Elasticsearch를 시작하는 방법에 대한 자세한 내용은 [“Elasticsearch 시작” 페이지 139](#) 항목을 참조하십시오.
9. Elasticsearch 상태 API를 실행하여 Elasticsearch 서버에 HTTPS 프로토콜이 사용되는지 확인합니다. Elasticsearch API를 테스트하는 방법에 대한 자세한 내용은 Elasticsearch 설명서를 참조하십시오.
10. 기존 Elasticsearch 버전 6.2.3 및 버전 6.8.6 환경, MDM Hub 서버 및 Cleanse 서버 서비스를 중지합니다.
11. 다음 디렉터리로 이동합니다.
<elasticsearch 6.8.6 설치 디렉터리>/config
12. `elasticsearch.yml` 파일에서 다음 속성을 구성합니다.

속성	설명
<code>path.data</code>	기존 인덱스를 마이그레이션하려는 경우 Elasticsearch 6.2.3 데이터 디렉터리를 가리킵니다. 여러 데이터 디렉터리를 구성할 수 있습니다. 여러 데이터 디렉터리 구성에 대한 자세한 내용은 Elasticsearch 설명서를 참조하십시오. 참고: Elasticsearch 버전 6.8.6으로 마이그레이션하려면 기존 Elasticsearch 구성 속성에 <code>cluster.name</code> 및 <code>node.name</code> 속성을 포함해야 합니다. 이러한 속성이 없으면 기존 인덱스를 마이그레이션할 수 없습니다.
<code>path.logs</code>	로그 파일의 경로입니다.
<code>http.port</code>	HTTP 요청의 포트입니다. Elasticsearch 6.2.3 환경에서 설정한 것과 동일한 포트 번호를 입력합니다.
<code>transport.tcp.port</code>	TCP 바인딩 포트입니다. Elasticsearch 6.2.3 환경에서 설정한 것과 동일한 TCP 바인딩 포트 번호를 입력합니다.

13. Elasticsearch 버전 6.8.6 환경을 시작합니다.
14. Elasticsearch API를 테스트하여 인덱스가 열리고 오류가 표시되지 않는지 확인합니다. Elasticsearch API를 테스트하는 방법에 대한 자세한 내용은 Elasticsearch 설명서를 참조하십시오.
15. Hub 서버, 처리 서버 및 Elasticsearch 서비스를 시작합니다.
기존 MDM 환경이 Elasticsearch 6.8.6으로 구성되었습니다.

제 11 장

계층 업그레이드

이 장에 포함된 항목:

- [계층 업그레이드 개요, 148](#)
- [계층 관계 및 네트워크 관계 이해, 149](#)
- [계층 관리자 관계 복사 및 계층 관계 생성, 151](#)
- [계층 관리자 관계 복사 및 네트워크 관계 생성, 154](#)

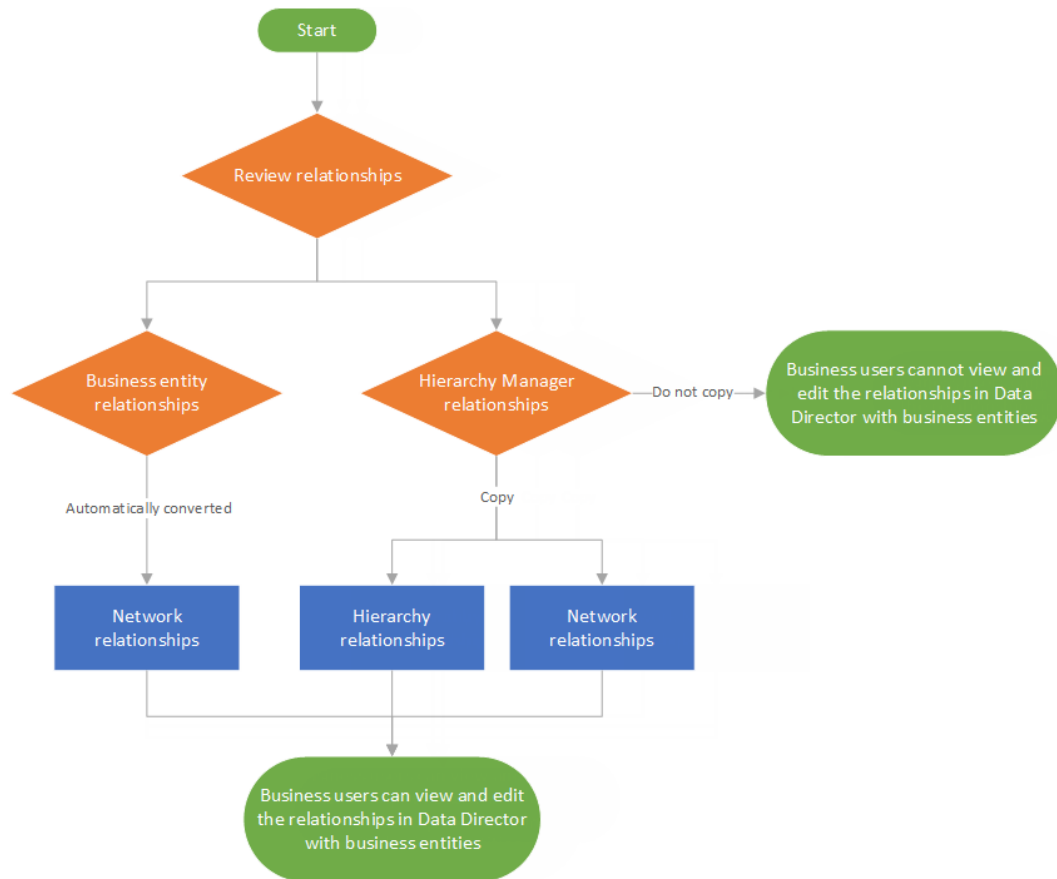
계층 업그레이드 개요

프로비저닝 도구를 사용하여 비즈니스 항목 간에 계층 관계와 네트워크 관계를 생성합니다. 그런 다음 **Data Director**에서 비즈니스 사용자는 **계층** 보기 및 **네트워크** 보기를 사용하여 레코드 간 관계를 생성하고 관리할 수 있습니다.

10.4 이전 버전에서 업그레이드하는 경우 프로비저닝 도구에 비즈니스 항목 관계가 있거나 계층 관리자에 계층 관계가 있을 수 있습니다. 10.4로 업그레이드하면 프로비저닝 도구가 비즈니스 항목 관계를 자동으로 네트워크 관계로 변환합니다.

계층 관리자에 관계가 있는 경우 해당 관계를 검토하고 계속 사용하려는 관계를 결정합니다. 그런 다음 요구 사항에 따라 어떤 관계를 복사하고 계층 관계 또는 네트워크 관계로 생성할지 결정합니다. 프로비저닝 도구를 사용하여 계층 관리자에서 관계를 복사하고 이를 계층 관계 또는 네트워크 관계로 생성합니다.

다음 다이어그램은 이러한 작업을 위한 단계를 안내합니다.



계층 관계 및 네트워크 관계 이해

프로비저닝 도구를 사용하여 계층 관리자에서 관계를 복사하고 이를 계층 관계 또는 네트워크 관계로 생성합니다.

비즈니스 항목 간 계층 관계를 정의하려는 경우에는 관계를 복사한 후 이를 계층 관계로 생성합니다. 관련된 비즈니스 항목 간 연결을 생성하려는 경우에는 관계를 복사한 후 이를 네트워크 관계로 생성합니다.

계층 관리자에서 관계를 복사할 때 생성하는 새 관계는 Hub 콘솔의 동일한 관계 기본 개체를 참조합니다. 사용할 관계에 대한 복사를 완료한 후에는 관계 기본 개체를 기본 개체로 되돌려야 합니다. 관계 기본 개체를 기본 개체로 되돌리면 개체에서 계층 관리자 메타데이터가 제거되고 프로비저닝 도구에서 계층 관계와 네트워크 관계를 생성 및 업데이트할 수 있습니다.

계층 관계

계층 관계는 비즈니스 항목 간 상위-하위 관계입니다. 계층에는 비즈니스 항목 간 계층 관계의 컬렉션이 포함되어 있습니다. 여러 계층을 생성하여 조직에 중요한 계층 관계를 정의할 수 있습니다.

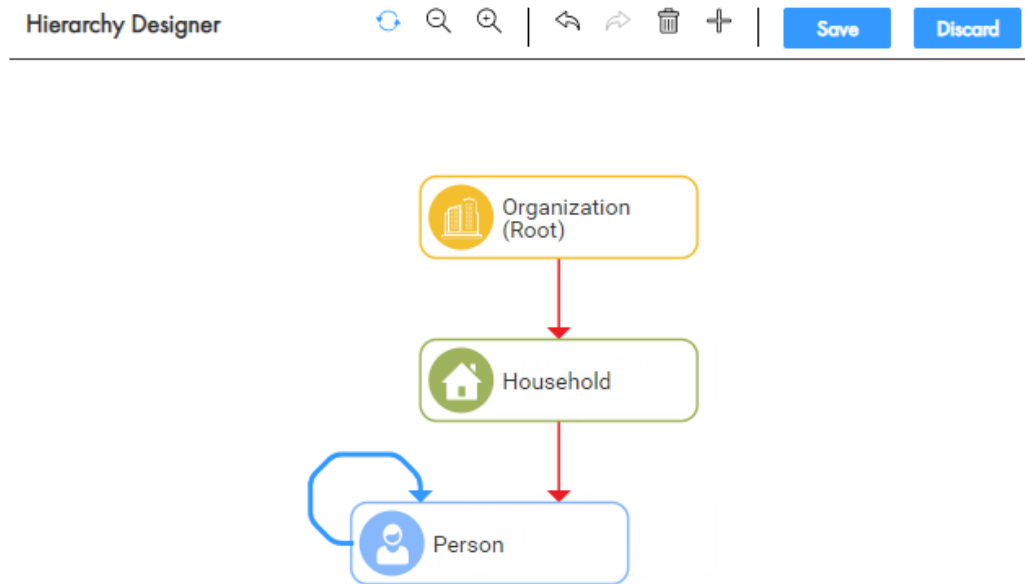
계층 관계를 생성할 때 다음 지침을 고려하십시오.

- 특정 비즈니스 항목을 루트 비즈니스 항목으로 지정해야 합니다.
- 계층의 각 비즈니스 항목과 루트 비즈니스 항목 간의 직접적 또는 간접적인 관계를 정의해야 합니다.

- 한 비즈니스 항목에서 동일한 비즈니스 항목으로의 관계 루프를 생성할 수 있습니다. 예를 들어 **Person** 비즈니스 항목에서 **Person** 비즈니스 항목으로의 관계를 생성할 수 있습니다.

참고: 프로비저닝 도구는 루트 비즈니스 항목에서 관계 루프를 지원하지 않지만 **Data Director**는 루트 비즈니스 항목 수준에서 데이터 루프가 있는 레코드에 대해 계층을 표시할 수 없습니다. 예를 들어 프로비저닝 도구에서 **Organization** 루트 비즈니스 항목에 관계 루프를 생성할 수 있습니다. **Data Director**에서 **Informatica** 레코드와 **Informatica** 레코드 사이에 관계가 있는 경우 **Data Director**는 해당 **Informatica** 레코드에 대한 계층을 표시할 수 없습니다.

다음 이미지는 프로비저닝 도구의 샘플 계층을 보여 줍니다.



네트워크 관계

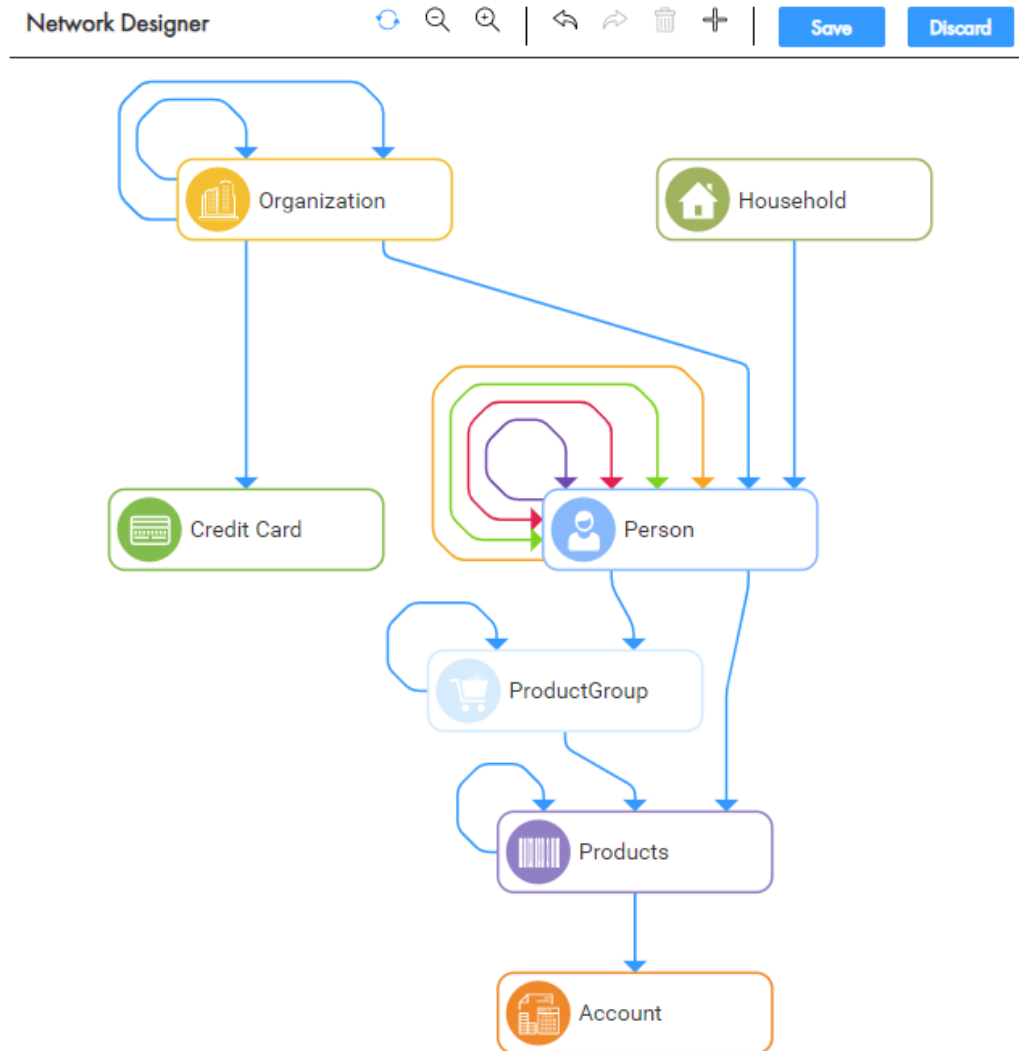
네트워크 관계는 관련된 비즈니스 항목 간의 연결입니다. 생성한 각 네트워크 관계는 네트워크의 일부가 됩니다.

네트워크는 비즈니스 항목 간 네트워크 관계의 컬렉션입니다. 네트워크 관계를 추가하는 네트워크는 하나뿐입니다.

네트워크 관계를 생성할 때 다음 지침을 고려하십시오.

- 루트 비즈니스 항목을 지정하지 않습니다.
- 한 비즈니스 항목에서 동일한 비즈니스 항목으로의 관계 루프를 생성할 수 있습니다.

다음 이미지는 프로비저닝 도구의 샘플 네트워크를 보여 줍니다.



계층 관리자 관계 복사 및 계층 관계 생성

프로비저닝 도구를 사용하여 계층 관리자에서 관계를 복사하고 계층을 생성합니다. 계층 관리자에서 추가 관계를 복사하여 여러 계층을 생성할 수 있습니다.

관계를 복사하여 계층을 생성할 때 프로비저닝 도구는 루트 비즈니스 항목 노드로 구성할 해당 비즈니스 항목을 찾습니다. 프로비저닝 도구는 선택한 **ROOT HM** 항목을 기반으로 루트 비즈니스 항목을 찾으려고 시도합니다. 해당하는 비즈니스 항목을 찾을 수 없는 경우 비즈니스 항목을 생성합니다. 그런 다음 프로비저닝 도구는 복사된 관계를 기반으로 나머지 계층 모델을 생성합니다. 계층과 이 프로세스에서 생성된 계층 관계를 검토해야 합니다.

계층 관리자에서 관계를 복사한 다음 계층을 생성하려면 다음 태스크를 수행합니다.

1. 계층 관리자에서 관계 복사 및 계층 생성.
2. 계층 REST API 지원을 위해 **Hub** 서버 구성.
3. 계층에 대한 액세스 권한 구성.

4. 관계 기본 개체를 기본 개체로 되돌리기.

계층 관리자에서 관계 복사 및 계층 생성

계층 관리자에서 관계를 복사하고 계층 관계가 포함된 계층을 생성합니다. 계층 관리자에서 추가 계층 관계를 복사하여 추가 계층을 생성할 수 있습니다.

중요: 관계를 복사하여 계층을 생성할 때 프로비저닝 도구는 루트 비즈니스 항목 노드로 구성할 해당 비즈니스 항목을 찾습니다. 프로비저닝 도구는 선택한 **ROOT HM** 항목을 기반으로 루트 비즈니스 항목을 찾으려고 시도합니다. 해당하는 비즈니스 항목을 찾을 수 없는 경우 비즈니스 항목을 생성합니다. 그런 다음 프로비저닝 도구는 복사된 관계를 기반으로 나머지 계층 모델을 생성합니다. 계층과 관계를 복사하여 생성된 계층 관계를 검토해야 합니다.

1. **비즈니스 항목 > 모델링**을 클릭한 다음 **계층**을 선택합니다.
2. **생성 > 계층 관리자에서 복사된 계층**을 클릭합니다.
계층 관리자에서 복사된 계층 생성 창이 나타납니다.
3. **기존 계층 선택** 열에서 복사할 계층을 선택합니다.
4. **루트 항목 선택** 열에서 다음 루트 항목 설정을 구성합니다.
 - a. **ROOT HM 항목** 필드에서 루트 노드로 사용할 **HM** 항목을 선택합니다.
프로비저닝 도구는 **ROOT HM** 항목을 사용하여 계층에 대한 루트 비즈니스 항목을 결정합니다.
 - b. **새 계층 이름** 필드에서 계층에 대한 이름을 입력합니다.
5. **새 계층 생성**을 클릭합니다.
계층 디자이너가 열립니다.
6. 비즈니스 항목 및 관계를 검토하고 문제를 기반으로 문제를 해결합니다.
 - 루트 비즈니스 항목에 관계 루프가 있는 경우 관계 삭제를 통해 관계 루프를 제거하거나 계층을 편집하여 새 비즈니스 항목을 루트 비즈니스 항목으로 지정합니다.
경고: 프로비저닝 도구는 루트 비즈니스 항목에서 관계 루프를 지원하지만 **Data Director**는 루트 비즈니스 항목 수준에서 데이터 루프가 있는 레코드에 대해 계층을 표시할 수 없습니다. 계층 생성을 위한 지침에 대한 자세한 내용은 [“계층 관계” 페이지 149](#) 항목을 참조하십시오.
 - 오류가 있는 관계가 있는 경우 관계를 선택하고 관계 속성을 검토합니다. 오류가 있는 관계는 빨간색 줄로 나타납니다.
 - 오류가 있는 비즈니스 항목 노드가 있는 경우 노드를 선택하고 비즈니스 항목을 검토합니다. 오류가 있는 비즈니스 항목 노드가 빨간색 테두리와 함께 나타납니다.
7. **저장**을 클릭합니다.

계층에 대한 Hub 서버 구성

Hub 콘솔의 Hub 서버 도구와 `cmxserver.properties` 파일을 사용하여 계층 REST API를 지원하도록 Hub 서버 속성을 구성합니다.

계층 REST API를 사용하여 계층과 상호 작용할 수 있습니다. 계층 REST API에 대한 자세한 내용은 *Multidomain MDM 비즈니스 항목 서비스 가이드*를 참조하십시오.

1. 텍스트 편집기를 사용하여 다음 위치의 `cmxserver.properties` 파일을 엽니다. <MDM Hub 설치 디렉터리>\hub\server\resources\cmxserver.properties

2. 다음 속성을 수동으로 추가하고 구성합니다.

`cmx.server.hierarchy.max-search-depth`

계층 REST API를 사용하여 계층 경로를 찾을 때 검색되는 최대 깊이입니다. 기본값은 100입니다.

`cmx.server.hierarchy.max-search-width`

계층 REST API를 사용하여 내보낼 때 포함할 검색된 계층의 최대 너비입니다. 기본값은 1000000입니다.

관계 기본 개체를 기본 개체로 변환

관계 기본 개체를 기본 개체로 되돌리면 관계 개체에서 계층 관리자 메타데이터가 제거됩니다. 관계 개체는 기본 개체로 남아 있지만 계층 관리자에는 기본 개체가 표시되지 않습니다.

되돌리려는 관계-유형 열이 조화를 위한 준비 테이블에 있는 경우 관계 기본 개체를 되돌리려면 준비 테이블 열이 비어 있어야 합니다.

계층 관리자 관계를 업그레이드 중이라면 관계 기본 개체를 되돌리기 전에 계층 관리자의 관계를 프로비저닝 도구에 복사합니다.

1. 계층 도구에서 쓰기 잠금을 획득합니다.
2. 관계 기본 개체를 마우스 오른쪽 단추로 클릭하고 **항목/관계 개체를 BO로 되돌리기**를 선택합니다.
3. **항목/관계 개체 되돌리기** 대화 상자에서 **확인**을 클릭합니다.
항목이 되돌려지면 대화 상자가 나타납니다.

계층에 대한 액세스 권한 구성

계층을 보안 리소스로 구성하고 MDM 역할에 계층에 대한 액세스 권한을 부여합니다.

1. Hub 콘솔을 열고 관리할 데이터베이스를 선택합니다.
2. **쓰기 잠금 > 잠금 획득**을 클릭합니다.
3. 계층을 보안으로 구성합니다.
 - a. **작업 영역** 패널의 **보안 액세스 관리자**에서 **보안 리소스**를 클릭합니다.
보안 리소스 패널이 표시됩니다.
 - b. **리소스** 탭에서 **비즈니스 항목 계층** 노드를 확장합니다.
 - c. 보안으로 구성할 비즈니스 항목 계층을 선택한 다음 **리소스 상태를 보안으로 변경** 아이콘을 클릭합니다.
 - d. **저장** 아이콘을 클릭합니다.
4. MDM 역할에 계층에 대한 액세스 권한을 부여합니다.
 - a. **작업 영역** 패널의 **보안 액세스 관리자**에서 **역할**을 클릭합니다.
역할 패널이 나타납니다.
 - b. **DataSteward**를 선택합니다.
 - c. **리소스 권한** 탭에서 **비즈니스 항목 계층** 노드를 확장합니다.
 - d. 각 계층에 대해 **읽기**, **생성** 및 **업데이트** 확인란을 선택합니다.
 - e. 관리자 및 SrManager 역할에 대해 **c-d** 단계를 반복합니다.
 - f. **저장** 아이콘을 클릭합니다.
5. 리포지토리의 유효성을 검사합니다.
 - a. **작업 영역** 패널의 **구성**에서 **Repository Manager**를 클릭합니다.

- b. 마스터 데이터베이스에 연결을 클릭합니다.
- c. 유효성 검사 탭에서 계층 액세스 권한을 구성한 데이터베이스를 선택합니다.
- d. 유효성 검사 아이콘을 클릭합니다.
- e. 확인을 클릭합니다.

계층 관리자 관계 복사 및 네트워크 관계 생성

프로비저닝 도구를 사용하여 계층 관리자에서 관계를 복사하고 이를 네트워크 관계로 생성합니다.

계층 관리자에서 관계를 복사한 후 이를 네트워크 관계로 생성하려면 다음 태스크를 수행합니다.

1. 계층 관리자에서 관계 복사 및 네트워크 관계 생성.
2. 관계 기본 개체를 기본 개체로 되돌리기.

계층 관리자에서 관계를 복사하여 네트워크 관계 추가

계층 관리자에서 관계를 복사하여 관계를 네트워크 관계로 추가합니다. 계층 관리자의 다른 계층에서 관계를 복사하여 네트워크 관계를 더 추가할 수 있습니다.

1. **비즈니스 항목 > 모델링**을 클릭한 다음 **네트워크**를 선택합니다.
2. **생성 > 계층 관리자에서 복사된 관계**를 클릭합니다.
계층 관리자에서 복사된 관계 추가 창이 나타납니다.
3. **기존 계층 선택** 열에서 관계를 복사할 계층을 선택합니다.
4. **관계 선택** 열에서 복사할 관계를 선택합니다.
5. **선택한 항목을 네트워크에 추가**를 클릭합니다.
네트워크 디자이너가 열립니다.
6. 네트워크에서 관계를 검토합니다. 오류가 있는 관계가 있는 경우 관계를 선택하고 관계 속성을 검토합니다.
오류가 있는 관계는 빨간색 줄로 나타납니다.
7. **저장**을 클릭합니다.

관계 기본 개체를 기본 개체로 변환

관계 기본 개체를 기본 개체로 되돌리면 관계 개체에서 계층 관리자 메타데이터가 제거됩니다. 관계 개체는 기본 개체로 남아 있지만 계층 관리자에는 기본 개체가 표시되지 않습니다.

되돌리려는 관계-유형 열이 조회를 위한 준비 테이블에 있는 경우 관계 기본 개체를 되돌리려면 준비 테이블 열이 비어 있어야 합니다.

계층 관리자 관계를 업그레이드 중이라면 관계 기본 개체를 되돌리기 전에 계층 관리자의 관계를 프로비저닝 도구에 복사합니다.

1. 계층 도구에서 쓰기 잠금을 획득합니다.
2. 관계 기본 개체를 마우스 오른쪽 단추로 클릭하고 **항목/관계 개체를 BO로 되돌리기**를 선택합니다.
3. **항목/관계 개체 되돌리기** 대화 상자에서 **확인**을 클릭합니다.
항목이 되돌려지면 대화 상자가 나타납니다.

제 12 장

응용 프로그램 서버에 대한 ActiveVOS 설치 후 태스크

이 장에 포함된 항목:

- [응용 프로그램 서버에 대한 ActiveVOS 설치 후 태스크, 155](#)

응용 프로그램 서버에 대한 ActiveVOS 설치 후 태스크

클린 업그레이드 또는 인플레이스 업그레이드를 수행하는 경우 응용 프로그램 서버에 대한 ActiveVOS 설치 후 태스크를 수행하여 환경이 올바르게 구성되었는지 확인해야 합니다.

응용 프로그램 서버에 대한 ActiveVOS 설치 후 태스크는 환경에 해당하는 *Multidomain MDM 설치 가이드*에서 응용 프로그램 서버에 대한 ActiveVOS 설치 후 태스크 장을 참조하십시오.

제 13 장

비즈니스 항목 어댑터에 대한 ActiveVOS 업그레이드 후 태스크

이 장에 포함된 항목:

- [비즈니스 항목 어댑터에 대한 ActiveVOS 업그레이드 후 태스크, 156](#)
- [비즈니스 항목 워크플로우 어댑터의 ActiveVOS URN 구성, 157](#)
- [ActiveVOS 프로토콜을 HTTPS로 설정, 157](#)
- [비즈니스 항목에 대한 사용자 지정된 워크플로우 업데이트, 158](#)
- [ActiveVOS의 MDM ID 서비스 구성, 161](#)
- [사용자 지정 BeMDMWorkflow 프로젝트\(인플레이스 업그레이드\), 162](#)
- [병합 해제 및 병합 워크플로우 트리거 구성\(인플레이스 업그레이드\), 162](#)
- [Entity 360 프레임워크 태스크 관리자 추가, 162](#)

비즈니스 항목 어댑터에 대한 ActiveVOS 업그레이드 후 태스크

클린 업그레이드 또는 인플레이스 업그레이드 수행 시 비즈니스 항목에 기반한 ActiveVOS 워크플로우 어댑터를 사용하는 경우 비즈니스 항목 어댑터에 대한 ActiveVOS 업그레이드 후 태스크를 수행하여 환경이 올바르게 구성되었는지 확인해야 합니다.

비즈니스 항목 워크플로우 어댑터의 ActiveVOS URN 구성

ActiveVOS Server에는 내부적으로 사용되는 두 개의 미리 정의된 URN(Uniform resource name)이 있습니다. ActiveVOS Server가 실행되는 호스트 이름과 포트 번호를 사용하려면 URN 매핑에서 URL을 업데이트해야 합니다.

1. ActiveVOS Console을 시작합니다. 브라우저에서 올바른 호스트 이름과 포트 번호 대신 다음 URL을 입력합니다.

암호화된 연결. `https://[host]:[port]/activevos`

암호화되지 않은 연결. `http://[host]:[port]/activevos`

2. ActiveVOS Console의 홈 페이지에서 **관리 > 서버 구성 > URN 매핑**을 클릭합니다.
3. 다음 URN에서 ActiveVOS 서버의 호스트 이름과 포트 번호를 반영하도록 경로를 업데이트합니다.

URN	URL 경로
ae:internal-reporting	암호화된 연결. <code>https://[host]:[port]/activevos/internalreports</code> 암호화되지 않은 연결. <code>http://[host]:[port]/activevos/internalreports</code>
ae:task-inbox	암호화된 연결. <code>https://[host]:[port]/activevos-central/avc</code> 암호화되지 않은 연결. <code>http://[host]:[port]/activevos-central/avc</code>

4. **urn:mdm:service**가 MDM Hub 서버의 호스트 이름 및 포트 번호에 매핑되는지 확인합니다.

암호화된 연결. `https://[host]:[port]/cmx/services/BeServices`

암호화되지 않은 연결. `http://[host]:[port]/cmx/services/BeServices`

ActiveVOS 프로토콜을 HTTPS로 설정

ActiveVOS와 MDM Hub 간에 보안 통신을 활성화하려면 Hub 콘솔 워크플로우 관리자에서 프로토콜을 HTTPS로 설정합니다.

먼저 HTTPS 통신을 사용하도록 응용 프로그램 서버를 구성해야 합니다.

1. Hub 콘솔을 시작합니다.
2. 쓰기 잠금을 획득합니다.
3. 구성 작업 영역 아래에서 **워크플로우 관리자**를 클릭합니다.
4. 워크플로우 관리자에서 **워크플로우 엔진** 탭을 클릭합니다.
5. ActiveVOS 워크플로우 엔진을 선택한 다음 **편집** 단추를 클릭합니다.
6. 워크플로우 편집 대화 상자에서 프로토콜을 HTTPS로 설정합니다.
7. WebLogic 환경의 워크플로우 편집 대화 상자에서 **abAdmin** 역할에 속하는 사용자의 사용자 이름 및 암호를 입력합니다.

비즈니스 항목에 대한 사용자 지정된 워크플로우 업데이트

비즈니스 항목 어댑터에서 사용할 워크플로우를 사용자 지정할 경우 변경 내용 목록을 검토하고 사용자 지정된 워크플로우를 필요에 따라 업데이트해야 합니다.

다음 테이블에는 릴리스에 따라 기본 워크플로우에 적용된 변경 내용이 나열되어 있습니다.

릴리스	변경	지시 사항
10 HotFix 1	암호 암호화를 활성화합니다.	각 워크플로우에 대해 <code>sif:encrypted</code> 를 <code>true</code> 로 설정합니다.
10 HotFix 1	태스크 받은 편지함에 대한 프레젠테이션 매개 변수를 업데이트합니다.	“비즈니스 항목에 대한 워크플로우에서 프레젠테이션 매개 변수 업데이트” 페이지 158
10.3	파일 첨부 기능을 활성화합니다.	“비즈니스 항목에 대한 워크플로우에서 첨부 파일 활성화” 페이지 160

비즈니스 항목에 대한 워크플로우에서 프레젠테이션 매개 변수 업데이트

비즈니스 항목을 기반으로 워크플로우를 사용자 지정할 경우 `.bpel` 파일을 열고 프레젠테이션 매개 변수를 다음 매개 변수 목록과 비교합니다. 필요한 경우 각 태스크에 대해 프레젠테이션 매개 변수를 추가 또는 업데이트합니다.

식에 매개 변수에 대한 절대 참조가 포함될 수 있지만 사용자 지정된 워크플로우의 모범 사례는 태스크에 대한 `getInput()` 메서드를 가리키는 것입니다. 프레젠테이션 매개 변수에 대한 절대 참조를 `getInput()` 메서드에 대한 호출로 바꾸는 것을 고려하십시오. 예를 들어 `$ProcessTaskRequest/mdmavxsd:INFATask/htd:getInput('processTaskRequest')/mdmavxsd:INFATask/`로 바꿀 수 있습니다. 다음 테이블의 식은 입력 메서드를 호출합니다.

팁: 식에서는 네임스페이스 접두사 `mdmavxsd`를 사용합니다. 사용자 지정 워크플로우를 업데이트한 후 문제가 발생하면 `mdmavxsd`가 `urn:informatica.mdm.av.xsd`에 대한 접두사로 정의되었는지 확인합니다.

다음 테이블에는 프레젠테이션 매개 변수와 식이 나열되어 있습니다.

매개 변수	식
hubUsername	<code>htd:getInput('processTaskRequest')/mdmavxsd:INFATask/mdmavxsd:hubUsername/text()</code>
hubPassword	<code>htd:getInput('processTaskRequest')/mdmavxsd:INFATask/mdmavxsd:hubPassword/text()</code>
securityPayload	<code>htd:getInput('processTaskRequest')/mdmavxsd:INFATask/mdmavxsd:securityPayload/text()</code>
orsId	<code>htd:getInput('processTaskRequest')/mdmavxsd:INFATask/mdmavxsd:orsId/text()</code>
taskTypeName	<code>htd:getInput('processTaskRequest')/mdmavxsd:INFATask/mdmavxsd:taskType/mdmavxsd:name/text()</code>
taskTypeDisplayName	<code>htd:getInput('processTaskRequest')/mdmavxsd:INFATask/mdmavxsd:taskType/mdmavxsd:displayName/text()</code>

매개 변수	식
taskTypeDescription	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskType/mdmavxsd:description/text()
pendingBVT	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskType/mdmavxsd:pendingBVT
taskTypeDataUpdateType	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskType/mdmavxsd:dataUpdateType/text()
taskTypeDisplayType	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskType/mdmavxsd:displayType/text()
defaultApproval	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskType/mdmavxsd:defaultApproval
taskDataTaskId	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskData/mdmavxsd:taskId/text()
taskDataOwnerUID	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskData/mdmavxsd:ownerUID/text()
taskDataGroups	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskData/mdmavxsd:groups/mdmavxsd:groups/text()
dueDate	let \$in := htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskData/mdmavxsd:dueDate/text() let \$out :=
상태	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskData/mdmavxsd:status/text()
taskDataPriority	length(\$out > 0))), string-length(\$out) + (string-length(\$in) * xsd:int((string- length(\$out) = 0)))
taskDataSubjectAreaUID	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskData/mdmavxsd:subjectAreaUID/text()
taskDataTitle	let \$in := htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskData/mdmavxsd:title/text() let \$out :=
taskDataComments	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskData/mdmavxsd:comments/text()
taskDataInteractionId	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskData/mdmavxsd:interactionId/text()
taskDataCreator	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskData/mdmavxsd:creator/text()
createDate	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskData/mdmavxsd:createDate
taskDataUpdatedBy	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskData/mdmavxsd:updatedBy/text()

매개 변수	식
lastUpdateDate	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskData/mdmavxsd:lastUpdateDate
workflowVersion	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:workflowVersion/text()
beRowId	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskData/mdmavxsd:taskRecords/ mdmavxsd:INFARecordKey[1]/mdmavxsd:rowId/text()
bePkeySrcObject	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskData/mdmavxsd:taskRecords/ mdmavxsd:INFARecordKey[1]/mdmavxsd:pkeySrcObject/text()
beSystem	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskData/mdmavxsd:taskRecords/ mdmavxsd:INFARecordKey[1]/mdmavxsd:system/text()
beRowidXref	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskData/mdmavxsd:taskRecords/ mdmavxsd:INFARecordKey[1]/mdmavxsd:rowidXref/text()
beTableUID	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskData/mdmavxsd:taskRecords/ mdmavxsd:INFARecordKey[1]/mdmavxsd:tableUID/text()
taskTypeCreationType	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskType/mdmavxsd:creationType/text()

.bpel 파일 업데이트에 대한 자세한 내용은 **ActiveVOS** 설명서를 참조하십시오.

비즈니스 항목에 대한 워크플로우에서 첨부 파일 활성화

버전 10.3에서는 사용자가 태스크에 파일을 첨부하는 것을 허용하도록 사용자 지정된 워크플로우를 업데이트할 수 있습니다. 사용자 지정된 워크플로우에서 이 기능을 활성화하려면 데이터를 초기화하는 프로세스를 업데이트하고 모든 사용자 프로세스를 업데이트합니다.

참고: 첨부 파일 기능을 일부 태스크로 제한하려는 경우 프로비저닝 도구에서 태스크 설정을 구성할 수 있습니다. 태스크 구성 업데이트에 대한 자세한 내용은 **Multidomain MDM 프로비저닝 도구 가이드** 항목을 참조하십시오.

1. **ActiveVOS Designer**에서 사용자 지정된 워크플로우의 .bpel 파일을 엽니다.
워크플로우가 캔버스에 나타납니다.
2. 데이터를 초기화하는 프로세스를 업데이트합니다.
 - a. **데이터 초기화** 프로세스를 선택합니다.
작업 대화 상자가 표시됩니다.
 - b. **새 스크립트**를 클릭합니다.
스크립트 작성기 대화 상자가 나타납니다.

- c. 스크립트 상자에 다음 문을 추가합니다.
`abx:copyAllAttachments('StartRequest', '_peopleActivityAttachments')`
`StartRequest`를 워크플로우의 데이터를 초기화할 때 사용한 변수 이름으로 바꿉니다.
 - d. 확인을 클릭합니다.
스크립트 작성기 대화 상자가 닫히고 스크립트가 **작업** 목록에 나타납니다.
 - e. 확인을 클릭합니다.
작업 대화 상자가 닫힙니다.
3. 모든 사용자 프로세스를 업데이트합니다.
 - a. 사용자 프로세스를 선택합니다.
 - b. **속성** 탭을 클릭합니다.
 - c. **모두** 탭을 클릭합니다.
 - d. **첨부 파일 전파**를 확장합니다.
 - e. **시작 프로세스**를 **모두**로 설정합니다.
 - f. **종료 프로세스**를 **newOnly**로 설정합니다.
 4. .bpel 파일을 저장합니다.

ActiveVOS의 MDM ID 서비스 구성

포함된 ActiveVOS를 사용하는 경우에는 MDM ID 서비스를 사용하도록 ActiveVOS를 구성해야 합니다. ActiveVOS에 대해 MDM ID 서비스를 구성하려면 ActiveVOS 콘솔을 사용하여 ID 서비스 암호를 MDM Hub 워크플로우 엔진 사용자의 암호로 설정합니다.

1. ActiveVOS 콘솔에서 **관리 > 서비스 구성 > ID 서비스**를 선택합니다.
2. 공급자 구성 섹션에서 **활성화** 확인란을 활성화하고 **공급자 유형** 목록에서 **MDM**을 선택합니다.
3. 연결 탭에서 사용자 이름이 **admin**인 MDM Hub 사용자의 암호를 입력합니다.
참고: 나중에 관리 사용자의 암호를 변경할 경우에는 ActiveVOS ID 서비스 설정에 새 암호를 입력해야 합니다.
4. **업데이트**를 클릭합니다.
5. ActiveVOS가 MDM Hub에 admin사용자로 로그인할 수 있는지 그리고 **테스트할 사용자**로 지정된 사용자의 역할 목록을 검색할 수 있는지 테스트합니다.
 - a. **테스트** 탭을 선택합니다.
 - b. **테스트할 사용자** 필드에 역할에 할당된 MDM Hub 사용자를 입력합니다.
 - c. **테스트 설정**을 클릭합니다.
참고: 연산 참조 저장소가 구성되지 않았거나, 테스트 사용자가 역할에 속하지 않거나, 역할 이름에 공백이 포함되는 경우 테스트가 실패합니다.

사용자 지정 BeMDMWorkflow 프로젝트(인플레이스 업그레이드)

BeMDMWorkflow 프로젝트의 워크플로우를 사용자 지정한 경우에는 프로젝트를 업데이트한 후 재배포해야 합니다.

1. Informatica ActiveVOS Designer에서 프로젝트 탐색기를 사용하여 **BeCommonMDM > wsdl > cs.wsdl**로 이동합니다.
2. **cs.wsdl**을 마우스 오른쪽 단추로 클릭한 다음 **연결 프로그램 > 텍스트 편집기**를 선택합니다.
3. 텍스트 편집기에서 다음 코드로 이동합니다.

```
<xsd:complexType name="TaskFilter">
  <xsd:sequence>
```
4. 다음 코드를 추가합니다.

```
<xsd:element minOccurs="0" name="overdueOnly" type="xsd:boolean" />
```
5. Informatica ActiveVOS Designer에서 내보내려는 BeCommonMDM 프로젝트를 엽니다.
6. **파일 > 내보내기**를 클릭합니다.
내보내기 대화 상자가 열립니다.
7. **Orchestration** 아래에서 **기여 - 비즈니스 프로세스 보관**을 선택합니다. **다음**을 클릭합니다.
8. **서버 배포 옵션** 아래의 **배포 URL** 필드에 ActiveVOS 인스턴스의 URL을 입력합니다. **마침**을 클릭합니다.
9. **배포 완료** 대화 상자에서 **확인**을 클릭합니다.

병합 해제 및 병합 워크플로우 트리거 구성(인플레이스 업그레이드)

Multidomain MDM 버전 10.2에서 도입된 병합 해제 및 병합 워크플로우 트리거를 구성해야 합니다. 병합 해제 및 병합 워크플로우 트리거를 구성하려면 프로비저닝 도구를 사용합니다.

자세한 내용은 *Multidomain MDM 프로비저닝 도구 가이드*를 참조하십시오.

Entity 360 프레임워크 태스크 관리자 추가

비즈니스 항목 ActiveVOS 워크플로우 어댑터를 사용하는 경우 Entity 360 프레임워크 태스크 관리자 및 Entity 360 프레임워크 태스크 받은 편지함을 사용합니다.

레거시 태스크 받은 편지함을 대체하려면 Informatica Data Director 구성을 업데이트합니다. Entity 360 프레임워크 태스크 받은 편지함을 **홈** 페이지에 추가할 수 있습니다. Informatica Data Director 사용자 인터페이스 설계에 대한 자세한 내용은 *Multidomain MDM 프로비저닝 도구 가이드*를 참조하십시오.

제 14 장

제목 영역 어댑터에 대한 ActiveVOS 업그레이드 후 태스크

이 장에 포함된 항목:

- [제목 영역 어댑터에 대한 ActiveVOS 업그레이드 후 태스크, 163](#)
- [ActiveVOS URN 업데이트, 164](#)
- [ActiveVOS의 트러스트된 사용자 확인, 164](#)
- [제목 영역을 기반으로 ActiveVOS 워크플로우에 대한 Informatica Data Director 태스크 구성 업데이트, 165](#)
- [제목 영역에 대한 사용자 지정된 워크플로우 업데이트, 167](#)
- [제목 영역을 기반으로 하는 ActiveVOS 워크플로우 재배포, 169](#)
- [비즈니스 항목 및 비즈니스 항목 서비스 구성 파일 생성, 170](#)

제목 영역 어댑터에 대한 ActiveVOS 업그레이드 후 태스크

클린 업그레이드 또는 인플레이스 업그레이드 수행 시 제목 영역에 기반한 ActiveVOS 워크플로우 어댑터를 사용하는 경우 제목 영역 어댑터에 대한 ActiveVOS 업그레이드 후 태스크를 수행하여 환경이 올바르게 구성되었는지 확인해야 합니다.

ActiveVOS URN 업데이트

MDM Hub와 ActiveVOS 사이의 보안 통신을 위해 HTTPS(HTTP Secure) 프로토콜을 사용하려면 URN 경로의 URL을 http에서 https로 변경합니다.

1. ActiveVOS Console을 시작합니다. 브라우저에서 올바른 호스트 이름과 포트 번호 대신 다음 URL을 입력합니다.

JBoss 버전 7.2 또는 7.1	URL 경로
보안 연결	https://<호스트>:<포트>/activevos
비보안 연결	http://<호스트>:<포트>/activevos

2. ActiveVOS Console의 홈 페이지에서 **관리 > 서버 구성 > URN 매핑**을 클릭합니다.
3. 다음 URN에서 ActiveVOS 서버의 호스트 이름과 포트 번호를 반영하도록 경로를 업데이트합니다.

URN	URL 경로
ae:internal-reporting	보안 연결. https://<호스트>:<포트>/activevos/internalreports 비보안 연결. http://<호스트>:<포트>/activevos/internalreports
ae:task-inbox	보안 연결. https://<호스트>:<포트>/activevos-central/avc 비보안 연결. http://<호스트>:<포트>/activevos-central/avc

4. **MDMHost:InfamDM**이 MDM Hub 서버의 호스트 이름 및 포트 번호에 매핑되는지 확인합니다.

보안 연결.https://<호스트>:<포트>/cmx/services/SifService

비보안 연결.http://<호스트>:<포트>/cmx/services/SifService

ActiveVOS의 트러스트된 사용자 확인

Hub 콘솔에서 ActiveVOS 워크플로우 엔진 설정에 트러스트된 사용자가 지정되었는지 확인합니다.

1. Hub 콘솔의 구성 작업 영역에서 **워크플로우 관리자**를 클릭합니다.
2. **워크플로우 엔진** 탭을 선택합니다.
3. 쓰기 잠금을 획득합니다.
4. **ActiveVOS**를 선택하고 **편집** 단추를 클릭합니다.
5. 워크플로우 편집 대화 상자에서 트러스트된 사용자의 사용자 이름과 암호를 입력합니다.
6. **확인**을 클릭합니다.

제목 영역을 기반으로 ActiveVOS 워크플로우에 대한 Informatica Data Director 태스크 구성 업데이트

태스크 관리자와 함께 제목 영역 기반 ActiveVOS 워크플로우 어댑터를 사용하려면 Informatica Data Director 구성 파일을 업데이트해야 합니다. 제목 영역을 기반으로 하는 ActiveVOS 워크플로우를 사용하는 경우에는 비즈니스 항목을 기반으로 하는 ActiveVOS 워크플로우로 마이그레이션할 수 없습니다.

Informatica Data Director 구성 파일에서 다음과 같은 태스크 매개 변수를 구성할 수 있습니다.

taskType

태스크 유형을 설명합니다.

taskId

프로세스 이름입니다.

name

taskType 이름입니다. 이 이름은 ActiveVOS 워크플로우 구성의 태스크 이름과 동일해야 합니다.

제목 영역 기반 ActiveVOS 어댑터에 대한 IDD 구성 업데이트

제목 영역 기반 ActiveVOS 워크플로우 어댑터를 비즈니스 항목 기반 태스크 관리자와 사용하려면 Data Director 구성 파일을 업데이트합니다. Data Director 구성 파일을 업데이트하지 않는 경우 태스크 관리자를 사용하여 태스크를 생성할 수 없습니다.

다음 코드 샘플은 Multidomain MDM에서 제공되는 워크플로우의 Data Director 구성 파일에서 제목 영역 기반 ActiveVOS 태스크를 구성하는 방법을 보여 줍니다.

```
<tasks includeUnassignedTasks="true">
<!-- Task Definitions -->
<taskType taskId="IDDMergeTask" name="AVOSMerge" displayName="Merge" creationType="MERGE"
displayType="MERGE">
<description>Merge two records together.</description>
</taskType>

<taskType taskId="IDDUnmergeTask" name="AVOSUnmerge" displayName="Unmerge" creationType="UNMERGE"
displayType="UNMERGE">
<description>Unmerge an XREF record from a Base Object record.
</description>
</taskType>

<taskType taskId="IDDOneStepApprovalTask" name="AVOSFinalReview" displayName="Final review" creationType="NONE" pendingBVT="true">
<description>Update a record and require the user to go through an approval process before completing
the task.
</description>
</taskType>

<taskType name="Notification" displayName="Notification" creationType="NONE" displayType="NORMAL">
<description>Notification step in the workflow</description>
</taskType>

<taskType taskId="IDDTwoStepApprovalTask" name="AVOSReviewNoApprove"
displayType="NORMAL" displayName="Review no approve" creationType="NONE"
defaultApproval="true" pendingBVT="true">
<description>Update a record and require the user to go through an approval process
before completing the task.
</description>
</taskType>

<taskType taskId="IDDUpdateWithApprovalTask" name="Update" displayType="NORMAL" displayName="Update"
creationType="CREATE" pendingBVT="true">
<description>Update a record and do not require the user to go through an approval
```

```

        process before completing the task. The approval step is optional.
    </description>
</taskType>

</tasks>

```

제목 영역 워크플로우 어댑터의 태스크 트리거 구성

태스크 관리자와 함께 제목 영역을 기반으로 ActiveVOS 워크플로우를 사용하도록 태스크 트리거를 구성해야 합니다. 태스크 트리거를 구성하지 않으면 태스크가 태스크 관리자에 표시되지 않습니다.

트리거를 구성하려면 프로비저닝 도구를 사용하여 고급 구성 페이지에서 태스크 구성 파일을 편집합니다. 자세한 내용은 *Multidomain MDM 프로비저닝 도구 가이드*를 참조하십시오.

다음 startWorkflow 특성을 구성하여 태스크 트리거를 구성할 수 있습니다.

process

ActiveVOS 워크플로우 프로세스의 이름입니다.

taskKind

프로세스에 필요한 사용자 인터페이스의 유형을 정의합니다. REVIEW, MERGE 또는 UNMERGE가 될 수 있습니다. taskKind는 ActiveVOS 워크플로우 엔진에 의해 반환됩니다.

taskTemplate

사용할 태스크 템플릿의 이름입니다.

firstTask Type

워크플로우의 첫 번째 태스크입니다. 선택 사항입니다. 이 매개 변수는 태스크가 작성될 때 태스크가 할당되도록 허용합니다.

2단계 승인 코드 샘플

다음 코드 샘플은 2단계 승인 태스크에 대해 제목 영역 기반 ActiveVOS 어댑터의 startWorkflow 요소 구성을 보여 줍니다.

```

<trigger name="DefaultApproval">
  <startWorkflow process="IDDTwoStepApprovalTask" taskKind="REVIEW" taskTemplate="DefaultApproval"
    firstTaskType="AVOSReviewNoApprove"/>
  <event name="CreateBE"/>
  <event name="UpdateBE"/>
  <role name="*/>
</trigger>

```

1단계 승인 코드 샘플

다음 코드 샘플은 1단계 승인 태스크에 대해 제목 영역 기반 ActiveVOS 어댑터의 startWorkflow 요소 구성을 보여 줍니다.

```

<trigger name="DefaultApproval">
  <startWorkflow process="IDDOneStepApprovalTask" taskKind="REVIEW" taskTemplate="DefaultApproval"
    firstTaskType="AVOSFinalReview"/>
  <event name="CreateBE"/>
  <event name="UpdateBE"/>
  <role name="*/>
</trigger>

```

승인과 함께 업데이트 코드 샘플

다음 코드 샘플은 승인과 함께 업데이트 태스크에 대해 제목 영역 기반 ActiveVOS 어댑터의 startWorkflow 요소 구성을 보여 줍니다.

```

<trigger name="DefaultApproval">
  <startWorkflow process="IDDUUpdateWithApprovalTask" taskKind="REVIEW" taskTemplate="DefaultApproval"
    firstTaskType="Update"/>

```

```
<event name="CreateBE"/>
<event name="UpdateBE"/>
<role name="*" />
</trigger>
```

병합 코드 샘플

다음 코드 샘플은 병합 태스크에 대해 제목 영역 기반 ActiveVOS 어댑터의 startWorkflow 요소 구성을 보여 줍니다.

```
<trigger name="Matched">
<startWorkflow process="IDDMergeTask" taskKind="MERGE" taskTemplate="MergeTaskGenerator"
firstTaskType="AVOSMerge"/>
<event name="MatchedBE"/>
<role name="SYSTEM"/>
</trigger>
```

제목 영역에 대한 사용자 지정된 워크플로우 업데이트

제목 영역 어댑터에서 사용할 워크플로우를 사용자 지정할 경우 변경 내용 목록을 검토하고 사용자 지정된 워크플로우를 필요에 따라 업데이트해야 합니다.

다음 테이블에는 릴리스에 따라 기본 워크플로우에 적용된 변경 내용이 나열되어 있습니다.

릴리스	변경	지시 사항
10 HotFix 1	암호 암호화를 활성화합니다.	각 워크플로우에 대해 <code>sif:encrypted</code> 를 <code>true</code> 로 설정합니다.
10 HotFix 1	태스크 받은 편지함에 대한 프레젠테이션 매개 변수를 업데이트합니다.	“제목 영역에 대한 워크플로우에서 프레젠테이션 매개 변수 업데이트” 페이지 167
10.3	파일 첨부 기능을 활성화합니다.	“제목 영역에 대한 워크플로우에서 첨부 파일 활성화” 페이지 168

제목 영역에 대한 워크플로우에서 프레젠테이션 매개 변수 업데이트

제목 영역을 기반으로 워크플로우를 사용자 지정할 경우 .bpel 파일을 열고 프레젠테이션 매개 변수를 다음 매개 변수 목록과 비교합니다. 필요한 경우 각 태스크에 대해 프레젠테이션 매개 변수를 추가 또는 업데이트합니다.

식에 매개 변수에 대한 절대 참조가 포함될 수 있지만 사용자 지정된 워크플로우의 모범 사례는 태스크에 대한 `getInput()` 메서드를 가리키는 것입니다. 프레젠테이션 매개 변수에 대한 절대 참조를 `getInput()` 메서드에 대한 호출로 바꾸는 것을 고려하십시오. 예를 들어 `htd:getInput('processTaskRequest')/mdmavxsd:INFATask/`를 `htd:getInput('processTaskRequest')/mdmavxsd:INFATask/`로 바꿀 수 있습니다. 다음 테이블의 식은 입력 메서드를 호출합니다.

팁: 식에서는 네임스페이스 접두사 `mdmavxsd`를 사용합니다. 사용자 지정 워크플로우를 업데이트한 후 문제가 발생하면 `mdmavxsd`가 `urn:informatica.mdm.av.xsd`에 대한 접두사로 정의되었는지 확인합니다.

다음 테이블에는 프레젠테이션 매개 변수와 식이 나열되어 있습니다.

매개 변수	유형	식
subjectareauuid	문자열	\$InfaTask/mdmavxsd:taskData/mdmavxsd:subjectAreaUID
title	문자열	\$InfaTask/mdmavxsd:taskData/mdmavxsd:title
creator	문자열	\$InfaTask/mdmavxsd:taskData/mdmavxsd:creator
mdmtasktype	문자열	\$InfaTask/mdmavxsd:taskType/mdmavxsd:name
orsId	문자열	\$InfaTask/mdmavxsd:orsId
duedate	문자열	\$InfaTask/mdmavxsd:taskData/mdmavxsd:dueDate
tasktypename	문자열	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskType/mdmavxsd:name
taskTypeDisplayName	문자열	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskType/mdmavxsd:displayName
taskTypeDescription	문자열	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskType/mdmavxsd:description
taskTypePendingBVT	부울	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskType/mdmavxsd:pendingBVT
taskTypeDataUpdateType	문자열	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskType/mdmavxsd:dataUpdateType
taskTypeDisplayType	문자열	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskType/mdmavxsd:displayType
priorityOut	문자열	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:taskData/mdmavxsd:priority
workflowVersion	문자열	htd:getInput('processTaskRequest')/mdmavxsd:INFATask/ mdmavxsd:workflowVersion/text()

.bpel 파일 업데이트에 대한 자세한 내용은 ActiveVOS 설명서를 참조하십시오.

제목 영역에 대한 워크플로우에서 첨부 파일 활성화

버전 10.3에서는 사용자가 태스크에 파일을 첨부하는 것을 허용하도록 사용자 지정된 워크플로우를 업데이트할 수 있습니다. 사용자 지정된 워크플로우에서 이 기능을 활성화하려면 데이터를 초기화하는 프로세스를 업데이트하고 모든 사용자 프로세스를 업데이트합니다.

참고: 첨부 파일 기능을 일부 태스크로 제한하려는 경우 프로비저닝 도구에서 태스크 설정을 구성할 수 있습니다. 태스크 구성 업데이트에 대한 자세한 내용은 *Multidomain MDM 프로비저닝 도구 가이드* 항목을 참조하십시오.

1. ActiveVOS Designer에서 사용자 지정된 워크플로우의 .bpel 파일을 엽니다.
워크플로우가 캔버스에 나타납니다.

2. 데이터를 초기화하는 프로세스를 업데이트합니다.
 - a. **데이터 초기화** 프로세스를 선택합니다.
작업 대화 상자가 표시됩니다.
 - b. **새 스크립트**를 클릭합니다.
스크립트 작성기 대화 상자가 나타납니다.
 - c. 스크립트 상자에 다음 문을 추가합니다.
`abx:copyAllAttachments('StartRequest', '_peopleActivityAttachments')`
*StartRequest*를 워크플로우의 데이터를 초기화할 때 사용한 변수 이름으로 바꿉니다.
 - d. **확인**을 클릭합니다.
스크립트 작성기 대화 상자가 닫히고 스크립트가 **작업** 목록에 나타납니다.
 - e. **확인**을 클릭합니다.
작업 대화 상자가 닫힙니다.
3. 모든 사용자 프로세스를 업데이트합니다.
 - a. 사용자 프로세스를 선택합니다.
 - b. **속성** 탭을 클릭합니다.
 - c. **모두** 탭을 클릭합니다.
 - d. **첨부 파일 전파**를 확장합니다.
 - e. **시작 프로세스**를 모두로 설정합니다.
 - f. **종료 프로세스**를 **newOnly**로 설정합니다.
4. .bpel 파일을 저장합니다.

제목 영역을 기반으로 하는 ActiveVOS 워크플로우 재배포

버전 10.0 HotFix 2에서는 제목 영역을 기반으로 하는 워크플로우 어댑터의 ActiveVOS 워크플로우가 변경되었습니다. 제목 영역 기반 워크플로우 어댑터를 사용할 경우에는 리소스 키트에 제공되는 기본 워크플로우를 재배포해야 합니다. 태스크 워크플로우가 포함된 Informatica ActiveVOS 프로젝트를 MDM Hub 서버에 배포하려면 CommonMDM 프로젝트를 먼저 내보낸 후 MDMWorkflow 프로젝트를 내보냅니다.

1. Informatica ActiveVOS Designer에서 내보내려는 BeCommonMDM 프로젝트를 엽니다.
2. **파일 > 내보내기**를 클릭합니다.
내보내기 대화 상자가 열립니다.
3. **Orchestration** 아래에서 **기여 - 비즈니스 프로세스 보관**을 선택합니다. **다음**을 클릭합니다.
4. **서버 배포 옵션** 아래의 **배포 URL** 필드에 ActiveVOS 인스턴스의 URL을 입력합니다. **마침**을 클릭합니다.
5. **배포 완료** 대화 상자에서 **확인**을 클릭합니다.
6. BeMDMWorkflow 프로젝트에 대해 모든 단계를 반복합니다.
BeMDMWorkflow 프로젝트를 내보내기 전에 BeCommonMDM 프로젝트를 내보내야 합니다.

비즈니스 항목 및 비즈니스 항목 서비스 구성 파일 생성

비즈니스 항목 및 비즈니스 항목 서비스 구성 파일을 생성하려면 Informatica Data Director 구성 관리자를 사용하십시오.

1. 구성 관리자 **응용 프로그램** 창에서 비즈니스 항목 및 비즈니스 항목 서비스 구성으로 생성하고자 하는 구성을 가진 Informatica Data Director 응용 프로그램을 선택하십시오.

2. **비즈니스 항목 스키마 생성**을 클릭하십시오.

구성 관리자가 비즈니스 항목 및 비즈니스 항목 서비스 구성을 생성합니다.

3. 구성 관리자가 비즈니스 항목 및 비즈니스 항목 서비스 구성을 생성하는 동안 발생한 문제의 메시지를 표시합니다. 메시지는 구성 관리자가 생성 프로세스에서 문제를 해결하였는지 여부를 표시하고 구성 관리자가 문제를 해결하기 위해 변경한 사항을 설명합니다. 구성 관리자가 문제를 해결하지 못한 경우, 문제와 문제를 해결하기 위한 권고 조치를 기록해 두십시오.

부록 A

업그레이드 프로세스 문제 해결

업그레이드가 실패하거나 업그레이드 중에 문제가 발생하면 다음 정보를 사용하여 문제를 해결합니다.

JBoss 환경에서 EAR 파일이 허용된 시간 내에 배포되지 않습니다.

연산 참조 저장소 수를 늘리면 EAR 파일 배포 시간이 길어집니다. EAR 파일 배포 시간이 JBoss 환경의 허용된 배포 시간을 초과하면 업그레이드가 실패합니다.

이 문제를 해결하려면 EAR 파일 배포 시간을 수용하도록 허용된 배포 시간을 늘립니다. 허용되는 배포 시간의 기본값은 600초입니다.

1. 다음 디렉터리의 build.properties 파일에서 deploy.wait.time 속성 값을 늘립니다. <infadm 설치 디렉터리>/hub/server/bin
2. <JBoss 설치 디렉터리>/standalone/configuration 디렉터리로 이동합니다.
3. standalone-full.xml 파일에 다음 코드를 구성하여 제한 시간 값을 늘립니다.

```
<subsystem xmlns="urn:jboss:domain:deployment-scanner:1.1">  
  <deployment-scanner path="deployments" relative-to="jboss.server.base.dir" scan-interval="5000"  
    deployment-timeout="1200"/>  
</subsystem>
```

Hub 서버 업그레이드 오류가 발생했습니다.

이 문제를 해결하려면 EAR 파일을 재배포하여 Hub 서버 업그레이드를 다시 시도합니다.

참고: JBoss 환경에서 JBoss가 실행 중일 때 standalone-full.xml 파일의 데이터 소스 구성을 수동으로 변경하면 patchInstallSetup 스크립트 실행 시 구성 변경 사항이 유실됩니다.

1. 다음 디렉터리로 이동합니다. <MDM Hub 설치 디렉터리>/hub/server
2. 다음 명령을 실행하여 Hub 서버 응용 프로그램을 배포하고 응용 프로그램 서버 구성에 변경 사항을 적용합니다.

참고: 환경에 포함된 ActiveVOS가 없는 경우에는 명령에 ActiveVOS 사용자 이름과 암호를 포함하지 않아도 됩니다.

UNIX의 경우

WebLogic

```
patchInstallSetup.sh -Dweblogic.password=<WebLogic password> -Ddatabase.password=<MDM Hub Master  
database password> -Davos.username=<ActiveVOS Console username> -Davos.password=<ActiveVOS Console  
password> -Davos.jdbc.database.password=<ActiveVOS database password>
```

보안이 활성화된 WebSphere

```
patchInstallSetup.sh -Dwebsphere.password=<WebSphere password> -Ddatabase.password=<MDM Hub Master  
database password> -Davos.username=<ActiveVOS Console username> -Davos.password=<ActiveVOS Console  
password> -Davos.jdbc.database.password=<ActiveVOS database password>
```

보안이 비활성화된 WebSphere

```
patchInstallSetup.sh -Ddatabase.password=<MDM Hub Master database password> -  
Davos.username=<ActiveVOS Console username> -Davos.password=<ActiveVOS Console password> -  
Davos.jdbc.database.password=<ActiveVOS database password>
```

JBoss

```
patchInstallSetup.sh -Ddatabase.password=<MDM Hub Master database password> -  
Davos.username=<ActiveVOS Console username> -Davos.password=<ActiveVOS Console password> -  
Davos.jdbc.database.password=<ActiveVOS database password>
```

참고: UNIX의 경우 암호에 느낌표(!) 문자를 넣는 경우 느낌표(!) 문자 앞에 백슬래시를 넣어야 합니다. 예를 들어 암호가 `!!cmx!!`이면 `\!cmx\!`를 입력합니다.

Windows의 경우

WebLogic

```
patchInstallSetup.bat -Dweblogic.password=<WebLogic password> -Ddatabase.password=<MDM Hub Master  
database password> -Davos.username=<ActiveVOS Console username> -Davos.password=<ActiveVOS Console  
password> -Davos.jdbc.database.password=<ActiveVOS database password>
```

보안이 활성화된 WebSphere

```
patchInstallSetup.bat -Dwebsphere.password=<WebSphere password> -Ddatabase.password=<MDM Hub Master  
database password> -Davos.username=<ActiveVOS Console username> -Davos.password=<ActiveVOS Console  
password> -Davos.jdbc.database.password=<ActiveVOS database password>
```

보안이 비활성화된 WebSphere

```
patchInstallSetup.bat -Ddatabase.password=<MDM Hub Master database password> -  
Davos.username=<ActiveVOS Console username> -Davos.password=<ActiveVOS Console password> -  
Davos.jdbc.database.password=<ActiveVOS database password>
```

JBoss

```
patchInstallSetup.bat -Ddatabase.password=<MDM Hub Master database password> -  
Davos.username=<ActiveVOS Console username> -Davos.password=<ActiveVOS Console password> -  
Davos.jdbc.database.password=<ActiveVOS database password>
```

ActiveVOS 콘솔 자격 증명은 응용 프로그램 서버의 관리자 자격 증명과 같습니다.

ActiveVOS 데이터베이스 자격 증명은 `create_bpm` 스크립트를 실행하는 데 사용된 자격 증명과 같습니다.

MDM Hub 클린 업그레이드 오류가 발생했습니다.

버전 10.4로 업그레이드한 후 클린 업그레이드를 수행할 때 기존 데이터베이스를 사용하면 업그레이드가 실패합니다. 이 문제는 클린 업그레이드를 수행할 때마다 새로운 공개 키 및 개인 키 쌍이 생성되기 때문에 발생합니다. 새로운 키 쌍을 사용하여 기존 데이터베이스에 액세스할 수 없습니다.

이 문제를 해결하려면 다음 단계를 수행합니다.

1. Hub 서버의 새로운 공개 키 및 개인 키 쌍을 이전 공개 키 및 개인 키 쌍으로 바꿉니다.

Hub 서버의 공개 키 및 개인 키 쌍은 다음 디렉터리에 저장됩니다.

<MDM Hub 설치 디렉터리>/hub/server/resources/certificates

2. `postInstallSetup` 스크립트를 실행하여 Hub 서버 응용 프로그램을 배포하고 응용 프로그램 서버 구성에 변경 내용을 적용합니다.

`postInstallSetup` 스크립트 실행에 대한 자세한 내용은 *Multidomain MDM 설치 가이드*에서 Hub 서버 설치 후 태스크 장을 참조하십시오.

3. 처리 서버의 새로운 공개 키 및 개인 키 쌍을 이전 공개 키 및 개인 키 쌍으로 바꿉니다.

처리 서버의 공개 키 및 개인 키 쌍은 다음 디렉터리에 저장됩니다.

<MDM Hub 설치 디렉터리>/hub/cleanse/resources/certificates

4. `postInstallSetup` 스크립트를 실행하여 처리 서버 응용 프로그램을 배포하고 응용 프로그램 서버 구성에 변경 내용을 적용합니다.

`postInstallSetup` 스크립트 실행에 대한 자세한 내용은 *Multidomain MDM 설치 가이드*에서 처리 서버 설치 후 태스크 장을 참조하십시오.

WebLogic 환경에서 처리 서버 업그레이드 오류가 발생했습니다.

WebLogic 환경에서 처리 서버를 업그레이드할 때 다음 업그레이드 오류가 발생할 수도 있습니다.

```
Unable to start application, deployment error msg: weblogic.management.ManagementException: [Deployer:
149196]Rejecting start request for application siperian-mrm-cleanse.ear because stop request is running
for the application.
```

이 문제를 해결하려면 WebLogic 관리 콘솔을 사용하여 `siperian-mrm-cleanse.ear` 파일을 수동으로 배포한 다음 응용 프로그램 서버를 다시 시작합니다.

처리 서버 업그레이드 오류가 발생했습니다.

이 문제를 해결하려면 `EAR` 파일을 재배포하여 처리 서버 업그레이드를 다시 시도합니다.

참고: JBoss가 실행 중일 때 `standalone-full.xml` 파일의 데이터 소스 구성을 수동으로 변경하면 `patchInstallSetup` 스크립트 실행 시 구성 변경 사항이 유실됩니다.

1. 다음 디렉터리로 이동합니다. <MDM Hub 설치 디렉터리>/hub/cleanse
2. 다음 명령을 실행하여 처리 서버 응용 프로그램을 배포하고 응용 프로그램 서버 구성에 변경 사항을 적용합니다.

UNIX의 경우

WebLogic

```
patchInstallSetup.sh -Dweblogic.password=<WebLogic 암호> -Ddatabase.password=<데이터베이스 암호>
```

WebSphere

```
patchInstallSetup.sh -Ddatabase.password=<데이터베이스 암호>
```

JBoss

```
patchInstallSetup.sh -Ddatabase.password=<데이터베이스 암호>
```

Windows의 경우

WebLogic

```
patchInstallSetup.bat -Dweblogic.password=<WebLogic 암호> -Ddatabase.password=<데이터베이스 암호>
```

WebSphere

```
patchInstallSetup.bat -Ddatabase.password=<데이터베이스 암호>
```

JBoss

```
patchInstallSetup.bat -Ddatabase.password=<데이터베이스 암호>
```

참고: UNIX의 경우 암호에 느낌표(!) 문자를 넣는 경우 느낌표(!) 문자 앞에 백슬래시를 넣어야 합니다. 예를 들어 암호가 `!!cmx!!`이면 `\!cmx\!`를 입력합니다.

연산 참조 저장소 업그레이드에서 ORA-00955 오류가 발생했습니다.

연산 참조 저장소를 업그레이드하면 업그레이드는 성공하지만 `sip_ant` 로그에 다음 오류가 나타납니다.

```
[exec] CREATE SEQUENCE "C_REPOS_ZDT_EVENT_SEQ" MINVALUE 1 MAXVALUE 9999999999999999
      INCREMENT BY 1 START WITH 1 CACHE 20 NOORDER CYCLE
[exec] *
[exec] ERROR at line 1:
[exec] ORA-00955: name is already used by an existing object
[exec]
```

이 오류는 무시해도 안전합니다.

변경 목록을 빈 연산 참조 저장소로 승격했을 때 ORA-00910 오류가 발생했습니다.

변경 목록을 빈 연산 참조 저장소로 승격할 때 일치 열의 총 길이가 4000보다 크면 다음 오류가 발생합니다.

ORA-00910: specified length too long for its datatype

변경 목록을 빈 연산 참조 저장소로 승격하려면 MDM Hub가 외부 일치 입력 테이블에 추가하는 최대 열 길이가 4000을 초과하지 않아야 합니다. 일치 열 길이는 일치 열의 소스인 모든 기본 개체 열의 길이와 소스 열 수의 합입니다.

Oracle 환경의 연산 참조 저장소 업그레이드에서 ORA-20005 오류가 발생했습니다.

sip_ant updateorsdatabase 실행 시 ORA-20005 오류가 발생한 경우 다음 단계를 수행하십시오.

1. 다음 명령을 실행하여 필수 사용 권한을 부여합니다.

```
exec dbms_java.grant_permission(upper('ORS_USER'),'SYS:java.net.SocketPermission','*','connect,resolve');
```

2. 다음 명령을 실행하여 Oracle에서 Java 클래스가 로드되는지 확인합니다.

```
select dbms_java.longname(object_name), status from user_objects where object_type='JAVA CLASS';
```

3. 클래스가 로드되지 않는 경우 다음 명령을 실행하여 클래스를 다시 로드합니다.

```
loadjava -verbose -force -resolve -oracleresolver -user &ors_name/&ors_passwd@&tns_name siperian-cleansecaller.jar
loadjava -verbose -force -resolve -oracleresolver -user &ors_name/&ors_passwd@&tns_name siperian-dbutil.jar
```

Hub 저장소 업그레이드 오류가 발생했습니다.

부분적으로 업그레이드된 스키마에서 Hub 저장소 업그레이드를 다시 실행할 수 없습니다. 업그레이드 오류가 발생하면 전체 백업에서 데이터베이스를 복원한 다음 Hub 저장소 업그레이드를 다시 실행합니다.

예약어가 포함된 열 이름으로 인해 Hub 저장소 업그레이드가 실패한 경우 Informatica 글로벌 고객 지원 센터에 문의하여 이름이 바뀐 열로 데이터를 마이그레이션하는 스크립트를 요청하십시오.

영어가 아닌 로캘에서 업그레이드하면 어떤 테이블은 영어로 되어 있고 어떤 테이블은 로캘 언어로 되어 있습니다.

Hub 저장소 데이터베이스 환경이 영어가 아닌 로캘로 설정된 경우 MDM Hub 마스터 데이터베이스 및 연산 참조 저장소를 업그레이드하도록 업그레이드 스크립트를 실행하기 전에, 문자 집합을 변경하여 유니코드로 설정해야 합니다. 업그레이드 중에 모든 테이블 메타데이터가 변환 키가 포함된 영어로 변환됩니다. 유니코드 문자 집합을 선택하지 않은 경우에는 일부 테이블만 변환됩니다.

Hub 콘솔이 시작되지 않음

Hub 콘솔에서 지원되는 JRE(Java Runtime Environment)를 사용하고 있는지 확인합니다. 시스템 요구 사항은 Informatica Network에서 이 버전의 Multidomain MDM에 대한 PAM(Product Availability Matrix)을 참조하십시오.

<https://network.informatica.com/community/informatica-network/product-availability-matrices/overview>

JBoss 환경에서 Hub 콘솔 실행 시 오류가 발생합니다.

JBoss 환경에서 JBoss 응용 프로그램 서버를 다시 시작하지 않으면 Hub 콘솔을 실행할 수 없습니다. MDM Hub에서 리포지토리 계층이 초기화되지 않았음을 표시하는 오류가 생성됩니다.

이 문제를 해결하려면 배치 파일에서 다음 코드를 실행하여 JBoss를 다시 시작합니다.

```
rmmdir C:\<JBoss installation directory>\standalone\tmp /s /q
mkdir C:\<JBoss installation directory>\standalone\tmp
C:\<JBoss installation directory>\bin\standalone.bat -c standalone-full.xml -b 0.0.0.0
```

Db2 환경에서 Hub 콘솔 실행 시 오류가 발생합니다.

Db2 데이터 소스가 있는 MDM Hub 환경에서 다음과 같이 Hub 콘솔 실행 시 오류가 발생합니다.

SIP-09070: SIP-10318: 데이터 액세스 오류로 인해 사용자를 가져올 수 없습니다.

SIP-10324: 데이터 개체를 로드하는 중 예기치 않은 예외가 발생했습니다. `java.lang.NullPointerException`

이 문제는 MDM Hub 및 응용 프로그램 서버에서 관리 사용자 이름에 사용된 대/소문자의 불일치로 인해 발생합니다. 예를 들어, MDM Hub에는 관리 사용자 DB2ADMIN(대문자)이 있지만 응용 프로그램 서버에는 db2admin(소문자)이 있습니다.

이 문제를 해결하려면 응용 프로그램 서버의 사용자 이름이 MDM Hub의 사용자 이름과 정확하게 일치하는지 확인합니다.

참고: 대/소문자 구분과 관련된 문제를 방지하기 위해 Informatica는 Db2용 사용자 이름을 정의할 때 전체를 대문자로 사용할 것을 권장합니다.

예를 들어 WebSphere를 사용 중인 경우 WebSphere 콘솔에서 사용자 이름을 설정합니다.

1. WebSphere 콘솔을 엽니다.
2. 리소스 > 데이터 소스 > siperian-cmx_system-ds > 사용자 지정 속성으로 이동합니다.
3. 사용자 필드에서 다음과 같이 대문자로 입력합니다. DB2ADMIN
4. 암호 필드에서 이 사용자에 대한 암호를 입력합니다.
5. 적용을 클릭한 다음 저장을 클릭합니다.
6. WebSphere를 다시 시작합니다.
7. Hub 콘솔을 실행하고 로그인합니다.

프로비저닝 도구에서 작성된 변경 내용을 Db2 환경에서 적용할 수 없습니다.

Db2 환경에서 연산 참조 저장소가 큰 경우 프로비저닝 도구에서 변경 내용을 적용하면 다음 오류 메시지가 나타납니다.

```
Failed to set user workspace configuration.
```

이 문제를 해결하려면 연산 참조 저장소에서 다음 Db2 명령을 실행하여 사용자 작업 공간의 열 길이를 늘립니다.

```
UPDATE C_REPOS_COLUMN SET DATA_LENGTH = 50000000 WHERE TABLE_NAME = 'C_REPOS_USER_WORKSPACE' AND  
COLUMN_NAME = 'WORKSPACE_DATA'  
ALTER TABLE C_REPOS_USER_WORKSPACE ALTER COLUMN WORKSPACE_DATA SET DATA TYPE BLOB(50M)  
REORG TABLE C_REPOS_USER_WORKSPACE  
COMMIT
```

IDD가 레거시 데이터 보기를 사용하여 제목 영역에 기반하는 데이터를 볼 수 없습니다.

IDD에서 레코드를 보는 기본 페이지는 비즈니스 항목에 기반한 항목 보기입니다.

레거시 데이터 보기를 사용하려면 cmxserver.properties 파일에서 dataview.enabled를 true로 설정합니다.

자세한 내용은 *비즈니스 항목 데이터 모델로 IDD 응용 프로그램 마이그레이션 방법* 문서를 참조하십시오.

IDD에서 SIP-BV-11500 오류가 발생합니다.

IDD에서 다음 오류가 발생할 수 있습니다. SIP-BV-11500 치명적 오류 연산 참조 저장소 localhost-orcl-MDM_SAMPLE에 워크플로우 엔진이 구성되지 않았습니다. 워크플로우가 사용되지 않을 경우에도 IDD에서 사용할 수 있도록 각 연산 참조 저장소에 워크플로우 엔진이 구성되어 있어야 합니다.

이 문제를 해결하려면 기본 워크플로우 어댑터가 구성되었는지 확인하십시오.

자세한 내용은 다음 KB 문서를 참조하십시오.

<https://kb.informatica.com/solution/23/Pages/55/381456.aspx?myk=381456>

메타데이터 유효성을 검증할 때 메타데이터에 개체가 있지만 데이터베이스에는 없다는 내용의 오류가 표시됩니다.

리포지토리 관리자를 사용하여 문제를 해결하려고 하면 다음 오류가 발생합니다. ORA-00955 기존 개체에서 사용하는 이름입니다.

이 문제를 해결하려면 오류가 발생한 테이블에 프록시 역할에 대한 올바른 권한이 부여되었는지 확인하십시오. 오류가 발생하지 않은 테이블을 참조하여 필요한 사용 권한 목록을 확인하십시오.

Windows에서 일치 토큰이 생성될 때 오류가 발생합니다.

일치 토큰 생성 프로세스에서 `ssa.ssaname3.jssan3cl` 클래스를 초기화할 수 없다는 내용의 오류가 반환됩니다.

1. SSA-NAME3에 대한 DLL(동적 연결 라이브러리) 파일이 포함된 <MDM 설치 디렉터리>/hub/cleanse/lib 디렉터리에 대한 경로가 PATH 환경 변수에 포함되어 있는지 확인합니다.
2. Microsoft Visual Studio 2015용 Visual C++ 재배포 가능 패키지가 MDM Hub의 검색 및 일치를 수행하는 처리 서버에 설치되어 있는지 확인합니다.
3. Microsoft Visual Studio 2015용 Visual C++ 재배포 가능 패키지가 설치되어 있는 경우 Dependency Walker(depends.exe) 같은 종속성 검사기를 사용하여 `jssan3cl.dll`을 로드하고 Visual C++ 재배포 가능 패키지가 성공적으로 적용되었는지 확인합니다.

팁: Visual Studio 2015용 Visual C++ 재배포 가능 패키지를 사용하려면 Windows Server에 운영 체제 패치가 설치되어 있어야 합니다. Visual C++ 재배포 가능 패키지를 설치하기 전에 운영 체제 요구 사항을 확인하십시오. 예를 들어 기존 버전인 Windows Server 2012에서는 Visual C++ 재배포 가능 패키지를 설치하기 전에 약 100개(전체적으로 약 2GB)의 패치를 운영 체제에 적용해야 합니다.

WebLogic 응용 프로그램 서버의 Microsoft SQL Server 환경에서 업그레이드한 후 Hub 콘솔에 로그인할 수 없습니다.

Hub 콘솔에 로그인하면 Null 포인터 예외가 발생합니다.

이 문제를 해결하려면 <Microsoft SQL Server 설치 디렉터리>/sqljdbc_4.0\enu\xa에 있는 `xa_install.sql` 스크립트에 삭제 명령을 주석 처리하고, 스키마 명령 및 역할 명령을 생성합니다. 스크립트를 실행한 후 응용 프로그램 서버를 다시 시작합니다.

WebSphere 응용 프로그램 서버에 Hub 서버를 설치하는 경우 업그레이드 구성 요소 `patchInstallSetup`이 실패합니다.

이 문제를 해결하려면 <WebSphere 프로파일 홈>/properties/soap.client.props 파일을 열고 `com.ibm.SOAP.requestTimeout`을 늘린 다음 WebSphere 서버 프로ファイルを 다시 시작합니다. `patchInstallSetup.bat`를 다시 실행합니다.

IBM AIX 환경에서 Hub 서버를 업그레이드하면 `entity360view.ear` 파일이 배포에 실패합니다.

이 문제를 해결하려면 `patchInstallSetup.sh` 스크립트를 실행합니다.

부록 B

FAQ

업그레이드하려면 새로운 라이선스 파일이 필요합니까?

예. Multidomain MDM 버전 10.1.x 이하에서 업그레이드하는 경우 새 라이선스 파일이 필요합니다.

10.0 이전 버전의 데이터베이스 사용자 종료를 사용할 수 있습니까?

아니요. 10.0 이상 버전에서는 데이터베이스 계층에서 실행되는 데이터베이스 사용자 종료가 사용되지 않습니다.

업그레이드 프로세스 중에 DBA 사용자 이름과 암호를 제공해야 하는 이유는 무엇입니까?

업그레이드 프로세스에서는 권한 부여 및 시퀀스 생성 등 DBA 수준 사용 권한이 필요한 작업이 수행됩니다. 업그레이드 프로세스에서 이러한 작업을 수행하려면 DBA 자격 증명이 필요합니다.

업그레이드 프로세스 중에 기존 버전의 ActiveVOS는 어떻게 됩니까?

ActiveVOS가 설치되어 있는 환경에서 인플레이스 업그레이드를 수행하는 경우 해당 ActiveVOS가 최신 버전이 아니면 업그레이드 프로세스 중에 최신 ActiveVOS가 설치됩니다. 필요한 ActiveVOS 버전을 보려면

Informatica 네트워크:

<https://network.informatica.com/community/informatica-network/product-availability-matrices/overview>에서 Product Availability Matrix를 참조하십시오.

ActiveVOS가 설치되지 않은 환경에서 MDM 업그레이드 프로세스를 수행하는 경우 필수적으로 ActiveVOS를 설치해야 합니까?

아니요. 업그레이드 프로세스 중에 ActiveVOS 설치를 선택하는 메시지가 표시됩니다.

이 버전에서 Informatica Data Director에 대한 권장 화면 해상도가 변경되었습니까?

아니요. 권장 화면 해상도는 변경되지 않았습니다. Informatica Data Director에 권장되는 화면 해상도는 1280 x 1024입니다.

Multidomain MDM 버전 10.2 이상으로 업그레이드할 때 Informatica Data Quality를 업그레이드해야 합니까?

예. 환경에서 IDQ(Informatica Data Quality)를 사용하는 경우 IDQ 버전 10.1로 업그레이드해야 합니다. 시스템 요구 사항은 Informatica 네트워크:

<https://network.informatica.com/community/informatica-network/product-availability-matrices/overview>에서 Product Availability Matrix를 참조하십시오.

MDM Hub 보안을 사용자 지정하려면 어떻게 해야 합니까?

Hub 보안 사용자 지정에 대한 자세한 내용은 *Multidomain MDM 보안 가이드* 및 *MDM Multidomain MDM의 보안 구성 유틸리티 사용 방법* 문서를 참조하십시오.

Java 8이 지원됩니까?

예. 이 버전의 Multidomain MDM은 Java 8을 지원합니다. 시스템 요구 사항은 Informatica 네트워크: <https://network.informatica.com/community/informatica-network/product-availability-matrices/overview> 에서 Product Availability Matrix를 참조하십시오.

참고: ActiveVOS에는 Java 7이 필요합니다.

부록 C

기존 ActiveVOS 태스크 처리

이 부록에 포함된 항목:

- [기존 ActiveVOS 태스크 처리 개요, 179](#)
- [마이그레이션 속성, 179](#)
- [마이그레이션 스크립트 실행 속성 파일 사용, 180](#)
- [마이그레이션 스크립트 실행 명령줄 속성 사용, 181](#)

기존 ActiveVOS 태스크 처리 개요

Multidomain MDM 버전 10.1 이하에서 생성된 ActiveVOS 태스크로 작업하려면 마이그레이션 스크립트를 주기적으로 실행하여 해당 태스크를 필수 프레젠테이션 매개 변수로 채워야 합니다. 마이그레이션 스크립트를 실행하지 않으면 태스크 관리자에 태스크가 표시되지 않습니다. 버전 10.1로 업그레이드하기 전에 작성된 모든 태스크가 처리될 때까지 마이그레이션 스크립트를 실행합니다.

마이그레이션 스크립트에서 일부 속성을 설정해야 합니다. 빌드 파일에 속성을 추가하거나 명령줄에서 속성을 추가할 수 있습니다.

마이그레이션 속성

다음 테이블에는 마이그레이션 속성이 설명되어 있습니다.

속성	설명
avos.jdbc.database.driver.jar	ActiveVOS 데이터베이스의 JDBC 드라이버가 포함된 JAR 파일의 경로입니다. 이 매개 변수는 Hub 서버를 설치하는 동안 avos 접두사 없이 <infamdm 설치 디렉터리>\conf\avos.install.properties에 채워집니다.
avos.jdbc.database.driver.class	ActiveVOS 데이터베이스의 JDBC 드라이버 클래스입니다. 이 매개 변수는 Hub 서버를 설치하는 동안 avos 접두사 없이 <infamdm 설치 디렉터리>\conf\avos.install.properties에 채워집니다.
avos.jdbc.database.url	ActiveVOS 데이터베이스 연결 URL입니다. 이 매개 변수는 Hub 서버를 설치하는 동안 avos 접두사 없이 <infamdm 설치 디렉터리>\conf\avos.install.properties에 채워집니다.

속성	설명
avos.jdbc.database.username	ActiveVOS 데이터베이스 사용자 이름입니다. 이 매개 변수는 Hub 서버를 설치하는 동안 avos 접두사 없이 <infamdm 설치 디렉터리>\conf\avos.install.properties에 채워집니다.
avos.jdbc.database.password	ActiveVOS 데이터베이스 암호입니다.
avos.ws.protocol	ActiveVOS 서버 연결을 위한 프로토콜입니다. http 또는 https일 수 있습니다.
avos.ws.host	ActiveVOS가 실행되는 응용 프로그램 서버의 호스트 이름입니다.
avos.ws.port	응용 프로그램 서버 연결의 포트 번호입니다.
avos.ws.trusted.username	트러스트된 사용자의 사용자 이름입니다. 참고: 트러스트된 사용자는 Multidomain MDM 설치 및 업그레이드 프로세스 중에 생성됩니다.
avos.ws.trusted.password	트러스트된 사용자의 암호입니다. 참고: 트러스트된 사용자는 Multidomain MDM 설치 및 업그레이드 프로세스 중에 생성됩니다.
avos.hub.username	모든 태스크 관리 역할에 속해 있는 MDM Hub 사용자입니다.
avos.ws.pagesize	단일 데이터베이스 트랜잭션에서 처리되고 ActiveVOS에서 일괄 로드되는 태스크의 수입니다.
avos.ws.statuses	선택 사항입니다. 처리해야 하는 ActiveVOS 태스크 상태의 심표로 구분된 목록입니다. 예를 들면 READY 또는 IN_PROGRESS 상태의 태스크입니다. 기본적으로 모든 태스크가 처리됩니다.

마이그레이션 스크립트 실행 속성 파일 사용

워크플로우에 연결된 역할에 속하는 MDM 사용자를 생성합니다. Hub 서버 build.properties 파일에 마이그레이션 속성을 추가한 후 마이그레이션 스크립트 명령을 사용하여 MDM Hub 마스터 데이터베이스를 실행합니다. 마이그레이션 후 태스크 소유자는 할당된 태스크에서 작업을 수행할 수 있습니다.

정기적인 일정에 따라 마이그레이션 스크립트를 실행합니다. 제목 영역 워크플로우 어댑터에 대한 모든 태스크가 처리된 후에는 스크립트를 실행할 필요가 없으며 이 목적으로 생성한 MDM 사용자를 삭제해도 됩니다.

1. MDM Hub 사용자를 생성하고 워크플로우에 참여하는 모든 역할을 사용자에게 할당합니다.
예를 들어 미리 정의된 워크플로우에는 DataSteward, Manager 및 SrManager 역할이 사용됩니다.
2. 텍스트 편집기에서 다음 파일을 엽니다.
<MDM Hub 설치 디렉터리>/hub/server/bin/build.properties
3. 마이그레이션 다음 속성을 build.properties 파일에 추가합니다. 속성 목록은 [“마이그레이션 속성” 페이지 179](#) 항목을 참조하십시오.
4. 명령 프롬프트를 엽니다.
5. 다음 디렉터리로 이동합니다.
<MDM Hub 설치 디렉터리>/hub/server/bin

6. 마이그레이션 스크립트 명령을 사용하여 MDM Hub 마스터 데이터베이스 업그레이드 스크립트를 실행합니다.
 - UNIX의 경우. `sip_ant.sh migrate-avos-sa-tasks`
 - Windows의 경우. `sip_ant.bat migrate-avos-sa-tasks`

마이그레이션 스크립트 실행 명령줄 속성 사용

워크플로우에 연결된 역할에 속하는 MDM 사용자를 생성합니다. 명령 프롬프트를 열고 원하는 속성을 사용하여 스크립트를 실행합니다. 마이그레이션 후 태스크 소유자는 할당된 태스크에서 작업을 수행할 수 있습니다.

정기적인 일정에 따라 마이그레이션 스크립트를 실행합니다. 제목 영역 워크플로우 어댑터에 대한 모든 태스크가 처리된 후에는 스크립트를 실행할 필요가 없으며 이 목적으로 생성한 MDM 사용자를 삭제해도 됩니다.

1. MDM Hub 사용자를 생성하고 워크플로우에 참여하는 모든 역할을 사용자에게 할당합니다.
예를 들어 미리 정의된 워크플로우에는 DataSteward, Manager 및 SrManager 역할이 사용됩니다.
2. 명령 프롬프트를 엽니다.
3. 다음 디렉터리로 이동합니다.
<MDM Hub 설치 디렉터리>/hub/server/bin
4. 마이그레이션 스크립트 명령 및 마이그레이션 속성을 사용하여 MDM Hub 마스터 데이터베이스 업그레이드 스크립트를 실행합니다. 속성 목록은 [“마이그레이션 속성” 페이지 179](#) 항목을 참조하십시오.
예를 들어 다음 명령 중 하나를 실행할 수 있습니다.

- UNIX의 경우.

```
sip_ant.sh migrate-avos-sa-tasks -Davos.jdbc.database.password=!!cmx!!  
-Davos.ws.protocol=http -Davos.ws.host=localhost -Davos.ws.port=8080 -Davos.ws.pagesize=100  
-Davos.ws.trusted.username=avos -Davos.ws.trusted.password=avos -Davos.hub.username=admin
```

- Windows의 경우.

```
sip_ant.bat migrate-avos-sa-tasks -Davos.jdbc.database.password=!!cmx!!  
-Davos.ws.protocol=http -Davos.ws.host=localhost -Davos.ws.port=8080 -Davos.ws.pagesize=100  
-Davos.ws.trusted.username=avos -Davos.ws.trusted.password=avos -Davos.hub.username=admin
```

부록 D

메타데이터 캐시 구성

이 부록에 포함된 항목:

- [메타데이터 캐시 구성\(선택 사항\), 182](#)

메타데이터 캐시 구성(선택 사항)

메타데이터 캐시는 데이터 개체, 리포지토리, 개체 및 검색 토큰과 같은 항목을 관리합니다. MDM Hub는 메타데이터 캐시에 Infinispan을 사용합니다. Infinispan은 Hub 서버를 통해 설치됩니다. Hub 서버에서 사용하는 캐시의 경우 Infinispan 구성 파일에 기본 특성 값이 포함됩니다.

버전 10.1 이하에서 MDM Hub는 메타데이터 캐시에 JBoss 캐시를 사용했습니다. 이러한 버전에서 업그레이드 한 후에는 MDM Hub 서버가 JBoss 캐시 구성 파일 대신 Infinispan 구성 파일을 사용합니다.

이전 버전의 MDM Hub에서 JBoss 캐시 구성 파일을 편집한 경우 Infinispan 구성 파일을 편집해야 할 수 있습니다. 편집이 필요한지 여부는 파일을 편집한 이유에 따라 달라집니다.

네트워크 정책

조직의 네트워크 정책을 해결하기 위해 JBoss 캐시 파일을 편집한 경우 동일한 정책 변경으로 Infinispan 파일 및 jgroups* 파일을 업데이트합니다.

성능

캐시 성능을 개선하기 위해 JBoss 캐시 파일을 편집한 경우 먼저, 기본 Infinispan 값으로 MDM Hub를 실행합니다. 성능 문제가 발생하면 JBoss 캐시 구성 파일에서 변경된 값을 Infinispan 구성 파일에 복사합니다. 여전히 성능 문제가 발생하면 Infinispan을 익힌 후 환경에 적합한 값으로 조정합니다.

Infinispan 특성

다음 테이블에는 기본 Infinispan 특성 값이 요약되고 특성과 이전 JBoss 특성 간의 매핑이 표시되어 있습니다.

Infinispan 요소 및 특성	기본값	설명	JBoss 특성
locking acquire-timeout	60000	Hub 서버가 잠금 획득을 시도할 수 있는 최대 시간입니다.	lockAcquisitionTimeout
transaction stop-timeout	30000	이 특성은 캐시가 중지된 경우 Hub 서버의 원격 및 로컬 트랜잭션이 완료될 때까지 Infinispan이 대기하는 최대 시간을 설정합니다.	sync replTimeout

Infinispan 요소 및 특성	기본값	설명	JBoss 특성
transport cluster	infinispan-cluster	기본 그룹 통신 클러스터의 이름입니다.	clustering
transport stack	UDP	구성 유형(UDP 또는 TCP)입니다. 구성은 jgroups-udp.xml 파일과 jgroups-tcp.xml 파일에 정의됩니다.	jgroupsConfig
transport node-name	\$node\$	현재 노드의 이름입니다. 이 특성은 Hub 서버가 설정합니다. 노드 이름은 호스트 이름과 난수의 조합으로 기본 설정됩니다. 동일한 호스트에 있는 여러 노드는 숫자로 구분됩니다.	--
transport machine	\$machine\$	노드가 실행 되는 시스템의 ID입니다. 이 특성은 Hub 서버가 설정합니다.	--
expiration lifespan	--	캐시 항목의 최대 수명(밀리초)입니다. 수명을 초과한 캐시 항목은 클러스터 내에서 만료됩니다. 성능을 최적화해야 하는 경우 DISABLE_WHEN_LOCK, DATA_OBJECTS 및 REPOS_OBJECTS 절에 대한 수명을 늘리십시오. 예를 들어 한 시간(3600000)에서 하루(86400000)로 수명을 늘릴 수 있습니다. 각 캐시에는 이 특성에 대한 고유한 기본값이 있습니다. 기본값을 찾으려면 infinispanConfig.xml 파일을 엽니다.	eviction timeToLive
expiration interval	--	수명을 확인하는 최대 간격입니다. 성능을 최적화해야 하는 경우 DISABLE_WHEN_LOCK, DATA_OBJECTS 및 REPOS_OBJECTS 절에 대한 간격을 늘리십시오. 예를 들어 5초(5000)에서 5분(300000)으로 간격을 늘릴 수 있습니다. 각 캐시에는 이 특성에 대한 고유한 기본값이 있습니다. 기본값을 찾으려면 infinispanConfig.xml 파일을 엽니다.	eviction timeToLive

Infinispan 특성 편집

메타데이터 캐시 특성을 구성하려면 Hub 서버에 대한 infinispanConfig.xml 파일을 편집합니다. Infinispan 구성 관련 도움말은 [Infinispan 설명서](#)를 참조하십시오.

참고: 처리 서버에도 Infinispan 구성 파일이 있습니다. 기본 특성 값을 사용하는 것으로도 충분하지만 처리 서버의 성능 문제가 발견될 경우 특성 값을 세부 조정할 수 있습니다.

1. 다음 디렉터리로 이동합니다. <MDM Hub 설치 디렉터리>/hub/server/resources
2. 다음 파일의 백업 사본을 생성합니다. infinispanConfig.xml
3. infinispanConfig.xml 파일을 열고 Infinispan 버전 번호를 확인합니다. 버전 번호는 xsi:schemaLocation 특성에 표시됩니다.
4. Infinispan 버전은 설명서를 검토하십시오.

참고: 다음 URL에서 경로에 #.#이 포함되는 모든 위치의 버전 번호를 대체합니다.

- 구성 스키마를 보려면 파일의 xsi:schemaLocation 특성에 포함된 URL로 이동합니다.

- 특성에 대한 자세한 내용은 <https://docs.jboss.org/infinispan/<#.x>/configdocs/>를 참조하십시오.
 - Infinispan에 대한 자세한 내용을 보려면 <http://infinispan.org/docs/<#.x>>에서 "FAQ" 링크를 선택하십시오.
5. 파일을 편집하고 저장합니다.

인덱스

A

ActiveVOS

- URN, 설정 [157](#), [164](#)
 - 연결 업데이트 [126](#)
 - 연결 테스트 [126](#)
 - 자동 설치 프로그램 속성 [88](#)
- ActiveVOS 콘솔 관리자
- abAdmin 역할 [50](#), [56](#), [66](#)
 - 생성 [50](#), [56](#), [66](#)

B

환경 보고서

- 검토 [128](#)
- 저장 [128](#)

C

cmxserver.log 파일 [90](#), [101](#)

D

Data Director

- 업그레이드 테스트 [132](#)
- 제목 영역에 대한 업그레이드 테스트 [132](#)

E

Elasticsearch

- 고가용성 [136](#)
- elasticsearch 보관 파일
- 추출 [137](#)
- Elasticsearch 설치
- 선행 조건 [136](#)
 - 설치 전 태스크 [136](#)

H

Hub 서버

- 그래픽 모드에서 업그레이드 [82](#)
 - 업그레이드 다시 적용 [91](#)
 - 자동 속성 파일 [88](#)
 - 자동 업그레이드 [87](#), [89](#)
- Hub 서버 업그레이드
- 로그 파일 [90](#)
 - 정보 [81](#)
 - 콘솔 모드 [85](#)
- Hub 저장소
- 테이블스페이스, 생성 [27](#), [37](#)

Hub 저장소 업그레이드

- 마스터 데이터베이스 [74](#)
- 연산 참조 저장소
- 업그레이드 [76](#)
- 정보 [73](#)
- hub_cleansse_install.bin [92](#)
- hub_cleansse_install.exe [92](#)
- hub_install.bin [82](#)
- hub_install.exe [82](#)

I

IBM Db2 구성

- MDM Hub용 [37](#)
- infamdm_installer_debug.txt 파일 [90](#), [101](#)
- Infinispan
- 구성 [183](#)
- Informatica ActiveVOS
- 스키마 생성 [30](#), [34](#), [42](#)

J

Java 옵션

- 구성 [61](#)
- JBoss
- JBoss 클러스터 노드에서 시작 [52](#)
 - 독립 실행형 JBoss 인스턴스에서 시작 [51](#)
- JBoss 구성
- MDM Hub용 [45](#)
- JBoss클러스터 노드에서 시작
- 독립 실행형 인스턴스에서 시작 [51](#)
- JVM 매개 변수
- 구성 [61](#)

M

MDM Hub

- 업그레이드 테스트 [131](#)
- MDM Hub 마스터 데이터베이스 업그레이드
- 자동 모드 [75](#)
 - 자세한 정보 표시 모드 [74](#)
- Microsoft SQL Server
- ODBC 드라이버 [34](#)
 - unixODBC 드라이버 [34](#)
 - 구성 [33](#)
 - 데이터 파일 저장소 [34](#)
 - 배포 트랜잭션 [33](#)
 - 설치 [33](#)
- Microsoft SQL Server 구성
- MDM Hub용 [33](#)

N

- 데이터베이스
 - 수동으로 생성 [38](#)
 - 연결 테스트 [27](#)
- 로그 파일
 - cmxserver.log 파일 [90](#), [101](#)
 - Hub 서버 업그레이드 [90](#)
 - Infamdm_Cleanse_Match_Server_InstallLog.xml 파일 [101](#)
 - infamdm_installer_debug.txt 파일 [90](#), [101](#)
 - Infamdm_Server_InstallLog.xml 파일 [90](#)
 - postInstallSetup.log 파일 [90](#), [101](#)
 - 응용 프로그램 서버 로그 파일 [90](#), [101](#)
 - 정리 일치 서버 업그레이드 [101](#)
- 리소스 키트
 - 업그레이드 [102](#)
 - 제거 [102](#)
- 메타데이터
 - 유효성 검사 [123](#)
 - 유효성 검사 메시지, 확인 [124](#)
- 메타데이터 유효성 검사
 - 유효성 검사 확인 [123](#)
- 문제 해결
 - 설치 후 프로세스 [171](#)
- 사용자 지정 코드, 테스트 [131](#)
- 서버 속성 구성 [49](#)
- 스키마 암호
 - 암호화 [126](#)
 - 업데이트 [126](#)
- 업그레이드
 - 개요 [11](#), [12](#)
 - 마스터 데이터베이스 [74](#)
 - 지침 [12](#)
- 업그레이드 테스트
 - Hub 콘솔 도구 [131](#)
 - 비즈니스 항목이 있는 Data Director [132](#)
 - 정보 [131](#)
 - 제목 영역이 있는 Data Director [132](#)
 - 프로비저닝 도구 [131](#)
- 업그레이드 프로세스
 - 인플레이스 업그레이드 [13](#)
 - 클린 업그레이드 [12](#)
- 연산 참조 저장소
 - 등록 [21](#)
- 연산 참조 저장소 업그레이드
 - 자동 모드 [79](#)
 - 자세한 정보 표시 모드 [76](#)
- 워크플로우
 - 사용자 지정, 첨부 파일 활성화 [160](#)
- 유효성 검사 결과
 - 저장 [124](#)
- 일치 인구집단
 - 활성화 [99](#)
- 일치 채우기
 - 활성화 [99](#)
- 자동 업그레이드
 - Hub 서버의 [89](#)
 - 처리 서버 자동 업그레이드 실행 [96](#)
- 전체 프로필 구성 [49](#)
- 정리 함수
 - 테스트 [127](#)
- 제거
 - 리소스 키트 [102](#)
- 제목 영역에 대한 워크플로우
 - 사용자 지정, 첨부 파일 활성화 [168](#)
- 처리 서버
 - 그래픽 모드에서 업그레이드 [92](#)

- 처리 서버 (계속)
 - 등록 [127](#)
 - 삭제 및 추가 [127](#)
 - 업그레이드 다시 적용 [101](#)
 - 자동 속성 파일 [96](#)
 - 자동 업그레이드 [95](#)
 - 콘솔 모드에서 업그레이드 [94](#)
- 처리 서버 업그레이드
 - 로그 파일 [101](#)
 - 정보 [92](#)
- 첨부 파일
 - 사용자 지정 워크플로우 활성화 [160](#)
 - 활성화, 제목 영역에 대한 워크플로우 [168](#)
- 테스트
 - 사용자 지정 코드 [131](#)
 - 업그레이드 테스트 [131](#)
- 테이블스페이스
 - 생성 [27](#), [37](#)

O

- Oracle 구성
 - init.ora 매개 변수 설정 [27](#)
 - MDM Hub용 [27](#)
 - Oracle 휴지통 비활성화 [27](#)
- ORS(연산 참조 저장소)
 - 연결 업데이트 [126](#)
 - 연결 테스트 [126](#)

P

- postInstallSetup.log 파일 [90](#), [101](#)

S

- stopwords.txt 파일
 - 구성 [139](#)
- synonyms.txt 파일
 - 구성 [139](#)

T

- TNS 이름
 - 추가 [27](#)

U

- URN
 - ActiveVOS 설정 [157](#), [164](#)

W

- WebLogic
 - 구성 [52](#)
- WebSphere
 - 구성 [61](#)
- WebSphere 관리 보안
 - EAR 파일 제거 [112](#)
 - Hub 서버 PostInstallSetup 스크립트 실행 [113](#)
 - 처리 서버 PostInstallSetup 스크립트 실행 [113](#)
- WebSphere 구성
 - Informatica Data Director [72](#)

WebSphere 보안
ORS 등록 해제 [111](#)



검색
stopwords.txt 파일 [139](#)

검색 (계속)
synonyms.txt 파일 [139](#)
동의어 [139](#)
정지 단어 [139](#)
관계 기본 개체
기본 개체로 되돌리기 [153](#), [154](#)
기본 개체
관계 기본 개체에서 되돌리기 [153](#), [154](#)