



Informatica® Multidomain MDM
10.4 HotFix 2

Oracle Database with Red Hat JBoss **에 대한 설치 가이드**

Informatica Multidomain MDM Oracle Database with Red Hat JBoss 에 대한 설치 가이드

10.4 HotFix 2

2020년12월

© 저작권 Informatica LLC 2001, 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: 2021-02-19

목차

서문	7
Informatica 리소스	7
Informatica 네트워크	7
Informatica 기술 자료	7
Informatica 설명서	7
Informatica Product Availability Matrix	8
Informatica Velocity	8
Informatica Marketplace	8
Informatica 글로벌 고객 지원 센터	8
장 1: 설치 개요	9
Multidomain MDM 설치	9
설치 토폴로지	11
설치 태스크	11
장 2: 설치 전 태스크	12
설치 준비	12
환경 준비	13
데이터베이스 환경 설정	14
1단계. Oracle 설치 및 구성	15
2단계. 데이터베이스 및 테이블스페이스 생성	15
3단계. 데이터베이스 권한 및 연결 설정	18
4단계. ActiveVOS 스키마 생성	18
응용 프로그램 서버 환경 설정	20
Java Virtual Machine 구성	20
전체 프로필에 대한 서버 속성 구성	23
TLS(Transport Layer Security) 구성	25
보안 HTTP 응답 헤더	25
ActiveVOS 콘솔 관리자 생성	25
JBoss 시작	26
자동 설치를 위한 속성 파일 구성	27
Hub 서버 속성 파일 구성	27
처리 서버 속성 파일 구성	27
장 3: Hub 저장소 설치	28
Create the MDM Hub Master Database	28
연산 참조 저장소 생성	30
MDM Hub 마스터 데이터베이스로 메타데이터 가져오기	31
연산 참조 저장소로 메타데이터 가져오기	33

장 4: Hub 저장소 설치 후 태스크.....	35
Oracle 구성 요소에 대한 액세스 확인.	35
장 5: Hub 서버 설치.....	37
Hub 서버 설치.	37
설치 프로그램 워크플로우 검토.	38
설치 값 수집.	39
설치 마법사에서 Hub 서버 설치.	43
명령줄에서 Hub 서버 설치(UNIX만 해당).	44
자동으로 Hub 서버 설치.	45
클러스터의 노드에 Hub 서버 설치.	45
장 6: Hub 서버 설치 후 태스크.....	47
설치 로그 파일 복사.	48
버전 및 빌드 번호 확인.	48
Elasticsearch 설치 및 구성.	49
Hub 콘솔 클라이언트 구성(조건부).	49
MDM Hub 마스터 데이터베이스 이름 구성.	49
응용 프로그램 서버 설정 확인 및 구성(조건부).	50
응용 프로그램 서버 설정 편집.	50
JBoss 다중 노드 또는 클러스터 환경에 대한 Hub 서버 구성.	50
Hub 서버 응용 프로그램 배포(조건부).	51
스크립트를 사용하여 Hub 서버 응용 프로그램 배포(조건부).	52
수동으로 Hub 서버 응용 프로그램 배포(조건부).	53
1단계. 데이터 소스 생성.	53
2단계. JMS 메시지 대기열 구성.	55
3단계. Hub 서버 응용 프로그램 배포.	57
4단계. Hub 서버에 JMS 메시지 대기열 구성.	58
5단계. Informatica Data Director의 서버 리소스 구성.	60
메타데이터 캐시 구성(선택 사항).	61
Infinispan 특성 편집.	62
Hub 콘솔 시작.	62
연산 참조 저장소 등록.	63
Oracle Configuration Manager 등록(선택 사항).	65
ActiveVOS용 Oracle Configuration Manager 설정(선택 사항).	66
추가 응용 프로그램 서버 구성(선택 사항).	67
독립 실행형 처리 서버 인스턴스에 대한 JBoss 구성.	67
EJB 보안 구성.	67
장 7: 처리 서버 설치.....	70
처리 서버 설치.	70

설치 프로그램 워크플로우 검토.....	71
설치 값 수집.....	72
설치 마법사에서 처리 서버 설치.....	74
명령줄에서 처리 서버 설치(UNIX만 해당).....	75
자동으로 처리 서버 설치.....	76
클러스터의 노드에 처리 서버 설치.....	76

장 8: 처리 서버 설치 후 태스크..... 78

설치 로그 파일 복사.....	78
버전 및 빌드 번호 확인.....	79
처리 서버 응용 프로그램 배포(조건부).....	79
1단계. 데이터 소스 생성(조건부).....	80
2단계. 처리 서버 응용 프로그램 배포(조건부).....	80
처리 서버에 대한 보안 통신 활성화.....	81
Elasticsearch 설치 및 구성.....	81
일치 인구집단 구성.....	82
일치 인구집단 활성화.....	82
처리 서버와 정리 엔진 구성.....	83

장 9: 응용 프로그램 서버에 대한 ActiveVOS 설치 후 태스크..... 84

JBoss에서 ActiveVOS 보안 도메인 편집.....	84
JBoss 환경에서 트러스트된 사용자 생성.....	84

장 10: 비즈니스 항목 어댑터에 대한 ActiveVOS 설치 후 태스크..... 86

ActiveVOS 웹 응용 프로그램.....	86
비즈니스 항목 워크플로우 어댑터의 ActiveVOS URN 구성.....	87
ActiveVOS 프로토콜을 HTTPS로 설정.....	87
기본 워크플로우 엔진 구성.....	88
ActiveVOS의 MDM ID 서비스 구성.....	88
태스크 구성.....	89

장 11: ActiveVOS 사용자 지정..... 90

ActiveVOS 속성 추가.....	90
----------------------	----

장 12: 리소스 키트 설치..... 91

MDM Hub 샘플 연산 참조 저장소 설정.....	91
Informatica MDM Hub 샘플 연산 참조 저장소 등록.....	94
그래픽 모드에서 리소스 키트 설치.....	96
콘솔 모드에서 리소스 키트 설치.....	98
자동 모드에서 리소스 키트 설치.....	100
속성 파일 구성.....	101
자동 설치 프로그램 실행.....	102

장 13: MDM Hub 문제 해결.....	104
Troubleshooting the Installation Process.	104
장 14: 제거.....	110
제거 개요.....	110
Hub 저장소 제거.....	110
그래픽 모드에서 처리 서버 제거.....	111
UNIX의 그래픽 모드에서 처리 서버 제거.....	111
Windows의 그래픽 모드에서 처리 서버 제거.....	111
그래픽 모드에서 Hub 서버 제거.....	112
UNIX의 그래픽 모드에서 Hub 서버 제거.....	112
Windows의 그래픽 모드에서 Hub 서버 제거.....	112
그래픽 모드에서 리소스 키트 제거.....	112
UNIX의 그래픽 모드에서 리소스 키트 제거.....	112
Windows의 그래픽 모드에서 리소스 키트 제거.....	113
콘솔 모드에서 처리 서버 제거.....	113
콘솔 모드에서 Hub 서버 제거.....	113
콘솔 모드에서 리소스 키트 제거.....	114
수동으로 처리 서버 배포 해제.....	114
수동으로 Hub 서버 배포 해제.....	114
인덱스.....	116

서문

Informatica® *Multidomain MDM 설치 가이드*의 지침에 따라 선택한 지원되는 데이터베이스 및 응용 프로그램 서버 환경에서 Multidomain MDM을 설치하고 설정합니다. 이 가이드에는 설치 태스크 외에도 설치 전 태스크와 설치 후 태스크가 포함되어 있습니다.

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

설치 개요

이 장에 포함된 항목:

- [Multidomain MDM 설치, 9](#)
- [설치 토폴로지, 11](#)
- [설치 태스크, 11](#)

Multidomain MDM 설치

Multidomain MDM은 데이터 신뢰성 및 데이터 유지 관리 절차를 향상시키기 위한 마스터 데이터 관리 솔루션입니다. Multidomain MDM을 MDM Hub라고도 합니다. Hub 콘솔을 통해 MDM Hub 기능에 액세스할 수 있습니다.

MDM Hub는 다수의 구성 요소로 구성됩니다. 그래픽 모드, 콘솔 모드 또는 자동 모드로 MDM Hub를 설치할 수 있습니다.

핵심 구성 요소

다음 테이블에는 핵심 설치 구성 요소가 설명되어 있습니다.

구성 요소	설명
MDM Hub 마스터 데이터베이스	MDM Hub에 대한 비즈니스 데이터를 저장 및 통합하는 스키마입니다. 사용자 계정, 보안 구성, 연산 참조 저장소 레지스트리 및 메시지 대기열 설정과 같은 MDM Hub 환경 구성 설정이 포함됩니다. MDM Hub 마스터 데이터베이스에서 연산 참조 저장소에 대해 액세스하고 관리할 수 있습니다. MDM Hub 마스터 데이터베이스의 기본 이름은 CMX_SYSTEM이지만 사용자 지정 이름을 사용할 수 있습니다. 동일한 데이터베이스 인스턴스에 개별 연산 참조 저장소 집합을 보유한 MDM Hub 마스터 데이터베이스를 여러 개 생성할 수 있습니다.
연산 참조 저장소	MDM Hub에 대한 비즈니스 데이터를 저장 및 통합하는 스키마입니다. 마스터 데이터, 콘텐츠 메타데이터 및 마스터 데이터를 처리하고 관리하기 위한 규칙을 포함합니다. 다양한 지역, 다양한 조직 부서, 개발 및 프로덕션 환경에 별도의 연산 참조 저장소 데이터베이스를 구성할 수 있습니다. 연산 참조 저장소 데이터베이스를 여러 서버 시스템으로 분산할 수 있습니다. 연산 참조 저장소의 기본 이름은 CMX_ORS입니다.
Hub 서버	응용 프로그램 서버에 배포하는 J2EE 응용 프로그램입니다. Hub 서버는 MDM Hub 내에서 저장된 데이터를 처리하고 MDM Hub를 외부 응용 프로그램과 통합합니다. Hub 서버는 MDM Hub에 대한 핵심 및 공통 서비스를 관리합니다.

구성 요소	설명
처리 서버	응용 프로그램 서버에 배포하는 J2EE 응용 프로그램입니다. 처리 서버는 로드, BVT 다시 계산 및 유효성 다시 검사와 같은 일괄 작업을 처리하고 데이터 정리 및 일치 작업을 수행합니다. 처리 서버는 일치 및 통합을 위해 데이터를 표준화 및 최적화하기 위해 구성하는 정리 엔진과 상호 작용합니다.
프로비저닝 도구	비즈니스 항목 모델을 작성하고 Data Director에 대한 Entity 360 프레임워크를 구성하기 위한 도구입니다. 비즈니스 항목 모델을 작성한 후 MDM Hub에 구성을 게시할 수 있습니다.
Informatica ActiveVOS [®]	MDM Hub 내부에서 데이터를 처리할 때 필요한 BPM(비즈니스 프로세스 관리) 도구입니다. Informatica ActiveVOS는 데이터에 대한 변경 승인 프로세스를 포함한 자동화된 비즈니스 프로세스를 지원합니다. BVT(최선의 진실, Best Version of the Truth) 레코드에 포함되기 전에 마스터 데이터에 대한 변경 내용이 검토 및 승인 프로세스를 거처도록 하려면 Informatica ActiveVOS를 사용해도 됩니다. Hub 서버 설치의 일부로 ActiveVOS Server를 설치하면 ActiveVOS Server, ActiveVOS Console 및 Process Central이 설치됩니다. 또한 미리 정의된 MDM 워크플로우, 태스크 및 역할도 설치됩니다.
IDD(Data Director)	MDM Hub에 저장된 데이터의 마스터 및 관리를 위한 사용자 인터페이스입니다. IDD에서 데이터는 고객, 공급자 및 직원과 같은 비즈니스 항목으로 구성됩니다. 비즈니스 항목은 조직에 대한 중요성을 가진 데이터 그룹입니다.

선택적 구성 요소

다음 테이블에는 선택적 설치 구성 요소가 설명되어 있습니다.

구성 요소	설명
리소스 키트	MDM Hub를 응용 프로그램 및 워크플로우에 통합하는 데 사용되는 샘플, 응용 프로그램 및 유틸리티의 집합입니다. 설치할 리소스 키트 구성 요소를 선택할 수 있습니다.
Dynamic Data Masking	중요 정보에 대한 무단 액세스를 방지하도록 MDM Hub와 데이터베이스 사이에서 작동하는 데이터 보안 도구입니다. Dynamic Data Masking은 데이터베이스에 전송된 요청을 가로채서 이 요청이 다시 MDM Hub로 전송되기 전에 데이터를 마스킹하도록 데이터 마스킹 규칙을 해당 요청에 적용합니다.
IDC(Informatica Data Control)	제목 영역 데이터 모델에 따라 IDD(Informatica Data Director)에만 적용됩니다. IDC는 비즈니스 사용자에게 의해 사용되는 타사 응용 프로그램의 MDM Hub 데이터를 공개하는 일련의 사용자 인터페이스 컨트롤입니다.
ZDT(Zero Downtime) 모듈	MDM Hub 업그레이드 중 응용 프로그램에 MDM Hub의 데이터에 대한 액세스 권한이 있도록 하는 모듈입니다. ZDT 환경에서 소스 데이터베이스와 대상 데이터베이스를 복제합니다. MDM Hub 업그레이드 중 ZDT 모듈은 소스 데이터베이스의 데이터 변경 내용을 대상 데이터베이스로 복제합니다. ZDT 모듈을 구입하려면 Informatica 영업 담당자에게 문의하십시오. ZDT(Zero Downtime) 환경 설치에 대한 자세한 내용은 데이터베이스에 대한 <i>Multidomain MDM ZDT(Zero Downtime) 설치 가이드</i> 를 참조하십시오.

설치 토폴로지

MDM Hub를 설치하기 전에 설치 토폴로지를 결정합니다. 일반적으로, 구현할 토폴로지는 인프라 계획자와 마스터 데이터 관리 솔루션 아키텍처 설계자가 결정합니다.

개발, 테스트 및 프로덕션과 같은 여러 환경에 MDM Hub를 설치할 수 있습니다. 각 환경 유형의 요구 사항 및 우선 순위는 고유합니다. 따라서 설치 토폴로지는 환경별로 다릅니다.

다음 테이블에는 사용할 수 있는 MDM Hub 설치 토폴로지가 설명되어 있습니다.

토폴로지	설명
독립 실행형 응용 프로그램 서버 인스턴스	모든 MDM Hub 구성 요소가 독립 실행형 응용 프로그램 서버 인스턴스에 설치됩니다.
여러 응용 프로그램 서버 인스턴스	MDM Hub 구성 요소가 여러 응용 프로그램 서버 인스턴스에 설치됩니다.
응용 프로그램 서버 클러스터	MDM Hub 구성 요소가 응용 프로그램 서버 클러스터에 설치됩니다.

설치 토폴로지에 대한 자세한 내용은 *Multidomain MDM 인프라 계획 가이드* 항목을 참조하십시오.

참고: MDM Hub 구현 내 모든 구성 요소의 버전이 같아야 합니다. 여러 MDM Hub 버전이 있는 경우 각 버전은 별도의 환경에 설치합니다.

설치 태스크

MDM Hub 구성 요소를 설치하기 전에 설치 전 태스크를 완료합니다. 설치한 후 설치 후 태스크를 완료합니다.

MDM Hub를 설치하려면 다음 태스크를 수행합니다.

1. 설치 전 태스크를 완료합니다. Hub 서버 및 처리 서버에 대한 설치 프로그램을 성공적으로 실행하고 Hub 저장소를 생성할 수 있으려면 설치 전 태스크를 완료합니다.
2. MDM Hub 마스터 데이터베이스를 생성합니다. Hub 서버 및 처리 서버를 설치하기 전에 MDM Hub 마스터 데이터베이스를 생성합니다.
MDM Hub 배포와 함께 제공된 설정 스크립트를 사용하여 MDM Hub 마스터 데이터베이스를 생성합니다.
3. 연산 참조 저장소를 생성합니다. 설치 전 태스크를 완료한 후 언제든지 연산 참조 저장소를 생성합니다.
MDM Hub 배포와 함께 제공된 설정 스크립트를 사용하여 연산 참조 저장소를 생성합니다.
4. Hub 서버를 설치합니다. MDM Hub 설치 프로그램을 사용하여 Hub 서버를 설치합니다.
5. 처리 서버를 설치합니다. MDM Hub 설치 프로그램을 사용하여 처리 서버를 설치합니다.
6. 설치 후 구성 태스크를 수행합니다. 데이터베이스 연결을 테스트합니다. MDM Hub 기능을 사용할 수 있도록 하려면 Hub 서버 및 처리 서버를 구성합니다.
7. 필요한 경우, 리소스 키트를 설치합니다. MDM Hub 설치 프로그램을 사용하여 리소스 키트를 설치합니다.

제 2 장

설치 전 태스크

이 장에 포함된 항목:

- [설치 준비, 12](#)
- [환경 준비, 13](#)
- [데이터베이스 환경 설정, 14](#)
- [응용 프로그램 서버 환경 설정, 20](#)
- [자동 설치를 위한 속성 파일 구성, 27](#)

설치 준비

MDM Hub를 설치하기 전에 설치를 준비합니다.

다음 테이블에는 설치를 위한 준비 태스크가 설명되어 있습니다.

태스크	설명
릴리스 노트 읽기	최신 <i>Multidomain MDM 릴리스 노트</i> 에서 설치 및 업그레이드 프로세스에 대한 업데이트 내용을 읽습니다. 중요: 일부 버전의 응용 프로그램 서버 및 데이터베이스에는 Multidomain MDM을 실행할 때 알려진 제한이 있습니다. 제안되는 모든 해결 방법을 수행하십시오.
PAM(Product Availability Matrix) 읽기	PAM(Product Availability Matrix)을 읽고 제품 요구 사항 및 지원되는 플랫폼에 대한 정보를 확인합니다. PAM은 https://network.informatica.com/community/informatica-network/product-availability-matrices 에서 액세스할 수 있습니다.
MDM 인프라 및 아키텍처 계획 이해	조직의 인프라 계획자 또는 MDM 솔루션 아키텍처 설계자로부터 MDM 인프라 및 아키텍처에 대한 계획을 받고 이해합니다. 인프라 계획 및 MDM 아키텍처에 대한 자세한 내용은 <i>Multidomain MDM 인프라 계획 가이드</i> 항목을 참조하십시오.
설치 프로그램 파일 다운로드 및 추출	Informatica Electronic Software Download 사이트에서 설치 파일을 시스템의 디렉터리에 다운로드합니다. 압축된 파일을 추출하려면 빈 폴더도 추출하는 추출 유틸리티를 사용합니다. 다음 설치 파일을 다운로드 및 추출합니다. <ul style="list-style-type: none">- 운영 체제에 대한 MDM Hub 설치 프로그램- 데이터베이스 파일- 운영 체제에 대한 ActiveVOS Server 설치 프로그램

태스크	설명
라이선스 키 확인	Informatica의 전자 메일 메시지에 있는 라이선스 키가 있는지 확인합니다. 제품을 설치하는 사용자 계정에서 액세스할 수 있는 디렉터리에 라이선스 키 파일을 복사합니다. 라이선스 키가 없으면 Informatica 글로벌 고객 지원 센터에 문의하십시오.
설치 기록 디렉터리 생성	유효성 검사 결과, 환경 보고서, 데이터베이스 디버그 로그 및 로그 파일과 같은 설치 파일의 사본을 저장하기 위한 디렉터를 생성합니다. 예를 들어 install_doc 디렉터를 생성합니다. 설치 문제를 해결하려면 분석을 위해 설치 기록 디렉터리의 보관 파일을 생성하여 Informatica 글로벌 고객 지원 센터로 보낼 수 있습니다.

환경 준비

MDM Hub를 설치하기 전에 설치 환경을 준비합니다.

다음 테이블에는 설치를 위한 환경을 준비하기 위해 수행하는 태스크가 설명되어 있습니다.

태스크	설명
최소 시스템 요구 사항 확인	시스템이 MDM Hub 설치를 위한 하드웨어 및 소프트웨어 요구 사항을 충족하는지 확인합니다. 하드웨어 요구 사항은 데이터, 처리 볼륨 및 비즈니스 규칙에 따라 다릅니다. MDM Hub를 설치하려면 시스템이 다음과 같은 최소 요구 사항을 충족해야 합니다. - 디스크 공간. 4.9GB - 개발 환경을 위한 RAM. 4GB MDM Hub 구성 요소의 런타임 실제 메모리 요구 사항을 확인하려면 다음 수식을 사용합니다. Total run-time memory requirement for MDM Hub components = JDK JVM max heap size of the application server + operating system native heap size
JDK(Java 개발 키트) 설치	MDM Hub를 설치할 시스템에 지원되는 버전의 JDK를 설치합니다. JDK는 MDM Hub 설치 프로그램과 함께 번들로 제공되지 않습니다. JBoss 환경에서 Informatica가 제공하는 Azul Zulu JDK의 버전을 설치해야 합니다. Azul Zulu JDK를 다운로드하려면 Informatica Shipping에 문의하십시오. 참고: Hub 콘솔을 실행할 시스템 및 응용 프로그램 서버 시스템에 동일한 Java 버전을 사용합니다.
Visual Studio 2015용 Visual C++ 재배포 가능 패키지 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.

태스크	설명
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	<p>Hub 서버, MDM Hub 마스터 데이터베이스, 연산 참조 저장소 및 Hub 콘솔에 대한 동일한 운영 체제 로컬을 설정합니다.</p>
UNIX에서 X Window System 설치	<p>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.</p>
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>
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>

제품 요구 사항 및 지원되는 플랫폼에 대한 자세한 내용은 Informatica 네트워크의 PAM(Product Availability Matrix)을 참조하십시오.

<https://network.informatica.com/community/informatica-network/product-availability-matrices>

데이터베이스 환경 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 생성

ActiveVOS를 설치하려면 ActiveVOS 스키마를 생성해야 합니다. 스키마를 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.

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.

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.

- 스키마를 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.

응용 프로그램 서버 환경 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.

Java 옵션	설명
-Didd.mdm.host -Didd.mdm.port -Didd.protocol	제목 영역이 있는 Data Director에 필요합니다. 제목 영역이 있는 Data Director를 배포하려면 다음 Java 옵션을 설정합니다. <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.
-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.
-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.

Java 옵션	설명
-DUseESLegacyFqSearch	필드가 지정된 검색에서 비즈니스 항목 유형에 대한 하위 노드 내의 정확한 일치치를 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.
-Dstricttransportsecurity.flag	웹 브라우저에서 HTTP 요청을 사용한 모든 Data Director 액세스 시도를 HTTPS 요청으로 변환할지 여부를 지정합니다. true로 설정합니다.
-Xms	초기 힙 크기입니다. 2048m으로 설정합니다.
-Xmx	최대 JVM 힙 크기입니다. 6GB 이상으로 설정합니다. 예를 들어 -Xmx를 6144m으로 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 512m으로 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="20000000"/>
        <https-listener name="https" socket-binding="https" security-realm="ApplicationRealm" enable-
http2="true" max-post-size="20000000"/>
    </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>
```


TLS(Transport Layer Security) 구성

중요: 보안 통신을 위해서는 Multidomain MDM을 배포한 응용 프로그램에서 TLS 1.0을 비활성화하고 TLS 1.2를 활성화합니다.

TLS 구성에 대한 자세한 내용은 사용 중인 응용 프로그램 서버의 설명서를 참조하십시오.

보안 HTTP 응답 헤더

응용 프로그램 서버 환경을 보호하기 위해 보안 HTTP 응답 헤더를 사용합니다. X-Powered-By 및 서버와 같은 헤더를 변경하거나 제거합니다.

HTTP 응답 헤더 변경 또는 제거에 대한 자세한 내용은 사용 중인 응용 프로그램 서버의 설명서를 참조하십시오.

ActiveVOS 콘솔 관리자 생성

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.

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.

자동 설치를 위한 속성 파일 구성

자동 모드에서 사용자 상호 작용 없이 Hub 서버 및 처리 서버를 설치하려는 경우 설치 속성 파일을 구성합니다. 다중 설치가 필요한 경우나 시스템 클러스터에 설치해야 하는 경우 자동 설치를 수행하고자 할 수 있습니다. 자동 설치에서는 어떠한 진행률 또는 실패 메시지도 표시하지 않습니다.

설치 프로그램은 자동 설치 속성 파일을 읽어 설치 옵션을 확인합니다. 설정이 잘못된 경우에도 자동 설치 프로세스가 성공적으로 완료될 수 있기 때문에 속성 파일에서 올바른 설정을 제공해야 합니다.

다음과 같은 자동 설치 속성 파일을 구성할 수 있습니다.

- Hub 서버. 자동 모드에서 Hub 서버를 설치하는 데 필요합니다.
- 처리 서버. 자동 모드에서 처리 서버를 설치하는 데 필요합니다.

참고: 자동 설치 속성 파일을 수동으로 구성하지 않으려는 경우 `-r` 명령줄 옵션을 사용하여 설치 중에 자동 설치 속성 파일을 생성할 수 있습니다.

Hub 서버 속성 파일 구성

자동 모드에서 Hub 서버를 설치하려는 경우 Hub 서버 속성 파일을 구성합니다. 속성 파일에서 설치 옵션을 지정하고 새 이름으로 파일을 저장합니다.

1. 다음 디렉터리에 있는 `silentInstallServer_sample.properties` 파일을 찾습니다. `/silent_install/mrmserver`
2. `silentInstallServer_sample.properties` 파일의 백업 사본을 생성합니다.
3. 텍스트 편집기에서 파일을 열고 설치 매개 변수의 값을 구성합니다.
4. `silentInstallServer.properties`와 같은 새 이름으로 속성 파일을 저장합니다.

처리 서버 속성 파일 구성

자동 모드에서 처리 서버를 설치하려는 경우 Hub 서버 속성 파일을 구성합니다. 속성 파일에서 설치 옵션을 지정하고 새 이름으로 파일을 저장합니다.

1. 다음 디렉터리에 있는 `silentInstallCleanse_sample.properties` 파일을 찾습니다. `/silent_install/mrmcleanse`
2. `silentInstallCleanse_sample.properties` 파일의 백업 사본을 생성합니다.
3. 텍스트 편집기에서 파일을 열고 설치 매개 변수의 값을 구성합니다.
4. `silentInstallCleanse.properties`와 같은 이름으로 속성 파일을 저장합니다.

제 3 장

Hub 저장소 설치

이 장에 포함된 항목:

- [Create the MDM Hub Master Database, 28](#)
- [연산 참조 저장소 생성, 30](#)
- [MDM Hub 마스터 데이터베이스로 메타데이터 가져오기, 31](#)
- [연산 참조 저장소로 메타데이터 가져오기, 33](#)

Create the MDM Hub Master Database

After you install Oracle, create an MDM Hub 마스터 데이터베이스. If you want to create multiple MDM Hub Master Databases, ensure that you create each MDM Hub Master Database in a different schema. The default name of the MDM Hub Master Database is CMX_SYSTEM, but you can use a custom name.

Note: If you change the folder names in the distribution directory, metadata import fails.

If you want to use the Oracle multitenant feature, create the MDM Hub Master Database in a pluggable database (PDB).

1. Open a command prompt, and navigate to the following directory:
 <MDM Hub 배포 디렉터리>/database/bin
2. To create the MDM Hub Master Database, run the following command:
 On UNIX. ./sip_ant.sh create_system
 On Windows. sip_ant.bat create_system
3. Answer the prompts described in the following table:

Note: The prompt displays default text in brackets. Press **Enter** to use the default value and go to the next prompt.

Prompts	Description
Enter the database type (ORACLE, MSSQL, DB2)	Database type. Specify ORACLE .
Enter Oracle Connection Type (service, sid). [service]	Connection type. Use one of the following values: - SERVICE. Uses the service name to connect to Oracle. - SID. Uses the Oracle System ID to connect to Oracle. Default is SERVICE .
Enter master user name [cmx_system]	User name to access the MDM Hub Master Database. Default is cmx_system .
Enter the database host name [localhost]	Name of the host that runs the database. Default is localhost . Important: In cluster environments, specify the absolute host name or the IP address to avoid caching issues.
Enter the database port number [1521]	Port number that the database listener uses. Default is 1521 .
Enter the database service name [orcl]	Name of the Oracle service. This prompt is displayed if the selected Oracle connection type is SERVICE. Default is orcl .
Enter database SID [orcl]	Name of the Oracle System ID. This prompt is displayed if the selected Oracle connection type is SID.
Enter Oracle Net connect identifier (TNS Name). [orcl]	TNS name that is used to connect to the Oracle database. Default is orcl .
Master Database Connect URL: "jdbc:oracle:thin:@// <host_name>:<port>/<service_name>".	Connect URL for the Oracle connection type SERVICE.
Enter master database user password	Password to access the MDM Hub Master Database.
Enter locale name from the list: de, en_US, fr, ja, ko, zh_CN. [en_US]	Operating system locale. Default is en_US .
Enter the DBA user name [SYS]	User name of the administrative user. Default is SYS .
Enter the DBA password	Password of the administrative user.
Enter the name of the MDM index tablespace [CMX_INDX]	Name of the tablespace to contain the index components for the MDM Hub 마스터 데이터베이스 . Default is CMX_INDX .

Prompts	Description
Enter the name of the MDM temporary tablespace (Not an Oracle temporary tablespace) [CMX_TEMP]	Name of the tablespace to contain the temporary components for the MDM Hub 마스터 데이터베이스. Default is CMX_TEMP .
Enter the name of the Oracle temporary tablespace [TEMP]	Name of the Oracle temporary tablespace. Default is TEMP .

4. To verify that the MDM Hub Master Database was created successfully, review sip_ant.log in the following directory:

<MDM Hub 배포 디렉터리>/database/bin

The sip_ant.log file logs any errors that might occur when you run the sip_ant script to create the MDM Hub 마스터 데이터베이스.

연산 참조 저장소 생성

설치 전 테스트를 완료한 후 ORS(연산 참조 저장소)를 생성합니다. ORS의 기본 이름은 CMX_ORS입니다.

참고: If you change the folder names in the distribution directory, metadata import fails.

Oracle 멀티테넌트 기능을 사용하려는 경우 PDB(플러깅 가능한 데이터베이스)에 ORS를 생성합니다.

1. 명령 프롬프트를 열고 다음 디렉터리로 이동합니다.

<MDM Hub 배포 디렉터리>/database/bin

2. ORS를 생성하려면 다음 명령을 실행합니다.

UNIX의 경우. ./sip_ant.sh create_ors

Windows의 경우. sip_ant.bat create_ors

3. 다음 테이블에 설명되어 있는 프롬프트에 응답합니다.

참고: 프롬프트에서 대괄호 안에 기본 텍스트를 표시합니다. **Enter** 키를 눌러 기본값을 사용하고 다음 프롬프트로 이동합니다.

프롬프트	설명
데이터베이스 유형(ORACLE, MSSQL, DB2) 입력	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 스키마 호스트 이름 입력 [localhost]	데이터베이스를 실행 중인 호스트의 이름입니다. 기본값은 localhost입니다.

프롬프트	설명
연산 참조 저장소 스키마 포트 번호 입력 [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 사용자 이름 입력 [cmx_ors]	연산 참조 저장소에 액세스하는 데 필요한 사용자 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]	관리자의 사용자 이름입니다. 기본값은 SYS입니다.
DBA 암호 입력	관리자의 암호입니다.
MDM 인덱스 테이블스페이스의 이름 입력 [CMX_INDX]	연산 참조 저장소에 대한 인덱스 구성 요소를 포함할 테이블스페이스의 이름입니다. 기본값은 CMX_INDX입니다.
MDM 임시 테이블스페이스의 이름 입력 [CMX_TEMP]	연산 참조 저장소에 대한 임시 구성 요소를 포함할 테이블스페이스의 이름입니다. 기본값은 CMX_TEMP입니다.
Oracle 임시 테이블스페이스의 이름 입력 [TEMP]	Oracle 임시 테이블스페이스의 이름입니다. 기본값은 TEMP입니다.

4. ORS가 성공적으로 생성되었는지 확인하려면 다음 디렉터리에서 sip_ant.log를 검토합니다.

<MDM Hub 배포 디렉터리>/database/bin

sip_ant.log 파일은 sip_ant 스크립트를 실행하여 ORS를 생성할 때 발생할 수 있는 모든 오류를 기록합니다.

MDM Hub 마스터 데이터베이스로 메타데이터 가져오기

MDM Hub 마스터 데이터베이스를 생성한 후 MDM Hub 마스터 데이터베이스로 초기 메타데이터를 가져옵니다. 초기 메타데이터에는 리포지토리 테이블과 MDM Hub이 Hub 저장소에서 필요로 하는 다른 개체가 포함됩니다.

참고: If you change the folder names in the distribution directory, metadata import fails.

- 명령 프롬프트를 열고 다음 디렉터리로 이동합니다.

<MDM Hub 배포 디렉터리>/database/bin

- 초기 메타데이터를 가져오려면 다음 명령을 실행합니다.

UNIX의 경우. `./sip_ant.sh import_system`

Windows의 경우. `sip_ant.bat import_system`

3. 다음 테이블에 설명되어 있는 프롬프트에 응답합니다.

참고: 프롬프트에서 대괄호 안에 기본 텍스트를 표시합니다. **Enter** 키를 눌러 기본값을 사용하고 다음 프롬프트로 이동합니다.

프롬프트	설명
데이터베이스 유형(ORACLE, MSSQL, DB2) 입력	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.
마스터 사용자 이름 입력 [cmx_system]	MDM Hub 마스터 데이터베이스에 액세스하는 데 필요한 사용자 이름입니다. 기본값은 cmx_system입니다.
데이터베이스 호스트 이름 입력 [localhost]	데이터베이스를 실행 중인 호스트의 이름입니다.
데이터베이스 포트 번호 입력 [1521]	데이터베이스 수신기에서 사용하는 포트 번호입니다. 기본값은 1521입니다.
MDM Hub 마스터 데이터베이스 서비스 이름 입력 [orcl]	Oracle 서비스의 이름입니다. 선택한 Oracle 연결 유형이 SERVICE이면 이 프롬프트가 표시됩니다. 기본값은 orcl입니다.
데이터베이스 SID 입력	Oracle 시스템 ID의 이름입니다. 이 프롬프트는 선택한 Oracle 연결 유형이 SID이면 표시됩니다.
Oracle Net 연결 식별자(TNS 이름)를 입력하십시오. [orcl]	Oracle 데이터베이스에 연결하는 데 사용되는 TNS 이름입니다. 기본값은 orcl입니다.
마스터 데이터베이스 연결 URL: "jdbc:oracle:thin:@//<host_name>:<포트>/ <service_name>"	Oracle 연결 유형 SERVICE의 연결 URL입니다.
마스터 데이터베이스 사용자 암호 입력	MDM Hub 마스터 데이터베이스에 액세스하는 데 필요한 암호입니다.
목록(de, en_US, fr, ja, ko, zh_CN)에서 로캘 이름 입력 [en_US]	운영 체제 로캘입니다. 기본값은 en_US입니다.

4. 초기 메타데이터를 가져온 후 다음 로그 파일에서 오류를 검토합니다.

- `seed.log`. 데이터베이스 오류를 포함합니다.
`seed.log` 파일은 다음 디렉터리에 있습니다. <MDM Hub 설치 디렉터리>/database/bin/oracle
- `sip_ant.log`. 사용자 입력 오류를 포함합니다.
`sip_ant.log` 파일은 다음 디렉터리에 있습니다. <배포 디렉터리>/database/bin

연산 참조 저장소로 메타데이터 가져오기

연산 참조 저장소를 생성한 후 연산 참조 저장소로 초기 메타데이터를 가져옵니다. 초기 메타데이터에는 리포트 토리 테이블과 MDM Hub이 Hub 저장소에서 필요로 하는 다른 개체가 포함됩니다.

참고: If you change the folder names in the distribution directory, metadata import fails.

1. 명령 프롬프트를 열고 다음 디렉터리로 이동합니다.

<MDM Hub 배포 디렉터리>/database/bin

2. 초기 메타데이터를 가져오려면 다음 명령을 실행합니다.

UNIX의 경우. ./sip_ant.sh import_ors

Windows의 경우. sip_ant.bat import_ors

3. 다음 테이블에 설명되어 있는 프롬프트에 응답합니다.

참고: 프롬프트에서 대괄호 안에 기본 텍스트를 표시합니다. **Enter** 키를 눌러 기본값을 사용하고 다음 프롬프트로 이동합니다.

프롬프트	설명
데이터베이스 유형(ORACLE, MSSQL, DB2) 입력	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 입력하십시오.[1521]	데이터베이스 수신기에서 사용하는 포트 번호입니다. 기본값은 1521입니다.
연산 참조 저장소 데이터베이스 서비스 이름 입력	Oracle 서비스의 이름입니다. 선택한 Oracle 연결 유형이 SERVICE이면 이 프롬프트가 표시됩니다.
Oracle Net 연결 식별자(TNS 이름)를 입력하십시오. [orcl]	Oracle 데이터베이스에 연결하는 데 사용되는 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	Oracle 시스템 ID의 이름입니다. 이 프롬프트는 선택한 Oracle 연결 유형이 SID이면 표시됩니다.
연산 참조 저장소 데이터베이스 사용자 이름 입력 [cmx_ors]	연산 참조 저장소에 액세스하는 데 필요한 사용자 이름입니다. 기본값은 cmx_ors입니다.

프롬프트	설명
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. 기본값은 en_US입니다.
연산 참조 저장소 시간 표시 막대 세분화 정수 코드(연: 5, 월: 4, 일: 3, 시: 2, 분: 1, 초: 0) 입력 [3]	<p>사용할 시간 표시 막대 단위를 지정합니다. 기본값은 일입니다.</p> <p>참고: 구성된 시간 표시 막대 세분성은 나중에 변경할 수 없습니다.</p> <p>시간 표시 막대에 대한 자세한 내용은 <i>Multidomain MDM 구성 가이드</i>를 참조하십시오.</p>

4. 초기 메타데이터를 가져온 후 다음 로그 파일에서 오류를 검토합니다.

- **seed.log.** 데이터베이스 오류를 포함합니다.
seed.log 파일은 다음 디렉터리에 있습니다. <MDM Hub 설치 디렉터리>/database/bin/oracle
- **sip_ant.log.** 사용자 입력 오류를 포함합니다.
sip_ant.log 파일은 다음 디렉터리에 있습니다. <MDM Hub 배포 디렉터리>/database/bin

제 4 장

Hub 저장소 설치 후 태스크

이 장에 포함된 항목:

- [Oracle 구성 요소에 대한 액세스 확인, 35](#)

Oracle 구성 요소에 대한 액세스 확인

Hub 저장소에 필수 Oracle 구성 요소에 대한 액세스 권한이 있어야 합니다.

Hub 저장소에는 다음 Oracle 구성 요소에 대한 액세스 권한이 필요합니다.

Oracle Java Virtual Machine

Oracle XML Database

ALL_CONSTRAINTS

ALL_CONS_COLUMNS

ALL_DIRECTORIES

ALL_INDEXES

ALL_IND_COLUMNS

ALL_JOBS(마이그레이션에 사용됨)

ALL_TABLES

ALL_TAB_COLUMNS

ALL_VIEWS

DBMS_APPLICATION_INFO

DBMS_JOB

DBMS_OUTPUT

DBMS_STANDARD

DBMS_SQL

DBMS_STATS

DBMS_UTILITY

DUAL

PLITBLM

STANDARD

SYS_STUB_FOR_PURITY_ANALYSIS

USER_CONSTRAINTS
USER_CONS_COLUMNS
USER_EXTERNAL_TABLES
USER_INDEXES
USER_JAVA_POLICY
USER_OBJECTS
USER_SEQUENCES
USER_SOURCE
USER_TABLES
USER_TAB_COLS
USER_TAB_COLUMNS
USER_TRIGGERS
UTL_FILE
V\$NLS_PARAMETERS
V\$VERSION

제 5 장



Hub 서버 설치

이 장에 포함된 항목:

- [Hub 서버 설치, 37](#)
- [설치 프로그램 워크플로우 검토, 38](#)
- [설치 값 수집, 39](#)
- [설치 마법사에서 Hub 서버 설치, 43](#)
- [명령줄에서 Hub 서버 설치\(UNIX만 해당\), 44](#)
- [자동으로 Hub 서버 설치, 45](#)
- [클러스터의 노드에 Hub 서버 설치, 45](#)

Hub 서버 설치

설치 마법사나 자동 설치 스크립트를 사용하여 또는 UNIX 시스템의 경우 명령줄 스크립트를 사용하여 Hub 서버를 설치할 수 있습니다. 설치 프로그램을 시작하기 전에 설치 전 태스크를 완료하고 필요한 정보를 수집한 경우 설치 프로세스에는 약 15분이 소요됩니다.

	중지! 설치 전 태스크를 완료했습니까? 설치 프로그램을 실행하기 전에 설치 전 태스크를 완료하지 않았다면 설치가 실패합니다.
	설치 준비 검사 목록 <ul style="list-style-type: none"><input type="checkbox"/> MDM 구현 계획을 생성했습니다.<input type="checkbox"/> 서버가 시스템 요구 사항을 충족하는지 확인했습니다.<input type="checkbox"/> 운영 체제 및 소프트웨어 버전이 지원되는지 확인했습니다.<input type="checkbox"/> 운영 체제 및 소프트웨어 버전의 알려진 제한을 검토했습니다.<input type="checkbox"/> 지원되는 응용 프로그램 서버 버전을 설치하고 구성했습니다.<input type="checkbox"/> 지원되는 데이터베이스 관리 시스템 버전을 설치하고 구성했습니다.<input type="checkbox"/> 환경에 맞는 설치 전 구성 태스크를 수행했습니다.<input type="checkbox"/> 액세스 가능한 위치에 MDM 라이선스 파일을 저장했습니다.

누락된 태스크가 있는 경우 태스크 완료에 대한 도움을 받으려면 이전 장으로 돌아가십시오.

진행할 준비가 되었으면 다음 단계를 수행합니다.

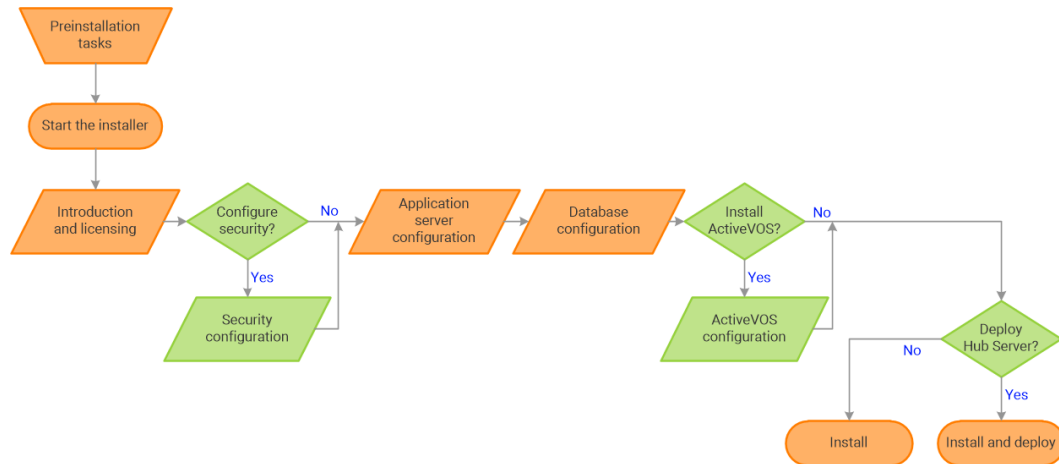
1. 설치 워크플로우를 검토합니다.

2. 설치 중에 입력할 값을 수집합니다.
3. 설치 마법사나 명령줄을 사용하여 또는 자동 모드에서 설치를 시작합니다.

설치 프로그램 워크플로우 검토

설치 마법사, 명령줄 프롬프트 또는 자동 설치 스크립트 중 무엇을 사용하든 설치 프로그램은 동일한 순서로 단계를 진행합니다. 결정 지점을 염두에 두고 설치 워크플로우를 따라야 합니다.

다음 다이어그램은 Hub 서버 설치 프로그램 워크플로우의 단계를 안내합니다.



설치 프로세스에서 결정 지점을 식별하십시오. 구현 계획을 참조하여 다음과 같은 각 결정 지점에서 어떤 경로를 선택할지 이해해야 합니다.

1. **보안을 구성합니까?** 기본 보안을 사용하거나 암호 해싱 알고리즘 및 인증서 공급자를 지정하여 사용자 고유의 보안을 구성합니다.
2. **ActiveVOS를 설치합니까?** 관리자는 Informatica ActiveVOS를 사용하여 검토 워크플로우를 설정합니다. 권한이 있는 사용자는 마스터 데이터가 업데이트되기 전에 레코드와 관련하여 제안된 변경 내용의 유효성을 검사할 수 있습니다.

참고: Informatica ActiveVOS에 대한 설치를 이미 완료했다면 Hub 서버 설치 중에 이를 다시 설치할 필요가 없습니다.

3. **Hub 서버를 배포합니까?** 설치 프로그램에서 postInstallSetup 스크립트를 실행하도록 허용합니다. 다른 중요 태스크 중에서 스크립트는 Hub 서버를 응용 프로그램 서버에 배포합니다. 또는 설치 프로그램을 종료한 후에 postInstallSetup 스크립트를 실행할 수 있습니다.

설치 값 수집

설치 프로그램을 실행하기 전에 설치 값을 수집합니다. 설치 프로그램이 응용 프로그램 서버, 데이터베이스 관리 시스템 및 기타 구성 요소에 대한 정보를 묻습니다. 가장 좋은 방법은 이러한 테이블을 인쇄하고 해당 환경의 값을 추가하는 것입니다.

응용 프로그램 서버: Red Hat JBoss

설치에 필요한 JBoss 세부 정보를 수집하려면 다음 테이블을 사용합니다.

속성	설명	기본값	서버 인스턴스 1 값
JBoss 설치 디렉터리	JBoss가 설치된 위치입니다. 참고: bin 디렉터리를 포함하지 마십시오.	-	
구성 이름	기본 구성 이름을 사용합니다. Hub 서버가 지정된 구성에 배포됩니다.	독립 실행형	독립 실행형
원격 포트	JBoss 원격 포트입니다. 클러스터된 환경에서 여러 포트를 식별합니다. 참고: JBoss 원격 포트가 기본 데이터베이스 서버 포트와 충돌할 수 있습니다. 포트가 충돌하는 경우 포트 중 하나를 변경하여 충돌을 해결합니다. 포트 변경에 대한 자세한 내용은 JBoss 설명서를 참조하십시오.	4447	
관리 포트	JBoss HTTP 관리 포트입니다. HTTPS 관리 포트는 지원되지 않습니다. 응용 프로그램 서버에 포트 오프셋이 구성된 경우 기본값에 오프셋 값을 추가합니다. 예를 들어 포트 오프셋이 100이면 관리 포트 값은 9990 + 100인 10090입니다.	9990	

클러스터된 환경에서 기타 JBoss 인스턴스에 대한 세부 정보를 다음 테이블에 추가합니다.

속성	서버 인스턴스 2 값	서버 인스턴스 3 값	서버 인스턴스 4 값
JBoss 설치 디렉터리			
구성 이름			
원격 포트			
관리 원시 포트			

데이터베이스: Oracle 데이터베이스

설치에 필요한 Oracle 데이터베이스 세부 정보를 수집하려면 다음 테이블을 사용합니다.

속성	설명	기본값	설치 값
서버	Oracle 데이터베이스가 상주하는 서버의 호스트 이름입니다.	localhost	
포트	Oracle 데이터베이스 서버에서 실행되는 Oracle 수신기의 TCP 포트 번호입니다. Hub 서버는 이 포트를 사용하여 Oracle 데이터베이스와 통신합니다.	1521	
연결 유형	Oracle 데이터베이스를 설정할 때 정의한 연결의 유형입니다.	서비스	
서비스 이름	정의한 Oracle 서비스의 이름입니다.	orcl	
Oracle SID	tnsnames.ora 파일에서 정의한 Oracle 시스템 ID의 이름입니다. 참고: TNS 이름은 Hub 서버 및 MDM 처리 서버 시스템 모두에 대해 동일해야 합니다.	orcl	
마스터 데이터베이스 스키마 이름	MDM 마스터 데이터베이스 스키마의 이름입니다.	cmx_system	
마스터 데이터베이스 스키마 암호	MDM 마스터 데이터베이스 사용자의 암호입니다.	-	

보안 - 선택 사항

해시 알고리즘 또는 인증서 공급자를 구성하거나 이 두 가지를 둘 다 구성하도록 선택할 수 있습니다.

해시 키 및 해시 알고리즘

해시 알고리즘을 구성하려면 siperian-server-hash.jar 파일에 Java 추상 클래스 `com.siperian.sam.security.hashing.algorithms.HashAlgorithm`을 구현해야 합니다. 이 JAR 파일은 다음 디렉터리에 위치합니다. <MDM 설치 디렉터리>/hub/server/lib/hashing

다음 테이블을 사용하여 siperian-server-hash.jar 파일의 Java 추상 클래스 구현에 사용된 해시 알고리즘 세부 정보를 수집합니다.

속성	설명	기본값	설치값
해시 키	선택 사항입니다. 구분자가 포함되지 않은 최대 32자의 16진수 시퀀스입니다. 키 크기는 최대 128비트일 수 있습니다. 키를 안전하게 저장하십시오. 주의: 사용자 지정 해시 키가 노출되거나 손실되는 경우 모든 사용자 암호를 재설정해야 합니다.	-	
해시 알고리즘	구성된 알고리즘의 목록입니다. 사용자 지정 알고리즘을 구성하려면 기타 를 선택합니다.	SHA-3	
이름	사용자 지정 해시 알고리즘의 이름입니다.	-	
구현 파일	사용자 지정 해시 알고리즘 보관의 위치입니다. 참고: 압축된 파일에는 필요한 모든 JAR 파일과 지원 파일이 포함되어 있어야 합니다.	-	
정규형 클래스 이름	해시 알고리즘 구현의 정규형 클래스 이름입니다. 예: \$HASHING_CLASS_NAME\$	-	

인증서 공급자

사용자 지정 인증서 공급자를 구현하려면 siperian-server-pkiutil.jar 파일에서 PKIUtil.java 인터페이스를 구현해야 합니다. 이 JAR 파일은 다음 디렉터리에 위치합니다. <MDM 설치 디렉터리>/hub/server/lib/pkiutils

다음 테이블을 사용하여 siperian-server-pkiutil.jar 파일의 Java 추상 클래스 구현에 사용된 사용자 지정 인증서 공급자의 세부 정보를 수집합니다.

속성	설명	기본값	설치값
인증서 공급자	기본적으로 MDM은 트러스트된 응용 프로그램을 인증합니다. 사용자 지정 공급자를 구성하려면 기타 를 선택합니다.	기본값	
구현 파일	사용자 지정 인증서 공급자의 이름입니다.	-	
클래스 이름	인증서 공급자 구현의 클래스 이름입니다.	-	

Hub 콘솔

Hub 콘솔에서 Hub 서버 시스템에 연결하려면 서버 세부 정보가 필요합니다.

Hub 콘솔에 필요한 서버 세부 정보를 수집하려면 다음 테이블을 사용합니다.

속성	설명	기본값	설치값
공개적으로 액세스할 수 있는 호스트 이름	응용 프로그램 서버가 바인딩되는 서버의 IP 주소 또는 공개적으로 액세스할 수 있는 호스트 이름(FQDN)입니다.	-	
HTTP 포트	Hub 콘솔에서 사용해야 하는 서버의 HTTP 포트입니다.	-	

Informatica ActiveVOS - 선택 사항

Informatica ActiveVOS, ActiveVOS Central 또는 ActiveVOS Server에 대한 자세한 내용은 [Informatica ActiveVOS](#) 설명서를 참조하십시오.

ActiveVOS 설치에 필요한 세부 정보를 수집하려면 다음 테이블을 사용합니다.

속성	설명	기본값	설치값
ActiveVOS 서버 설치 디렉터리	ActiveVOS 서버를 설치할 빈 디렉터리입니다.	<MDM Hub 설치 디렉터리>/avos/server	
연결 유형	Oracle 데이터베이스를 설정할 때 정의한 연결의 유형입니다.	서비스 이름	
서비스 이름	정의한 Oracle 서비스의 이름입니다.	orcl	
Oracle SID	tnsnames.ora 파일에서 정의한 Oracle 시스템 ID의 이름입니다.	orcl	
서버	ActiveVOS 스키마가 포함된 Oracle 데이터베이스를 호스팅하는 시스템의 호스트 이름 또는 IP 주소입니다.	localhost	
포트	서버에서 실행되는 Oracle 수신기의 TCP 포트 번호입니다. Hub 서버는 이 포트를 사용하여 Oracle 데이터베이스 서버와 통신합니다.	1521	
ActiveVOS 스키마 이름	ActiveVOS 스키마 사용자의 이름입니다.	avos	
ActiveVOS 스키마 암호	ActiveVOS 스키마 사용자의 암호입니다.	-	
웹 서비스 URL	ActiveVOS 서버가 실행되는 호스트와 포트입니다. http 또는 https 프로토콜을 사용하도록 선택합니다. Hub 서버는 ActiveVOS Central(Process Central이라고도 함)에서 ActiveVOS 서버를 호출하기 위해 사용하는 것과 동일한 URL을 사용합니다. Informatica ActiveVOS 설명서에서는 이 URL을 Process Central AeTaskService URL이라고 합니다. 이 URL은 다음과 같은 형식을 사용합니다. http://[hostname.domainname]:[port]/active-bpel/services/AeB4PTaskClient-taskOperations.	http://localhost:8080	
ActiveVOS 서버 설치 프로그램 파일	Multidomain MDM 배포 패키지에는 ActiveVOS 서버에 대한 설치 프로그램이 포함됩니다.	ActiveVOS_Server_ <운영 체제>_<버전>	
프로세스 콘솔 사용자 이름	ActiveVOS 프로세스 콘솔에 대해 권한이 있는 사용자입니다.	-	
프로세스 콘솔 암호	ActiveVOS 프로세스 콘솔 사용자의 암호입니다.	-	

제품 사용 툴킷

이 제품 사용 툴킷은 MDM 환경에 대한 정보를 Informatica로 보냅니다. 이 정보는 Informatica 글로벌 고객 지원에서 환경에 대한 권장 사항을 제공하고 문제를 해결하는 데 사용됩니다. 툴킷을 통해 Informatica로 정보를 보내지 않으려는 경우 MDM 설치 후 툴킷을 비활성화할 수 있습니다.

제품 사용 툴킷 설치에 필요한 세부 정보를 수집하려면 다음 테이블을 사용합니다.

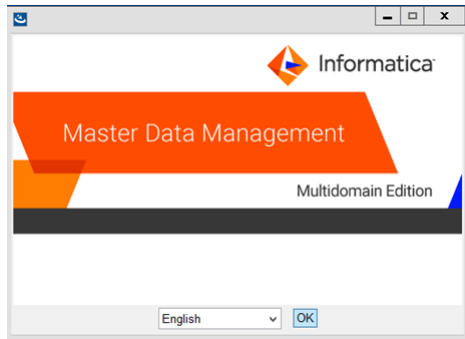
속성	설명	기본값	설치 값
산업	조직의 비즈니스에 가장 부합하는 산업 유형입니다.	-	
환경	설치 환경의 유형입니다. 명령줄에서 설치하는 경우 다음 중 하나를 입력합니다. - 1. 제품 환경 - 2. 테스트 또는 QA 환경 - 3. 개발 환경	-	
네트워크에서 프록시 서버를 사용합니까?	예인 경우 프록시 서버에 대한 세부 정보를 제공합니다.	아니요	
호스트	프록시 서버의 이름 또는 IP 주소입니다.	-	
포트	프록시 서버에 사용되는 포트 번호입니다.	-	
도메인 이름	프록시 서버가 도메인의 일부인 경우 도메인의 이름입니다.	-	
사용자 이름	보안 프록시 서버를 사용하는 경우 MDM에 액세스할 수 있는 사용자의 이름입니다.	-	
암호	사용자의 암호입니다.	-	

설치 마법사에서 Hub 서버 설치

그래픽 모드에서 Hub 서버를 설치하려는 경우 설치 마법사를 사용합니다. 설치 마법사는 설치 과정을 안내합니다.

동일한 사용자 이름을 사용하여 Hub 서버와 처리 서버를 설치해야 합니다.

- 응용 프로그램 서버를 시작합니다.
- 다음 디렉터리로 이동합니다.
<MDM Hub 배포 디렉터리>/<운영 체제 이름>/mrmserver
- 사용 중인 운영 체제에 해당하는 태스크를 수행하여 설치 프로그램을 시작합니다.
 - UNIX.** 명령줄에서 ./hub_install.bin을 실행합니다.
 - Windows.** 파일 탐색기에서 hub_install.exe를 두 번 클릭합니다.
 Hub 서버 설치 마법사가 시작됩니다.



팁: 설치 마법사가 시작되지 않으면 지원되는 **Java** 버전이 설치되어 있고 클래스 경로 또는 환경 변수에 포함되어 있는지 확인합니다.

4. 언어를 선택하고 **확인**을 클릭합니다.

소개 창이 표시됩니다.

5. 온라인 지침을 따릅니다. 메시지가 표시되면 수집한 설치 값을 입력합니다.
6. 설치가 끝나면 **구성 요약** 창에서 선택한 옵션을 검토합니다.
7. 내용을 변경하려면 **이전**을 클릭하여 해당 창으로 돌아갑니다. 완료했으면 **다음**을 클릭하여 마지막 창으로 이동합니다.
8. **설치**를 클릭합니다.
9. **다음 단계:** 다음 단계는 설치 프로그램에서 **Hub** 서버를 배포하기로 선택했는지 여부에 따라 다릅니다.
 - 설치 프로그램에서 **Hub** 서버를 배포하기로 선택한 경우에는 설치 후 태스크의 일부로 **Hub** 서버를 배포할 필요가 없습니다.
 - **Hub** 서버를 나중에 배포하기로 선택했다면 설치 후 태스크의 일부로 **Hub** 서버를 배포해야 합니다.

명령줄에서 Hub 서버 설치(UNIX만 해당)

UNIX의 경우 명령줄에서 **Hub** 서버를 설치할 수 있습니다. 스크립트를 실행하여 명령줄 설치를 시작합니다.

1. 응용 프로그램 서버를 시작합니다.
2. 명령줄에서 다음 디렉터리로 이동합니다.
`<MDM Hub 배포 디렉터리>/<운영 체제 이름>/mrmsrver`
3. 다음 명령을 실행합니다.
`./hub_install.bin -i console`
Hub 서버 설치 프롬프트가 나타납니다.
4. 수집한 설치 값을 입력합니다.
 괄호에 표시된 기본값을 사용하려면 **Enter** 키를 누릅니다.
5. **다음 단계:** 설치 완료 후 다음 단계는 **Hub** 서버를 배포하기로 선택했는지 여부에 따라 다릅니다.
 - 설치 프로그램에서 **Hub** 서버를 배포하기로 선택한 경우에는 설치 후 태스크의 일부로 **Hub** 서버를 배포할 필요가 없습니다.
 - **Hub** 서버를 나중에 배포하기로 선택했다면 설치 후 태스크의 일부로 **Hub** 서버를 배포해야 합니다.

자동으로 Hub 서버 설치

자동 모드에서 Hub 서버를 설치할 수 있습니다. 자동 설치를 시작하기 전에 자동 설치 속성 파일을 구성했는지 확인합니다.

1. 응용 프로그램 서버를 시작합니다.
2. 자동 설치 속성 파일을 대상 환경으로 복사합니다.
3. 대상 환경에서 사용 중인 운영 체제에 해당하는 명령을 실행합니다.

- **UNIX.** `./hub_install.bin -f <편집된 설치 프로그램 속성 파일의 절대 경로>`

- **Windows.** `hub_install.exe -f <편집된 설치 프로그램 속성 파일의 절대 경로>`

자동 설치 프로그램이 백그라운드에서 실행됩니다. 이 프로세스에는 다소 시간이 걸릴 수 있습니다.

4. 설치 프로그램을 통해 Hub 서버를 배포하도록 선택한 경우에는 `postinstallSetup.log`를 확인하여 설치가 제대로 되었는지 확인합니다.

log 파일은 다음 디렉터리에 있습니다.

<MDM Hub 설치 디렉터리>/hub/server/logs

5. **다음 단계:** 설치 완료 후 다음 단계는 Hub 서버를 배포하기로 선택했는지 여부에 따라 다릅니다.

- 설치 프로그램에서 Hub 서버를 배포하기로 선택한 경우에는 설치 후 태스크의 일부로 Hub 서버를 배포할 필요가 없습니다.
- Hub 서버를 나중에 배포하기로 선택했다면 설치 후 태스크의 일부로 Hub 서버를 배포해야 합니다.

클러스터의 노드에 Hub 서버 설치

응용 프로그램 서버 클러스터 환경에서, Hub 서버 응용 프로그램을 배포해야 하는 클러스터의 모든 노드에 Hub 서버를 설치합니다. 클러스터의 노드 하나에서 설치를 완료한 이후에 클러스터의 다른 노드에서 설치를 진행해야 합니다.

예를 들어 JBoss 클러스터에 `host1` 및 `host2`에서 실행되고 RMI 포트 4447 및 4448을 사용하는 노드 두 개가 있다고 가정해 보겠습니다. Hub 서버를 `node1`과 `node2`에 설치해야 합니다. 노드 하나(예: `node2`)에서 Hub 서버 설치를 완료한 이후 다른 노드(`node1`)에서 설치를 시작해야 합니다.

Hub 서버 설치의 디렉터리 구조가 모든 노드에서 동일한지 확인합니다.

1. Hub 서버를 설치할 JBoss 클러스터 노드를 시작합니다.
2. 다음 디렉터리로 이동합니다.

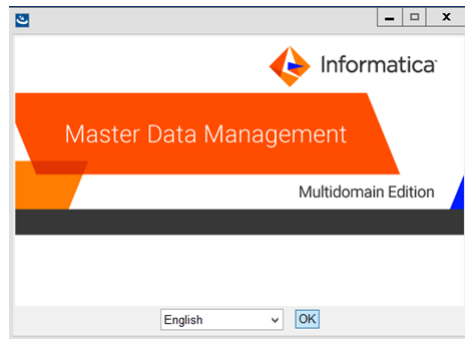
<MDM Hub 배포 디렉터리>/<운영 체제 이름>/mrmsrver

3. 클러스터 노드에서 Hub 서버 설치 프로그램을 시작하려면 사용 중인 운영 체제에 맞는 명령을 실행합니다.

UNIX. `./hub_install.bin`

Windows. `hub_install.exe`

Hub 서버 설치 마법사가 시작됩니다.



4. 온라인 지침을 따릅니다. 메시지가 표시되면 수집한 설치 값을 입력합니다.
5. **다음 단계:** 설치가 완료되면 설치가 있는 모든 노드에 Hub 서버를 수동으로 배포합니다.
기본 노드에 대해 자동 배포를 수행한 경우 기본 노드에서 수동으로 배포할 필요가 없습니다.
JBoss 클러스터의 아무 노드나 사용하여 Hub 서버에 액세스할 수 있습니다.

제 6 장

Hub 서버 설치 후 태스크

이 장에 포함된 항목:

- [설치 로그 파일 복사, 48](#)
- [버전 및 빌드 번호 확인, 48](#)
- [Elasticsearch 설치 및 구성, 49](#)
- [Hub 콘솔 클라이언트 구성\(조건부\), 49](#)
- [MDM Hub 마스터 데이터베이스 이름 구성, 49](#)
- [응용 프로그램 서버 설정 확인 및 구성\(조건부\), 50](#)
- [Hub 서버 응용 프로그램 배포\(조건부\), 51](#)
- [스크립트를 사용하여 Hub 서버 응용 프로그램 배포\(조건부\), 52](#)
- [수동으로 Hub 서버 응용 프로그램 배포\(조건부\), 53](#)
- [메타데이터 캐시 구성\(선택 사항\), 61](#)
- [Hub 콘솔 시작, 62](#)
- [연산 참조 저장소 등록, 63](#)
- [Oracle Configuration Manager 등록\(선택 사항\), 65](#)
- [ActiveVOS용 Oracle Configuration Manager 설정\(선택 사항\), 66](#)
- [추가 응용 프로그램 서버 구성\(선택 사항\), 67](#)

설치 로그 파일 복사

설치 로그 파일은 Hub 서버 설치 프로세스의 문제를 해결하는 데 유용합니다. 로그 파일을 설치 기록 디렉터리에 복사합니다. Informatica 글로벌 고객 지원 센터는 고객이 설치 문제와 관련하여 문의할 경우 로그 파일 사본을 요청할 수 있습니다.

다음 테이블에는 다른 유형의 설치 로그 파일이 설명되어 있습니다.

로그 파일 유형	설명
설치 로그	<ul style="list-style-type: none"> - 파일 이름. Informatica_MDM_Hub_Server_Install_<타임스탬프>.xml - 위치. <MDM Hub 설치 디렉터리>/hub/server/logs - 콘텐츠. 생성된 디렉터리 및 레지스트리 항목, 설치된 파일과 실행된 명령의 이름 및 설치된 각 파일의 상태입니다.
설치 선행 조건 로그	<ul style="list-style-type: none"> - 파일 이름. installPrereq.log - 위치. <MDM Hub 설치 디렉터리>/hub/server/logs - 콘텐츠. 설치 프로그램이 수행한 선행 조건 검사의 로그입니다.
디버그 로그	<ul style="list-style-type: none"> - 파일 이름. infamdm_installer_debug.txt - 위치. <MDM Hub 설치 디렉터리> /hub/server - 콘텐츠. 설치 중 수행된 선택 항목 및 설치 프로그램이 수행한 작업에 대한 자세한 정보입니다.
설치 후 설정 로그	<ul style="list-style-type: none"> - 파일 이름. postInstallSetup.log - 위치. <MDM Hub 설치 디렉터리>/hub/server/logs - 콘텐츠. 설치 후 프로세스 중 설치 프로그램이 수행한 작업 및 설치 후 프로세스 오류의 요약입니다.
Hub 서버 로그	<ul style="list-style-type: none"> - 파일 이름. cmxserver.log - 위치. <MDM Hub 설치 디렉터리>/hub/server/logs - 콘텐츠. Hub 서버 작업의 요약입니다.
JBoss 로그	<ul style="list-style-type: none"> - 파일 이름. server.log - 위치. <JBoss 설치 디렉터리>/standalone/log - 콘텐츠. JBoss 이벤트 로그 및 오류 메시지입니다.

버전 및 빌드 번호 확인

올바른 버전 및 빌드 번호의 Hub 서버가 설치되어 있는지 확인합니다.

1. 명령 프롬프트를 열고 다음 디렉터리로 이동합니다. <MDM Hub 설치 디렉터리>/hub/server/bin
2. Hub 서버 버전 및 빌드 번호를 확인하려면 다음 명령을 실행합니다.

UNIX의 경우. versionInfo.sh

Windows의 경우. versionInfo.bat

참고: AIX 시스템의 경우 <>/jre/bin 디렉터리에서 Java를 실행하도록 Java 홈versionInfo.sh 스크립트를 변경합니다.

Elasticsearch 설치 및 구성

검색을 사용하려면 MDM Hub용 Elasticsearch를 설치 및 설정하십시오.

검색을 설치 및 구성하는 방법에 대한 자세한 내용은 *Multidomain MDM 구성 가이드*의 "Elasticsearch로 검색" 장을 참조하십시오.

Hub 콘솔 클라이언트 구성(조건부)

Hub 콘솔에서 Hub 서버 시스템에 연결하려면 `build.properties` 파일에 호스트 이름과 포트 속성이 필요합니다. Hub 콘솔을 실행할 때 호스트 이름 및 포트 번호를 재정의할 수 있습니다.

다음 시나리오에서 속성 파일을 편집합니다.

- 응용 프로그램 서버에 대해 HTTPS가 활성화된 경우 HTTPS 포트를 사용해야 합니다.
- 다중 노드 또는 클러스터 환경에서, Hub 서버 설치 디렉터리를 한 노드에서 다른 노드로 복사했다면 각 노드를 편집합니다.

1. 다음 디렉터리에서 `build.properties` 파일을 엽니다.

<MDM Hub 설치 디렉터리>/hub/server/bin

2. 다음 속성을 편집합니다.

- `console.hostname`. 응용 프로그램 서버가 바인딩되는 서버의 IP 주소 또는 공개적으로 액세스할 수 있는 호스트 이름(FQDN)을 지정합니다.
- `console.webport`. Hub 콘솔에서 사용해야 하는 현재 노드의 HTTP 또는 HTTPS 포트를 지정합니다.

3. 파일을 저장합니다.

4. 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.

`build.properties` 파일을 편집한 후 `postInstallSetup` 스크립트를 실행하여 Hub 서버 응용 프로그램을 배포해야 합니다.

MDM Hub 마스터 데이터베이스 이름 구성

MDM Hub 마스터 데이터베이스의 이름이 `cmx_system`이 아닌 경우 `cmxserver.properties` 파일에서 MDM Hub 마스터 데이터베이스 이름을 구성합니다.

1. 다음 디렉터리에서 `cmxserver.properties` 파일을 엽니다.

<MDM Hub 설치 디렉터리>/hub/server/resources

2. `cmx.server.masterdatabase.schemaname` 속성을 MDM Hub 마스터 데이터베이스에 지정한 이름으로 설정합니다.

이 이름은 MDM Hub 마스터 데이터베이스를 생성할 때 지정한 이름입니다.

응용 프로그램 서버 설정 확인 및 구성(조건부)

MDM Hub 환경의 요구 사항을 기반으로 응용 프로그램 서버 설정을 확인 및 구성합니다.

다음 테이블에는 수행할 수 있는 구성 태스크가 설명되어 있습니다.

구성 태스크	설명
응용 프로그램 서버 설정 편집	설치 중 <code>postInstallSetup</code> 스크립트를 실행하고 잘못된 응용 프로그램 서버 설정 때문에 스크립트가 실패한 경우에 필요합니다.
JBoss 다중 노드 또는 클러스터 환경에 대한 Hub 서버 구성	JBoss 다중 노드 또는 클러스터 환경에 Hub 서버를 설치한 경우에 필요합니다.

응용 프로그램 서버 설정 편집

설치 중 `postInstallSetup` 스크립트를 실행하고 잘못된 응용 프로그램 서버 설정 때문에 스크립트가 실패한 경우 `build.properties` 파일을 편집합니다. 또한 응용 프로그램 서버 설정을 변경하려는 경우에도 파일을 편집합니다.

1. 다음 디렉터리에서 `build.properties` 파일을 엽니다.

<MDM Hub 설치 디렉터리>/hub/server/bin

2. 응용 프로그램 서버 설정을 편집하고 파일을 저장합니다.

`build.properties` 파일을 편집한 후 `postInstallSetup` 스크립트를 실행하여 Hub 서버 응용 프로그램을 배포해야 합니다.

JBoss 다중 노드 또는 클러스터 환경에 대한 Hub 서버 구성

JBoss 다중 노드 또는 클러스터 환경에서 Hub 서버를 설치한 경우 JBoss 클러스터에 대한 Hub 서버를 구성합니다. JBoss 환경에 대한 Hub 서버를 구성하려면 `cmxserver.properties` 파일에서 JBoss 환경 속성을 구성합니다.

예를 들어 JBoss 클러스터 또는 다중 노드 환경에 `host1`과 `host2`에서 실행되는 2개의 노드가 있는 경우 두 노드에 JBoss 클러스터 속성을 구성합니다.

1. 환경의 모든 노드에서 다음 디렉터리의 `cmxserver.properties` 파일을 엽니다.

<MDM Hub 설치 디렉터리>/hub/server/resources

2. 다음과 같은 JBoss 환경 속성을 구성합니다.

속성	설명
jboss.cluster	EJB 서버가 다중 노드 서버인지, Hub 서버에 대해 클러스터된 서버인지 여부를 지정합니다. jboss.cluster 속성을 추가하고 true로 설정합니다.
cmx.appserver.hostname	JBoss 다중 노드 또는 클러스터 환경의 시스템 이름을 지정합니다. 시스템 이름은 다음 형식이어야 합니다. <호스트 이름>.<도메인>.com 예를 들어 클러스터 또는 다중 노드 환경에 host1과 host2에서 실행되고 RMI 포트 4447과 4448을 사용하는 2개의 노드가 있는 경우 다음을 설정합니다. cmx.appserver.hostname=host1.<도메인>.com,host2.<도메인>.com
cmx.appserver.rmi.port	원격 포트 번호를 지정합니다. 예를 들어 클러스터 또는 다중 노드 환경에 host1과 host2에서 실행되고 RMI 포트 4447과 4448을 사용하는 2개의 노드가 있는 경우 다음을 설정합니다. cmx.appserver.rmi.port=4447,4448 참고: 포트 번호 간에 공백 없이 심표를 사용하여 포트 번호를 구분합니다. 포트 번호의 순서가 호스트 이름의 순서와 일치해야 합니다.

Hub 서버 응용 프로그램 배포(조건부)

Hub 서버를 설치한 것과 동일한 시스템에 Hub 서버 응용 프로그램을 배포해야 합니다.

Hub 서버 응용 프로그램은 Hub 서버 응용 프로그램이 속해 있는 Hub 서버 설치를 찾을 수 있어야 합니다. 그러므로 다른 시스템에 배포하기 위해 EAR 파일을 전송하지 마십시오. 예를 들어 테스트 시스템에 Hub 서버를 설치한 다음 프로덕션 시스템에 응용 프로그램을 배포합니다. 프로덕션 시스템에 배포하는 응용 프로그램은 로깅 구성과 같은 정보를 위한 테스트 시스템에서 설치를 찾을 수 없습니다.

다음과 같은 시나리오에서 Hub 서버 응용 프로그램을 배포해야 합니다.

- 설치가 응용 프로그램 서버 다중 노드 환경 또는 클러스터 환경에서 수행됩니다.
- 설치가 완료되었지만 설치 중 실행한 postInstallSetup 스크립트가 실패합니다.
- 설치 중 postInstallSetup 스크립트를 건너뛰었습니다.

설치가 응용 프로그램 서버 다중 노드 또는 클러스터 환경에 있는 경우 다음 단계를 수행하여 배포를 완료합니다.

1. 하나의 노드에 Hub 서버 응용 프로그램을 배포합니다.
2. 응용 프로그램이 배포된 노드의 certificates 디렉터리에서 다른 모든 노드의 certificates 디렉터리로 모든 파일을 복사합니다. certificates 디렉터리는 다음 위치에 있습니다.
<MDM Hub 설치 디렉터리>/hub/server/resources
3. 인증서가 복사된 모든 노드에서 Hub 서버 EAR 파일을 다시 패키징합니다. 다시 패키징 프로세스는 hubConsole.jar 파일을 올바른 인증서로 업데이트합니다.
 - a. 다음 디렉터리로 이동합니다.
<MDM Hub 설치 디렉터리>/hub/server/bin
 - b. 다음 명령을 실행합니다.

UNIX의 경우.

```
./sip_ant.sh repack
```

Windows의 경우.

```
sip_ant.bat repack
```

4. 기타 노트에 Hub 서버 응용 프로그램을 배포합니다.

Hub 서버 응용 프로그램을 배포하려면 다음 테이블에 설명된 다음 프로시저 중 하나를 사용합니다.

프로시저	설명
배포에 스크립트 사용	postInstallSetup 스크립트를 실행하여 Hub 서버 응용 프로그램을 배포합니다. 또한 스크립트는 데이터 소스를 생성하고 JMS 메시지 대기열을 구성합니다.
수동 배포	Hub 서버 응용 프로그램을 수동으로 배포합니다. 또한 수동으로 데이터 소스를 생성하고 JMS 메시지 대기열을 구성해야 합니다.

스크립트를 사용하여 Hub 서버 응용 프로그램 배포(조건부)

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

1. 명령 프롬프트를 열고 다음 디렉터리로 변경합니다.

```
<MDM Hub 설치 디렉터리>/hub/server
```

2. postInstallSetup 스크립트를 실행합니다.

참고: MDM Hub 설치 프로그램과 함께 번들로 제공되는 ActiveVOS 버전을 설치하지 않은 경우 ActiveVOS 사용자 이름과 암호를 명령에 포함시키지 마십시오.

UNIX의 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>
```

참고: 암호에 느낌표(!)를 포함시킨 경우 느낌표 앞에 백슬래시를 포함시켜야 합니다. 예를 들어, 암호가 !!cmx!!, 다음 암호를 입력하십시오. \!cmx\!!

Windows의 경우.

```
postInstallSetup.bat  
-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>
```

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

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

수동으로 Hub 서버 응용 프로그램 배포(조건부)

설치 중 `postInstallSetup` 스크립트를 건너뛰었거나 `postInstallSetup` 스크립트가 실패한 경우 Hub 서버 응용 프로그램을 수동으로 배포할 수 있습니다. Hub 서버 설치 디렉터리에서 Hub 서버 응용 프로그램을 배포해야 합니다.

Hub 서버 응용 프로그램을 배포하려면 다음과 같은 작업을 수행합니다.

1. 데이터 소스 생성
2. JMS 메시지 대기열 구성
3. Hub 서버 응용 프로그램 배포
4. Hub 서버에서 JMS 메시지 대기열 구성
5. IDD(Informatica Data Director)의 서버 리소스 구성

1단계. 데이터 소스 생성

Hub 서버 응용 프로그램을 수동으로 배포하기 전에 데이터 소스를 생성합니다. 또한 여러 처리 서버를 구성하거나 설치 문제를 해결하려는 경우 데이터 소스를 생성합니다.

1. <JBoss 설치 디렉터리>/modules/ 아래에 다음과 같은 디렉터리 구조를 생성합니다.
/com/informatica/mdm/jdbc/main

2. main 디렉터리에서 다음 콘텐츠를 가진 `module.xml` 파일을 생성합니다.

```
<?xml version="1.0" encoding="UTF-8"?>
<module xmlns="urn:jboss:module:1.0" name="com.informatica.mdm.jdbc">
  <resources>
    <resource-root path="JDBC 드라이버 파일 이름"/>
  </resources>
  <dependencies>
    <module name="javax.api"/>
    <module name="javax.transaction.api"/>
  </dependencies>
</module>
```

3. 지원되는 JDBC 드라이버 버전을 다운로드하여 main 디렉터리에 복사합니다.
4. JBoss 관리 콘솔을 열고 **프로필**을 클릭합니다.
데이터 소스 페이지가 표시됩니다.
5. 커넥터 > 데이터 소스를 클릭합니다.
6. **XA 데이터 소스** 탭을 클릭합니다.
JDBC XA 데이터 소스 페이지가 나타납니다.
7. **추가**를 클릭합니다.
XA 데이터 소스 생성 대화 상자가 나타납니다.
8. **이름** 및 **JNDI 이름** 필드에 값을 입력하고 **다음**을 클릭합니다.

다음 테이블에는 지정할 값의 구문이 나열되어 있습니다.

필드 이름	값의 구문
이름	jdbc/siperian-<Oracle host name>-orcl-<Operational Reference Store name>-ds
JNDI 이름	java:jboss/datasources/jdbc/siperian-<Oracle host name>-orcl-<Operational Reference Store name>-ds

XA 데이터 소스 페이지가 표시됩니다.

- XA 데이터 소스 클래스 필드에 `com.informatica.mdm.jdbc`를 입력하고 **다음**을 클릭합니다.

XA 속성 페이지가 나타납니다.

- 키와 값 쌍을 입력하고 **다음**을 클릭합니다.
다음 테이블에는 키 값 쌍이 나열되어 있습니다.

키	값
ConnectionProperties	oracle.jdbc.J2EE13Compliant=true
URL	Oracle 연결 유형에 따라 다음 값 중 하나를 사용합니다. SERVICE의 경우. jdbc:oracle:thin:@//<Oracle 호스트 이름>:1521:<서비스 이름> SID의 경우. jdbc:oracle:thin:@<Oracle 호스트 이름>:1521:orcl

연결 설정 창이 나타납니다.

- 연결 설정에 대한 값을 입력하고 **완료**를 클릭합니다.
다음 테이블에는 연결 필드가 설명되어 있습니다.

필드 이름	설명
사용자 이름	연산 참조 저장소의 이름
암호	연산 참조 저장소에 액세스하는 데 필요한 암호입니다.

JDBC XA 데이터 소스 페이지가 나타납니다.

- 선택 섹션의 **풀** 탭에서 속성을 편집합니다.

속성	값
최소 풀 크기	5
최대 풀 크기	100
정확한 최솟값	false

속성	값
사전 채우기 활성화됨	false
플러시 전략	FailingConnectionOnly
유효 시간 초과	0
추적 설명	false

13. 선택 섹션의 **풀** 탭에서 속성을 편집합니다.

속성	값
유효한 연결 검사기	적용할 수 없음
유효한 SQL 검사	적용할 수 없음
일치 시 유효성 검사	false
배경 유효성 검사	false
유효성 검사 밀리초	0
부실한 연결 검사기	적용할 수 없음
예외 분류기	org.jboss.jca.adapters.jdbc.extensions.oracle.OracleExceptionSorter

14. 추가한 데이터 소스를 선택하고 **활성화**를 클릭합니다.
추가한 데이터 소스가 활성화됩니다.

2단계. JMS 메시지 대기열 구성

Hub 서버 응용 프로그램을 수동으로 배포하기 전에 JMS 메시지 대기열을 구성합니다. 또한 문제를 해결하려면 JMS 메시지 대기열을 수동으로 구성해야 할 수 있습니다. 예를 들어 자동화된 대기열 생성 프로세스가 실패하거나 설치 후 대기열이 실수로 삭제되는 경우 메시지 대기열을 수동으로 구성해야 합니다.

SIF(서비스 통합 프레임워크)는 JMS 메시지 대기열의 MDB(메시지 구동 빈)를 사용하여 수신 비동기 SIF 요청을 처리합니다. MDM Hub 구현에 대해 사용하는 응용 프로그램 서버의 메시지 대기열 및 연결 팩터리를 구성합니다. JMS 메시지 대기열을 구성하면 연결 팩터리도 생성하게 됩니다.

JMS 메시지 대기열을 구성하려면 다음 태스크를 수행합니다.

1. 연결 팩터리를 생성합니다.
2. SIF 요청에 대한 JMS 메시지 대기열을 생성합니다.
3. JMS 메시지 대기열을 확인합니다. 그리고 검색용 JMS 메시지 대기열을 생성합니다.

1단계. 연결 팩터리 생성

연결 팩터리를 수동으로 생성할 수 있습니다.

1. JBoss 관리 콘솔을 엽니다.
2. **프로필 > 메시징 > 대상**을 클릭합니다.

JMS 메시징 공급자 페이지가 나타납니다.

3. 기본값 JMS 메시징 공급자 설정을 보려면 **보기**를 클릭합니다.
메시징 대상 페이지가 나타납니다.
4. **연결 팩터리** 링크를 클릭합니다.
구성된 연결 팩터리가 나타납니다.
5. 연결 팩터리를 추가하려면 **추가**를 클릭합니다.
연결 팩터리 생성 대화 상자가 나타납니다.
6. 연결 팩터리 세부 정보를 입력합니다.
다음 테이블에는 연결 팩터리 세부 정보 필드가 설명되어 있습니다.

필드 이름	설명
이름	연결 팩터리 이름입니다. java:/siperian.mrm.jms.xaconnectionfactory를 지정합니다.
JNDI 이름	JNDI 이름. java:/siperian.mrm.jms.xaconnectionfactory를 지정합니다.
커넥터	서버에 연결하는 데 사용하는 커넥터입니다. 다음 커넥터를 사용할 수 있습니다. - in-vm. 로컬 서버에 연결하려면 in-vm 커넥터를 사용합니다. - netty. 원격 서버에 연결하려면 netty 커넥터를 사용합니다.

7. **저장**을 클릭합니다.
연결 팩터리가 생성됩니다.

2단계. JMS 메시지 대기열 생성 SIF 요청의 경우

JMS 메시지 대기열을 생성하려면 JBoss 관리 콘솔을 사용합니다.

1. JBoss 관리 콘솔을 엽니다.
2. **프로필 > 메시징 > 대상**을 클릭합니다.
JMS 메시징 공급자 페이지가 나타납니다.
3. 기본 JMS 메시징 공급자의 설정을 보려면 **보기**를 클릭합니다.
JMS 대기열 대상 페이지가 나타납니다.
4. **추가**를 클릭합니다.
JMS 대기열 생성 대화 상자가 나타납니다.
5. 다음 옵션을 지정합니다.

옵션	값
이름	java:/queue/siperian.sif.jms.queue
JNDI 이름	java:/queue/siperian.sif.jms.queue
지속형	지속형 확인란을 선택 취소합니다.

6. **저장**을 클릭합니다.
대기열이 생성됩니다.

3단계. 검색용 JMS 메시지 대기열 생성

검색 기능을 사용하여 Data Director 내에서 전체 텍스트 검색을 활성화하려면 JMS 대기열이 필요합니다. JBoss 관리 콘솔에서 검색용 대기열을 추가하거나 standalone-full.xml 파일을 편집하여 대기열을 추가할 수 있습니다. 파일을 편집하는 경우 연결 풀 및 SIF 요청의 JMS 대기열이 파일에 나타나는지 확인할 수 있습니다.

1. 다음 디렉터리로 이동합니다.

<JBoss 설치 디렉터리>/standalone/configuration

2. 텍스트 편집기에서 standalone-full.xml 파일을 엽니다.

3. queue/siperian.sif.jms.queue를 검색합니다.

다음 코드 샘플은 연결 풀 siperian.mrm.jms.xaconnectionfactory 및 JMS 대기열 queue/siperian.sif.jms.queue의 구성을 보여 줍니다.

```
<jms-connection-factories>
...
<pooled-connection-factory name="siperian.mrm.jms.xaconnectionfactory">
  <transaction mode="xa"/>
  <connectors>
    <connector-ref connector-name="in-vm"/>
  </connectors>
  <entries>
    <entry name="java:/siperian.mrm.jms.xaconnectionfactory"/>
  </entries>
</pooled-connection-factory>
</jms-connection-factories>

<jms-destinations>
...
<jms-queue name="queue/siperian.sif.jms.queue">
  <entry name="queue/siperian.sif.jms.queue"/>
  <entry name="jboss/exported/queue/siperian.sif.jms.queue"/>
</jms-queue>
...
</jms-destinations>
```

4. </jms-queue> 태그 뒤에 다음 코드를 삽입해서 검색용 JMS 대기열을 생성합니다.

```
<jms-queue name="queue/informatica.mdm.sss.jms.queue">
  <entry name="queue/informatica.mdm.sss.jms.queue"/>
  <entry name="jboss/exported/queue/informatica.mdm.sss.jms.queue"/>
</jms-queue>
```

5. 파일을 저장합니다.

3단계. Hub 서버 응용 프로그램 배포

수동으로 Hub 서버 응용 프로그램을 배포할 수 있습니다. Hub 서버 설치 디렉터리에서 Hub 서버 응용 프로그램을 배포해야 합니다.

1. JBoss 응용 프로그램 서버를 중지합니다.

2. 기존 배포가 있는 경우 JBoss 배포 디렉터리에서 다음 배포 파일을 제거합니다.

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.
ave_jboss.ear	필수 사항입니다. ActiveVOS 서버 응용 프로그램입니다.
activevos-central.war	필수 사항입니다. ActiveVOS Central 응용 프로그램입니다.

3. 소스에서 대상 디렉터리로 이전 단계에서 나열된 배포 파일을 복사합니다.

소스: Hub 서버 설치 디렉터리

대상: <JBoss 설치 디렉터리>\standalone\deployments

4. 응용 프로그램 서버를 시작합니다.

응용 프로그램 배포에 대한 자세한 내용은 JBoss 설명서를 참조하십시오.

4단계. Hub 서버에 JMS 메시지 대기열 구성

Hub 서버 응용 프로그램을 수동으로 배포한 후 Hub 서버에 JMS 메시지 대기열을 구성합니다.

Hub 서버에 JMS 메시지 대기열을 구성하려면 다음 태스크를 수행합니다.

1. Hub 콘솔을 시작합니다.
2. 메시지 대기열 서버를 추가합니다.
3. 메시지 대기열을 추가합니다.

1단계. Hub 콘솔 시작

MDM Hub에 액세스하려면 Hub 콘솔을 시작합니다.

1. 브라우저 창을 열고 다음 URL을 입력합니다.

`http://<MDM Hub 호스트>:<포트 번호>/cmx/`

Hub 콘솔 실행 페이지가 표시됩니다.

2. 사용자 이름 및 암호를 입력한 후 **다운로드**를 클릭합니다.

Hub 콘솔을 실행하는 데 필요한 MDM Hub 응용 프로그램 JAR 파일이 다운로드됩니다.

참고: MDM Hub 응용 프로그램 JAR 파일을 다운로드할 수 없는 경우에는 MDM 관리자에게 문의하십시오.

관리자는 다음 디렉터리에서 JAR 파일을 배포할 수 있습니다. <MDM Hub 설치 디렉터리>/hub/server/resources/hub

3. 응용 프로그램 JAR 파일을 실행합니다.

참고: 클라이언트 시스템에 SSL 인증서가 없는 경우 HTTPS 연결을 통해 Hub 콘솔에 액세스하려면 SSL 인증서를 설치해야 합니다. 다음 절차를 사용하여 설치할 수 있습니다.

- 다음 명령을 실행하여 로컬 클라이언트 시스템의 Java 키 저장소로 인증서를 가져옵니다.

```
keytool -import -trustcacerts -alias <certificate alias name> -file <certificate alias file> -keystore <local java cacerts keystore location>
```

- 다음 명령을 실행하여 인증서가 포함된 트러스트 저장소 파일의 위치 및 암호를 전달합니다.

```
java -Djavax.net.ssl.trustStore=<truststore file location> -Djavax.net.ssl.trustStorePassword=<truststore_password> -jar hubConsole.jar
```

기본 cacert 파일 대신 모든 사용자 지정 트러스트 인증서가 포함된 개별 트러스트 저장소를 사용합니다. 응용 프로그램 서버를 유지 관리하는 팀으로부터 인증서를 확보합니다. 서버는 자체 서명 인증서 또는 보안 인증서를 사용할 수 있습니다. .jar 파일은 서버에서 버전이 변경된 경우에만 다운로드합니다. .jar 파일을 다운로드할 때마다 동일한 명령을 사용하여 실행합니다.

4. 응용 프로그램에 대한 최대 메모리 할당 풀을 지정하려면 다음 명령을 실행합니다.

```
java -Xmx<n>G -jar hubConsole.jar
```

여기서 <n>은 GB 단위의 최대 메모리 할당입니다.

Informatica MDM Hub 로그인 대화 상자가 표시됩니다.

5. 사용자 이름과 암호를 입력합니다.

6. 특정 Hub 서버 노드에 연결하거나 로드 균형 조정기 또는 역방향 프록시 서버를 사용하려는 경우 연결 속성 필드에 미리 구성되어 있는 연결 매개 변수를 재정의합니다.

다음 형식으로 매개 변수를 입력합니다.

```
<host name>:<port name>
```

여기서 호스트 이름과 포트 이름은 Hub 서버의 호스트 이름과 포트 이름이거나, 사용하는 로드 균형 조정기 또는 역방향 프록시 서버의 호스트 이름과 포트 이름입니다.

7. **확인**을 클릭합니다.

데이터베이스 변경 대화 상자가 표시됩니다.

8. 대상 데이터베이스를 선택합니다.

대상 데이터베이스는 MDM Hub 마스터 데이터베이스입니다.

9. 목록에서 언어를 선택하고 **연결**을 클릭합니다.

Hub 콘솔 사용자 인스턴스가 선택한 언어로 표시됩니다. Hub 콘솔 사용자 인터페이스에서 표시되는 언어를 변경해야 하는 경우 언어를 선택하여 Hub 콘솔을 다시 시작합니다.

2단계. 메시지 대기열 서버 추가

메시지 대기열을 추가하기 전에 MDM Hub 구현에 메시지 대기열 서버를 추가해야 합니다.

1. Hub 콘솔의 구성 작업 영역에서 **메시지 대기열**을 클릭합니다.

2. **쓰기 잠금 > 잠금 획득**을 클릭합니다.

3. 메시지 대기열 도구의 가운데 창에서 마우스 오른쪽 단추를 클릭한 다음 **메시지 대기열 서버 추가**를 클릭합니다.

메시지 대기열 서버 추가 대화 상자가 표시됩니다.

4. 메시지 대기열 서버 세부 정보를 입력합니다.

다음 테이블에는 JMS 메시지 대기열 서버를 구성하는 데 사용하는 필드가 설명되어 있습니다.

필드 이름	값
연결 팩터리 이름	연결 팩터리의 이름입니다. java:/siperian.mrm.jms.xaconnectionfactory를 지정합니다.
표시 이름	Hub 콘솔에 표시되어야 하는 메시지 대기열 서버의 이름입니다. java:/siperian.mrm.jms.xaconnectionfactory를 지정합니다.

5. **확인**을 클릭합니다.
메시지 대기열 서버가 추가됩니다.

3단계. 메시지 대기열 추가

사용자 지정 메시지 대기열을 메시지 대기열 서버에 추가할 수 있습니다.

1. Hub 콘솔의 구성 작업 영역에서 **메시지 대기열**을 클릭합니다.
2. **쓰기 잠금 > 잠금 획득**을 클릭합니다.
3. 메시지 대기열 도구의 가운데 창에서 메시지 대기열 서버를 마우스 오른쪽 단추를 클릭한 다음 **메시지 대기열 추가**를 클릭합니다.
메시지 대기열 추가 대화 상자가 표시됩니다.
4. JMS 메시지 대기열 세부 정보를 입력합니다.
다음 테이블에는 JMS 메시지 대기열 필드가 설명되어 있습니다.

필드 이름	값
대기열 이름	메시지 대기열의 이름입니다. queue/siperian.sif.test.jms.queue를 지정합니다.
표시 이름	Hub 콘솔에 표시되어야 하는 메시지 대기열의 이름입니다. queue/siperian.sif.test.jms.queue를 지정합니다.

5. **확인**을 클릭합니다.
메시지 대기열이 메시지 대기열 서버에 추가됩니다.
6. 오른쪽 창에서 **메시지 트리거와 함께 사용** 옵션을 선택합니다.
7. **테스트**를 클릭합니다.
메시지 대기열 테스트의 결과가 나타납니다.

5단계. Informatica Data Director의 서버 리소스 구성

IDD(Informatica Data Director)를 사용하려는 경우 JNDI URL 리소스를 구성합니다.

1. 다음 디렉터리에서 standalone-full.xml 파일을 엽니다.
<JBoss 설치 디렉터리>/standalone/configuration
2. 다음 코드에 표시된 것과 같이 simple name 구성을 추가합니다.

이름이 urn:jboss:domain:naming:<n.n>인 하위 시스템에서 다음 simple name 구성을 bindings 요소 안에 추가합니다.

```
<simple name="java:jboss/url/hubserver/home" value="file:///<Hub Server installation directory>"
type="java.net.URL"/>
```

메타데이터 캐시 구성(선택 사항)

메타데이터 캐시는 데이터 개체, 리포지토리, 개체 및 검색 토큰과 같은 항목을 관리합니다. MDM Hub는 메타데이터 캐싱에 Infinispan을 사용합니다. Infinispan은 Hub 서버를 통해 설치됩니다. Hub 서버에서 사용하는 캐시의 경우 Infinispan 구성 파일에 기본 특성 값이 포함됩니다.

캐시의 기본 특성 값을 사용하여 MDM Hub를 실행합니다. 성능 문제가 발생하면 특성 값을 환경에 적합한 값으로 세부 조정할 수 있습니다.

다음 테이블에는 기본 특성 값이 요약되어 있습니다.

Infinispan 요소 및 특성	기본값	설명
locking acquire-timeout	60000	Hub 서버가 잠금 획득을 시도할 수 있는 최대 시간입니다.
transaction stop-timeout	30000	이 특성은 캐시가 중지된 경우 Hub 서버의 원격 및 로컬 트랜잭션이 완료될 때까지 Infinispan이 대기하는 최대 시간을 설정합니다.
transport cluster	infinispan-cluster	기본 그룹 통신 클러스터의 이름입니다.
transport stack	UDP	구성 유형(UDP 또는 TCP)입니다. 구성은 jgroups-udp.xml 파일과 jgroups-tcp.xml 파일에 정의됩니다.
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.
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.

Infinispan 특성 편집

메타데이터 캐시 특성을 구성하려면 Hub 서버에 대한 `infinispanConfig.xml` 파일을 편집합니다. Infinispan 구성 관련 도움말은 Infinispan 설명서를 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.

Hub 콘솔 시작

MDM Hub에 액세스하려면 HTTP 또는 HTTPS 연결을 사용하여 Hub 콘솔을 시작합니다.

Hub 콘솔을 시작하기 전에 다음과 같은 정보가 있는지 확인합니다.

- URL의 호스트 이름 및 포트 번호
- 사용자 이름 및 암호
- HTTPS 연결을 통해 Hub 콘솔에 액세스하려는 경우 클라이언트 시스템의 SSL 인증서

1. 브라우저 창을 열고 다음 URL을 입력합니다.

`http://<MDM Hub 호스트>:<포트 번호>/cmx/`

Hub 콘솔 실행 페이지가 표시됩니다.

2. 사용자 이름 및 암호를 입력한 후 **다운로드**를 클릭합니다.

Hub 콘솔을 실행하는 데 필요한 MDM Hub 응용 프로그램 JAR 파일이 다운로드됩니다.

참고: MDM Hub 응용 프로그램 JAR 파일을 다운로드할 수 없는 경우에는 MDM 관리자에게 문의하십시오. 관리자는 다음 디렉터리에서 JAR 파일을 배포할 수 있습니다. <MDM Hub 설치 디렉터리>/hub/server/resources/hub

3. 응용 프로그램 JAR 파일을 실행합니다.

참고: 클라이언트 시스템에 SSL 인증서가 없는 경우 HTTPS 연결을 통해 Hub 콘솔에 액세스하려면 SSL 인증서를 설치해야 합니다. 다음 절차를 사용하여 설치할 수 있습니다.

- 다음 명령을 실행하여 로컬 클라이언트 시스템의 Java 키 저장소로 인증서를 가져옵니다.

```
keytool -import -trustcacerts -alias <certificate alias name> -file <certificate alias file> -keystore <local java cacerts keystore location>
```

- 다음 명령을 실행하여 인증서가 포함된 트러스트 저장소 파일의 위치 및 암호를 전달합니다.

```
java -Djavax.net.ssl.trustStore=<truststore file location> -Djavax.net.ssl.trustStorePassword=<truststore_password> -jar hubConsole.jar
```

기본 cacert 파일 대신 모든 사용자 지정 트러스트 인증서가 포함된 개별 트러스트 저장소를 사용합니다. 응용 프로그램 서버를 유지 관리하는 팀으로부터 인증서를 확보합니다. 서버는 자체 서명 인증서 또는 보안 인증서를 사용할 수 있습니다. .jar 파일은 서버에서 버전이 변경된 경우에만 다운로드합니다. .jar 파일을 다운로드할 때마다 동일한 명령을 사용하여 실행합니다.

4. 응용 프로그램에 대한 최대 메모리 할당 풀을 지정하려면 다음 명령을 실행합니다.

```
java -Xmx<n>G -jar hubConsole.jar
```

여기서 <n>은 GB 단위의 최대 메모리 할당입니다.

Informatica MDM Hub 로그인 대화 상자가 표시됩니다.

5. 사용자 이름과 암호를 입력합니다.

6. 특정 Hub 서버 노드에 연결하거나 로드 균형 조정기 또는 역방향 프록시 서버를 사용하려는 경우 연결 속성 필드에 미리 구성되어 있는 연결 매개 변수를 재정의합니다.

다음 형식으로 매개 변수를 입력합니다.

```
<host name>:<port name>
```

여기서 호스트 이름과 포트 이름은 Hub 서버의 호스트 이름과 포트 이름이거나, 사용하는 로드 균형 조정기 또는 역방향 프록시 서버의 호스트 이름과 포트 이름입니다.

7. **확인**을 클릭합니다.

데이터베이스 변경 대화 상자가 표시됩니다.

8. 대상 데이터베이스를 선택합니다.

대상 데이터베이스는 MDM Hub 마스터 데이터베이스입니다.

9. 목록에서 언어를 선택하고 **연결**을 클릭합니다.

Hub 콘솔 사용자 인스턴스가 선택한 언어로 표시됩니다. Hub 콘솔 사용자 인터페이스에서 표시되는 언어를 변경해야 하는 경우 언어를 선택하여 Hub 콘솔을 다시 시작합니다.

연산 참조 저장소 등록

연산 참조 저장소를 생성한 후 Hub 콘솔을 통해 등록해야 합니다. 연산 참조 저장소를 단일 MDM Hub 마스터 데이터베이스로 등록합니다.

1. Hub 콘솔을 시작합니다.

데이터베이스 변경 대화 상자가 표시됩니다.

2. **MDM Hub 마스터 데이터베이스**를 선택하고 **연결**을 클릭합니다.

3. 구성 작업 영역에서 **데이터베이스 도구**를 클릭합니다.

4. 쓰기 잠금 메뉴에서 **잠금 획득**을 클릭합니다.

5. 데이터베이스 창에서 **데이터베이스 등록** 단추를 클릭합니다.

Informatica MDM Hub 연결 마법사가 표시됩니다.

6. Oracle 데이터베이스 유형 옵션을 선택하고 **다음**을 클릭합니다.
7. 데이터베이스의 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 설명되어 있습니다.

속성	설명
데이터베이스 표시 이름	Hub 콘솔에 표시되어야 하는 연산 참조 저장소 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	TNSNAMES.ORA 파일에 정의된 네트워크에 대한 데이터베이스 이름입니다. Oracle 데이터베이스를 설치할 때 Oracle TNS 이름을 설정합니다. Oracle TNS 이름에 대한 자세한 내용은 Oracle 설명서를 참조하십시오.
스키마 이름	연산 참조 저장소의 이름입니다.
암호	연산 참조 저장소 사용자 이름과 연결된 암호입니다.
DDM 연결 URL	선택 사항입니다. Dynamic Data Masking 응용 프로그램에 연결하는 데 사용하는 URL입니다. 이 URL은 Dynamic Data Masking 응용 프로그램 URL이 Dynamic Data Masking 호스트 이름 및 포트 번호를 사용하는 것을 제외하면 데이터베이스에 연결하는 데 사용하는 URL과 유사합니다.

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 유형의 경우 다른 연결 URL을 사용자 지정하고 나중에 테스트할 수 있습니다.
등록 후 데이터 소스 생성	등록 후 응용 프로그램 서버에 대해 데이터 소스를 생성하려면 선택합니다. 해당 옵션을 선택하지 않는 경우에는 데이터 소스를 수동으로 구성해야 합니다. 참고: 응용 프로그램 서버 클러스터 환경에서 연산 참조 저장소에 대한 데이터 소스 및 연결 풀을 수동으로 생성합니다.

- 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 연산 참조 저장소를 등록합니다.

10. 등록된 연산 참조 저장소를 선택하고 **데이터베이스 연결 테스트** 단추를 클릭합니다.

데이터베이스 테스트 대화 상자에 데이터베이스 연결 테스트 결과가 표시됩니다.

11. **확인**을 클릭합니다.

연산 참조 저장소가 등록되고 데이터베이스에 대한 연결이 테스트됩니다.

Oracle Configuration Manager 등록(선택 사항)

Oracle Configuration Manager(OCM)는 MDM 환경에 설치된 응용 프로그램과 데이터베이스 간의 모든 트랜잭션에 대한 데이터를 수집합니다. Hub 콘솔에서 단일 MDM Hub 마스터 데이터베이스를 통해 OCM을 등록해야 합니다.

1. Hub 콘솔을 시작합니다.

데이터베이스 변경 대화 상자가 표시됩니다.

2. **MDM Hub 마스터 데이터베이스**를 선택하고 **연결**을 클릭합니다.

3. 구성 작업 영역에서 **데이터베이스 도구**를 클릭합니다.

4. 쓰기 잠금 메뉴에서 **잠금 획득**을 클릭합니다.

5. 데이터베이스 창에서 **데이터베이스 등록** 단추를 클릭합니다.

Informatica MDM Hub 연결 마법사가 표시됩니다.

6. 서비스 Oracle 연결 방법을 선택하고 다음을 클릭합니다.
연결 속성 페이지가 표시됩니다.
7. 연결 URL 필드에 다음 OCM 서버 연결 URL을 입력합니다.

```
jdbc:oracle:thin:@(description=(address_list=(address=(protocol=tcps)(port=2484)
(host=invlrcon19.informatica.com))(address=(protocol=tcp)(port=1521)
(host=invlrcon21.informatica.com))))(connect_data=(SERVICE_NAME=ORCL19C.informatica.com)))
```
8. 마침을 클릭합니다.
데이터베이스 등록 대화 상자가 표시됩니다.
9. 확인을 클릭합니다.
MDM Hub가 Oracle Configuration Manager를 등록합니다.

ActiveVOS용 Oracle Configuration Manager 설정(선택 사항)

사용자 지정된 JDBC(Java Database Connectivity) URL을 추가하여 Oracle Configuration Manager를 지원하기 위한 ActiveVOS를 설정할 수 있습니다.

1. 명령 프롬프트를 엽니다.
2. 다음 디렉터리로 이동합니다.

```
infamdm installation directory>\conf\avos.install.properties
```
3. avos.jdbc.url 속성을 편집합니다. 다음 사용자 지정 JDBC URL 링크를 입력합니다.

```
jdbc:oracle:thin:@(description=(address_list=(address=(protocol=tcps)(port=2484)
(host=invlrcon19.informatica.com))(address=(protocol=tcp)
(port=1521)(host=invlrcon21.informatica.com))))(connect_data=(SERVICE_NAME=ORCL19C.informatica.com)))
```
4. avos.jdbc.database.url 속성을 편집합니다. 다음 ActiveVOS 데이터베이스 연결 URL을 입력합니다.

```
jdbc.database.url=jdbc:oracle:thin:@(description)=(address_list)=(address)=(protocol=tcps)(port
\=2484)(host=invlrcon19.informatica.com))(address)=(protocol=tcp)(port=1521)(host
\=invlrcon21.informatica.com))))(connect_data)=(SERVICE_NAME=ORCL19C.informatica.com)))
```
5. 다음 디렉터리로 이동합니다.

```
<MDM Hub installation directory>\hub\server\bin\build.properties
```
6. avos.jdbc.database.url 속성을 편집합니다. 다음 ActiveVOS 데이터베이스 연결 URL을 입력합니다.

```
activevos.db.jdbc.url=jdbc:oracle:thin:@(description=(address_list=(address=(protocol=tcps)
(port=2484)(host=invlrcon19.informatica.com))(address=(protocol=tcp)(port=1521)
(host=invlrcon21.informatica.com))))(connect_data=(SERVICE_NAME=ORCL19C.informatica.com)))
```
7. 명령 프롬프트를 엽니다.
8. 다음 디렉터리로 이동합니다.

```
<MDM Hub installation directory>/hub/server/bin
```
9. 다음 명령을 실행합니다.
 - UNIX의 경우.

```
sip_ant.sh avos.deploy -Ddatabase.password=<master database password> -Davos.username=<activeVOS
console username> -Davos.password=<activeVOS console password> -
Davos.jdbc.database.username=<avos database username> -Davos.jdbc.database.password=<avos
database Password> -Davos.jdbc.database.url=<JDBC database URL>
```

- Windows의 경우.

```
sip_ant.bat avos.deploy -Ddatabase.password=<master database password> -Davos.username=<activeVOS console username> -Davos.password=<activeVOS console password> -Davos.jdbc.database.username=<avos database username> -Davos.jdbc.database.password=<avos database Password> -Davos.jdbc.database.url=<JDBC database URL>
```

ActiveVOS가 OCM으로 설정됨

추가 응용 프로그램 서버 구성(선택 사항)

MDM Hub 환경의 요구 사항을 기반으로 추가 JBoss 구성을 수행합니다.

다음 테이블에는 수행할 수 있는 구성이 설명되어 있습니다.

구성	설명
독립 실행형 처리 서버 인스턴스에 대한 JBoss 구성	다음과 같은 시나리오에서 독립 실행형 처리 서버 인스턴스에 대한 JBoss를 구성하는 데 필요합니다. <ul style="list-style-type: none"> - Hub 서버가 설치되지 않은 JBoss 인스턴스에 처리 서버 인스턴스를 설치하려고 합니다. - 여러 독립 실행형 처리 서버 인스턴스를 설치하려고 합니다.
EJB 보안 구성	JBoss 응용 프로그램 서버 수준에서 EJB 보안을 구성하려는 경우에 필요합니다.

독립 실행형 처리 서버 인스턴스에 대한 JBoss 구성

Hub 서버가 설치되지 않은 JBoss 인스턴스에 처리 서버 인스턴스를 설치하려면 JBoss 인스턴스를 구성합니다. 또한 여러 처리 서버 인스턴스를 설치하려면 각각의 추가 처리 서버에 대해 별도의 JBoss 인스턴스를 구성합니다.

- ▶ Hub 서버의 JBoss 인스턴스에서 JDBC 파일을 처리 서버를 배포할 JBoss 인스턴스로 복사합니다.

JDBC 파일은 다음 디렉터리에 있습니다. <JBoss 설치 디렉터리>/modules/com/informatica/mdm/jdbc/main

EJB 보안 구성

MDM Hub의 데이터 및 기타 리소스에 무단으로 액세스하지 못하도록 JBoss 응용 프로그램 서버 수준이나 Hub 서버 수준에서 EJB 보안을 구성할 수 있습니다. JBoss 응용 프로그램 서버 수준에서 EJB 보안을 구성하려면 JBoss Remoting-Connector 보안을 활성화합니다.

1. Hub 콘솔에 로그인하고 Hub 콘솔 암호를 JBoss 암호 정책을 따르는 암호로 변경합니다.

참고: 설정한 암호가 JBoss 암호 정책 및 구성된 MDM Hub 글로벌 암호 정책을 준수하는지 확인합니다. 이는 Hub 콘솔과 JBoss에 대해 동일한 암호가 설정되어야 하기 때문에 중요합니다.

MDM Hub 글로벌 암호 정책에 대한 자세한 내용은 *Multidomain MDM 구성 가이드*를 참조하십시오.

- a. Remoting-Connector 보안을 비활성화합니다.
- b. JBoss 암호 정책을 따르도록 Hub 콘솔 암호를 변경합니다.

1. Hub 콘솔에 로그인하고 데이터베이스를 MDM Hub 마스터 데이터베이스로 변경한 다음 **연결**을 클릭합니다.

2. 구성 작업 영역에서 **사용자** 도구를 선택하고 쓰기 잠금을 획득합니다.
 3. **사용자** 탭에서 admin 사용자를 선택하고 **암호 변경** 아이콘을 클릭합니다.
암호 변경 대화 상자가 나타납니다.
 4. JBoss 암호 정책을 따르도록 암호를 변경하고 **확인**을 클릭합니다.
2. standalone-full.xml 파일에서 Remoting-Connector 보안을 활성화합니다.
 - a. 다음 디렉터리에서 standalone-full.xml 파일을 엽니다.
<JBoss 설치 디렉터리>/standalone/configuration
 - b. 다음 코드에 표시된 대로 security-realm 특성을 추가합니다.
이름이 urn:jboss:domain:remoting:<n.n>인 하위 시스템에서 다음과 같은 특성을 remoting-connector 구성에 추가합니다.
security-realm=<"보안 영역 이름">
3. JBoss에서 Hub 서버에 대한 액세스 권한이 있어야 하는 MDM Hub 사용자를 등록합니다.
 - a. MDM Hub 사용자를 등록하기 위한 다음과 같은 스크립트를 실행합니다.
UNIX의 경우. <JBoss 설치 디렉터리>/bin/add-user.sh
Windows의 경우. <JBoss 설치 디렉터리>\bin\add-user.bat
 - b. 다음 테이블의 프롬프트에 응답합니다.

프롬프트	값
어떤 사용자 유형을 추가하시겠습니까? a) 관리 사용자(mgmt-users.properties) b) 응용 프로그램 사용자(application-users.properties)	b 옵션을 지정합니다.
영역(ApplicationRealm)	보안 영역의 이름을 사용합니다. 기본값은 ApplicationRealm입니다.
사용자 이름	MDM Hub 사용자의 이름입니다.
암호	JBoss 암호 정책을 준수하는 MDM Hub 사용자의 암호입니다.
이 사용자에게 어떤 역할을 지정하시겠습니까?	hubconsole을 지정해야 합니다.

4. Hub 서버에서 JBoss 보안을 활성화합니다.
 - a. 다음 디렉터리에서 cmxserver.properties를 엽니다.
<MDM Hub 설치 디렉터리>/hub/server/resources
 - b. 다음 속성의 설명을 제거합니다.
#cmx.jboss7.security.enabled=true
5. Hub 서버 EAR 파일을 다시 패키징합니다.
 - a. 다음 디렉터리로 이동합니다.
<MDM Hub 설치 디렉터리>/hub/server/bin
 - b. 다음 명령을 실행합니다.

UNIX의 경우.

```
./sip_ant.sh repack
```

Windows의 경우.

```
sip_ant.bat repack
```

6. Hub 서버 EAR 파일을 배포합니다.

- a. JBoss 응용 프로그램 서버를 중지합니다.
- b. 기존 배포가 있는 경우 JBoss 배포 디렉터리에서 다음 배포 파일을 제거합니다.

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.
ave_jboss.ear	필수 사항입니다. ActiveVOS 서버 응용 프로그램입니다.
activevos-central.war	필수 사항입니다. ActiveVOS Central 응용 프로그램입니다.

- c. 소스에서 대상 디렉터리로 이전 단계에서 나열된 배포 파일을 복사합니다.

소스: Hub 서버 설치 디렉터리

대상: <JBoss 설치 디렉터리>\standalone\deployments

- d. 응용 프로그램 서버를 시작합니다.

응용 프로그램 배포에 대한 자세한 내용은 JBoss 설명서를 참조하십시오.

제 7 장



처리 서버 설치

이 장에 포함된 항목:

- [처리 서버 설치, 70](#)
- [설치 프로그램 워크플로우 검토, 71](#)
- [설치 값 수집, 72](#)
- [설치 마법사에서 처리 서버 설치, 74](#)
- [명령줄에서 처리 서버 설치\(UNIX만 해당\), 75](#)
- [자동으로 처리 서버 설치, 76](#)
- [클러스터의 노드에 처리 서버 설치, 76](#)

처리 서버 설치

설치 마법사나 자동 설치 스크립트를 사용하여 또는 UNIX 시스템의 경우 명령줄 스크립트를 사용하여 처리 서버를 설치할 수 있습니다. 설치 프로그램을 시작하기 전에 설치 전 태스크를 완료하고 필요한 정보를 수집한 경우 설치 프로세스에는 약 15분이 소요됩니다.

	중지! 설치 전 태스크를 완료했습니까? 설치 프로그램을 실행하기 전에 설치 전 태스크를 완료하지 않았다면 설치가 실패합니다.
	설치 준비 검사 목록 <ul style="list-style-type: none"><input type="checkbox"/> MDM 구현 계획을 생성했습니다.<input type="checkbox"/> 서버가 시스템 요구 사항을 충족하는지 확인했습니다.<input type="checkbox"/> 운영 체제 및 소프트웨어 버전이 지원되는지 확인했습니다.<input type="checkbox"/> 운영 체제 및 소프트웨어 버전의 알려진 제한을 검토했습니다.<input type="checkbox"/> 지원되는 응용 프로그램 서버 버전을 설치하고 구성했습니다.<input type="checkbox"/> 지원되는 데이터베이스 관리 시스템 버전을 설치하고 구성했습니다.<input type="checkbox"/> 환경에 맞는 설치 전 구성 태스크를 수행했습니다.<input type="checkbox"/> 액세스 가능한 위치에 MDM 라이선스 파일을 저장했습니다.

누락된 태스크가 있는 경우 태스크 완료에 대한 도움을 받으려면 이전 장으로 돌아가십시오.

진행할 준비가 되었으면 다음 단계를 수행합니다.

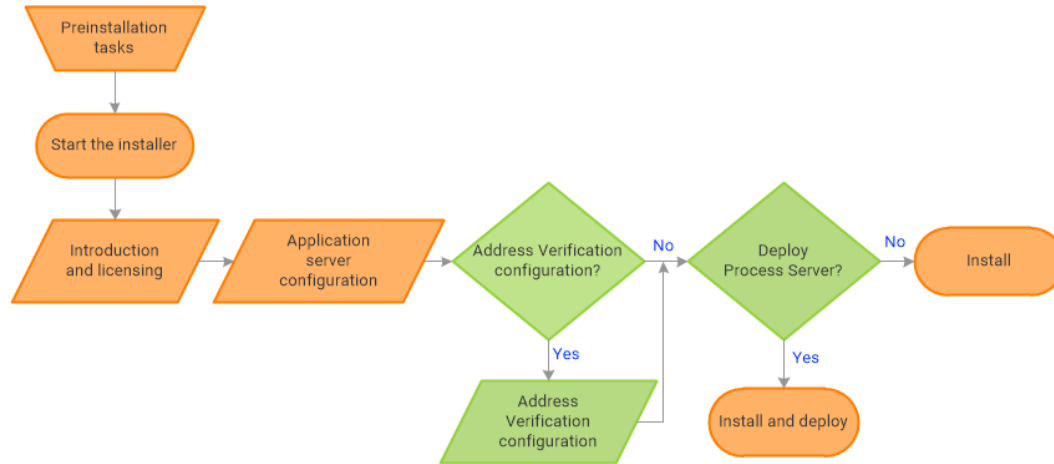
1. 설치 워크플로우를 검토합니다.

2. 설치 중에 입력할 값을 수집합니다.
3. 설치 마법사나 명령줄을 사용하여 또는 자동 모드에서 설치를 시작합니다.

설치 프로그램 워크플로우 검토

설치 마법사, 명령줄 프롬프트 또는 자동 설치 스크립트 중 무엇을 사용하든 설치 프로그램은 동일한 순서로 단계를 진행합니다. 결정 지점을 염두에 두고 설치 워크플로우를 따라야 합니다.

다음 다이어그램은 처리 서버 설치 프로그램 워크플로우의 단계를 안내합니다.



설치 프로세스에서 결정 지점을 식별하십시오. 구현 계획을 참조하여 다음과 같은 각 결정 지점에서 어떤 경로를 선택할지 이해해야 합니다.

1. **Address Verification을 구성합니까?** 관리자는 Informatica 주소 확인을 사용하여 레코드에 포함된 주소를 해석하고, 처리하고, 형식화합니다. 권한이 있는 사용자는 마스터 데이터가 업데이트되기 전에 레코드에 포함된 주소의 유효성을 검사하고, 수정하고, 보증합니다.

참고: 이미 지원되는 버전의 Informatica 주소 확인을 설치했다면 구성 및 매개 변수 파일 위치와 수행할 주소 처리 유형을 구성합니다. 그렇지 않은 경우라면 기본 파일 위치를 사용하고 수행할 주소 유효성 검사 유형을 지정합니다.

2. **처리 서버를 배포합니까?** 설치 프로그램에서 `postInstallSetup` 스크립트를 실행하도록 허용합니다. 다른 중요 태스크 중에서 스크립트는 처리 서버를 응용 프로그램 서버에 배포합니다. 또는 설치 프로그램을 종료한 후에 `postInstallSetup` 스크립트를 실행할 수 있습니다.

설치 값 수집

설치 프로그램을 실행하기 전에 설치 값을 수집합니다. 설치 프로그램이 응용 프로그램 서버, 데이터베이스 관리 시스템 및 기타 구성 요소에 대한 정보를 묻습니다. 가장 좋은 방법은 이러한 테이블을 인쇄하고 해당 환경의 값을 추가하는 것입니다.

응용 프로그램 서버: Red Hat JBoss

설치에 필요한 JBoss 세부 정보를 수집하려면 다음 테이블을 사용합니다.

속성	설명	기본값	서버 인스턴스 1 값
JBoss 설치 디렉터리	JBoss가 설치된 위치입니다. 참고: bin 디렉터리를 포함하지 마십시오.	-	
구성 이름	기본 구성 이름을 사용합니다. 처리 서버가 지정된 구성에 배포됩니다.	독립 실행형	독립 실행형
원격 포트	JBoss 원격 포트입니다. 클러스터된 환경에서 여러 포트를 식별합니다. 참고: JBoss 원격 포트가 기본 데이터베이스 서버 포트와 충돌할 수 있습니다. 포트가 충돌하는 경우 포트 중 하나를 변경하여 충돌을 해결합니다. 포트 변경에 대한 자세한 내용은 JBoss 설명서를 참조하십시오.	4447	
관리 포트	JBoss HTTP 관리 포트입니다. HTTPS 관리 포트는 지원되지 않습니다. 응용 프로그램 서버에 포트 오프셋이 구성된 경우 기본값에 오프셋 값을 추가합니다. 예를 들어 포트 오프셋이 100이면 관리 포트 값은 9990 + 100인 10090입니다.	9990	

클러스터된 환경에서 기타 JBoss 인스턴스에 대한 세부 정보를 다음 테이블에 추가합니다.

속성	서버 인스턴스 2 값	서버 인스턴스 3 값	서버 인스턴스 4 값
JBoss 설치 디렉터리			
구성 이름			
원격 포트			
관리 원시 포트			

Informatica Address Verification

참고: Address Verification의 이전 이름은 Informatica AddressDoctor입니다.

Informatica 주소 확인을 설치하려는 경우 다음 속성을 기록합니다.

속성	설명	기본값 값	서버 인스턴스 1 값
구성 파일	Informatica 주소 확인 구성 파일 SetConfig.xml의 위치입니다. 참조 주소 데이터베이스, Informatica Address Verification의 잠금 해제 코드 및 메모리 설정과 같은 일반 구성에 이 파일을 사용합니다. Address Verification을 이미 설치한 경우에는 구성 파일의 위치를 사용합니다. 그렇지 않은 경우 기본 파일 위치를 사용합니다.	<MDM Hub 설치 디렉터리> \hub\cleanse\resources \AddressDoctor \5\SetConfig.xml	
매개 변수 파일	Informatica 주소 확인 매개 변수 파일 Parameters.xml의 위치입니다. Informatica Address Verification의 주소 해석, 처리 및 서식 지정 방법을 구성할 때 이 파일을 사용합니다. Address Verification을 이미 설치한 경우에는 매개 변수 파일의 위치를 사용합니다. 그렇지 않은 경우 기본 파일 위치를 사용합니다.	<MDM Hub 설치 디렉터리> \hub\cleanse\resources \AddressDoctor \5\Parameters.xml	
수정 유형	수행하려는 주소 처리 유형입니다. 다음 연결 유형 중 하나를 사용합니다. - PARAMETERS_DEFAULT. 기본 수정 유형입니다. Parameters.xml 파일에 정의된 수정 유형의 사용을 나타냅니다. - PARSE_ONLY. 주소 요소를 구문 분석하고 해당하는 필드에 할당합니다. - CORRECT_ONLY. 우편 데이터를 기준으로 주소 유효성을 검사하고 주소를 수정합니다. - CERTIFY_ONLY. 우편 인증에 따라 주소 유효성을 검사하여 국가별 우체국의 요구 사항을 충족합니다. - CORRECT_THEN_CERTIFY. 우편 데이터를 기준으로 주소 유효성을 검사하고 주소를 수정합니다. 그런 다음 우편 인증에 따라 주소 유효성을 검사하여 국가별 우체국의 요구 사항을 충족합니다. - TRY_CERTIFY_THEN_CORRECT. 우편 인증에 따라 주소 유효성을 검사하여 국가별 우체국의 요구 사항을 충족합니다. 주소 유효성 검사에 실패하는 경우 프로세스에서 우편 데이터를 기준으로 주소 유효성을 검사하고 주소를 수정합니다.	PARAMETERS_DEFAULT	

제품 사용 톨킷

이 제품 사용 톨킷은 MDM 환경에 대한 정보를 Informatica로 보냅니다. 이 정보는 Informatica 글로벌 고객 지원에서 환경에 대한 권장 사항을 제공하고 문제를 해결하는 데 사용됩니다. 톨킷을 통해 Informatica로 정보를 보내지 않으려는 경우 MDM 설치 후 톨킷을 비활성화할 수 있습니다.

제품 사용 툴킷 설치에 필요한 세부 정보를 수집하려면 다음 테이블을 사용합니다.

속성	설명	기본값	설치 값
산업	조직의 비즈니스에 가장 부합하는 산업 유형입니다.	-	
환경	설치 환경의 유형입니다. 명령줄에서 설치하는 경우 다음 중 하나를 입력합니다. - 1. 제품 환경 - 2. 테스트 또는 QA 환경 - 3. 개발 환경	-	
네트워크에서 프록시 서버를 사용합니까?	예인 경우 프록시 서버에 대한 세부 정보를 제공합니다.	아니요	
호스트	프록시 서버의 이름 또는 IP 주소입니다.	-	
포트	프록시 서버에 사용되는 포트 번호입니다.	-	
도메인 이름	프록시 서버가 도메인의 일부인 경우 도메인의 이름입니다.	-	
사용자 이름	보안 프록시 서버를 사용하는 경우 MDM에 액세스할 수 있는 사용자의 이름입니다.	-	
암호	사용자의 암호입니다.	-	

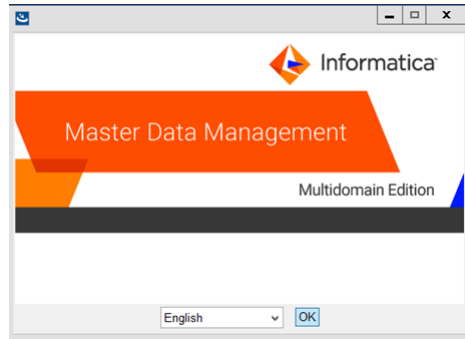
설치 마법사에서 처리 서버 설치

그래픽 모드에서 처리 서버를 설치하려는 경우 설치 마법사를 사용합니다. 설치 마법사는 설치 과정을 안내합니다.

동일한 사용자 이름을 사용하여 Hub 서버와 처리 서버를 설치해야 합니다.

- 응용 프로그램 서버를 시작합니다.
- 다음 디렉터리로 이동합니다.
<MDM Hub 배포 디렉터리>/<운영 체제 이름>/mrncleanse
- 사용 중인 운영 체제에 해당하는 태스크를 수행하여 설치 프로그램을 시작합니다.
 - UNIX.** 명령줄에서 ./hub_cleanse_install.bin을 실행합니다.
 - Windows.** 파일 탐색기에서 hub_cleanse_install.exe를 두 번 클릭합니다.

처리 서버 설치 마법사가 시작됩니다.



4. 언어를 선택하고 **확인**을 클릭합니다.
소개 창이 표시됩니다.
5. 온라인 지침을 따릅니다. 메시지가 표시되면 수집한 설치 값을 입력합니다.
6. 설치가 끝나면 **구성 요약** 창에서 선택한 옵션을 검토합니다.
7. 내용을 변경하려면 **이전**을 클릭하여 해당 창으로 돌아갑니다. 완료했으면 **다음**을 클릭하여 마지막 창으로 이동합니다.
8. **설치**를 클릭합니다.
9. **다음 단계**: 다음 단계는 설치 프로그램에서 처리 서버를 배포하기로 선택했는지 여부에 따라 다릅니다.
 - 설치 프로그램에서 처리 서버를 배포하기로 선택한 경우에는 설치 후 태스크의 일부로 처리 서버를 배포할 필요가 없습니다.
 - 처리 서버를 나중에 배포하기로 선택했다면 설치 후 태스크의 일부로 처리 서버를 배포해야 합니다.

명령줄에서 처리 서버 설치(UNIX만 해당)

UNIX의 경우 명령줄에서 처리 서버를 설치할 수 있습니다. 스크립트를 실행하여 명령줄 설치를 시작합니다.

1. 응용 프로그램 서버를 시작합니다.
2. 명령줄에서 다음 디렉터리로 이동합니다.
`<MDM Hub 배포 디렉터리>/<운영 체제 이름>/mrncleanse`
3. 다음 명령을 실행합니다.
`./hub_cleanse_install.bin -i console`
처리 서버 설치 프롬프트가 나타납니다.
4. 수집한 설치 값을 입력합니다.
괄호에 표시된 기본값을 사용하려면 **Enter** 키를 누릅니다.
5. **다음 단계**: 설치 완료 후 다음 단계는 처리 서버를 배포하기로 선택했는지 여부에 따라 다릅니다.
 - 설치 프로그램에서 처리 서버를 배포하기로 선택한 경우에는 설치 후 태스크의 일부로 처리 서버를 배포할 필요가 없습니다.
 - 처리 서버를 나중에 배포하기로 선택했다면 설치 후 태스크의 일부로 처리 서버를 배포해야 합니다.

자동으로 처리 서버 설치

자동 모드에서 처리 서버를 설치할 수 있습니다. 자동 설치를 시작하기 전에 자동 설치 속성 파일을 구성했는지 확인합니다.

1. 응용 프로그램 서버를 시작합니다.
2. 자동 설치 속성 파일을 대상 환경으로 복사합니다.
3. 대상 환경에서 사용 중인 운영 체제에 해당하는 명령을 실행합니다.

- **UNIX.** `./hub_install.bin -f <편집된 설치 프로그램 속성 파일의 절대 경로>`
- **Windows.** `hub_install.exe -f <편집된 설치 프로그램 속성 파일의 절대 경로>`

자동 설치 프로그램이 백그라운드에서 실행됩니다. 이 프로세스에는 다소 시간이 걸릴 수 있습니다.

4. 설치 프로그램을 통해 처리 서버를 배포하도록 선택한 경우에는 `postinstallSetup.log`를 확인하여 설치가 제대로 되었는지 확인합니다.

log 파일은 다음 디렉터리에 있습니다.

<MDM Hub 설치 디렉터리>/hub/server/logs

5. **다음 단계:** 설치 완료 후 다음 단계는 처리 서버를 배포하기로 선택했는지 여부에 따라 다릅니다.
 - 설치 프로그램에서 처리 서버를 배포하기로 선택한 경우에는 설치 후 태스크의 일부로 처리 서버를 배포할 필요가 없습니다.
 - 처리 서버를 나중에 배포하기로 선택했다면 설치 후 태스크의 일부로 처리 서버를 배포해야 합니다.

클러스터의 노드에 처리 서버 설치

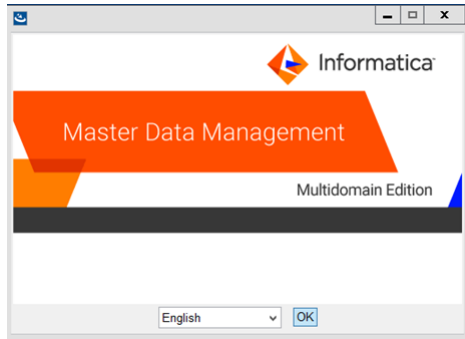
응용 프로그램 서버 클러스터 환경에서, 처리 서버 응용 프로그램을 배포해야 하는 클러스터의 모든 노드에 처리 서버를 설치합니다. 클러스터의 노드 하나에서 설치를 완료한 이후에 클러스터의 다른 노드에서 설치를 진행해야 합니다.

예를 들어 JBoss 클러스터에 `host1` 및 `host2`에서 실행되고 RMI 포트 `4447` 및 `4448`을 사용하는 노드 두 개가 있다고 가정해 보겠습니다. 처리 서버를 `node1`과 `node2`에 설치해야 합니다. 노드 하나(예: `node2`)에서 처리 서버 설치를 완료한 이후 다른 노드(`node1`)에서 설치를 시작해야 합니다.

처리 서버 설치의 디렉터리 구조가 모든 노드에서 동일한지 확인합니다.

1. 각 클러스터 노드에서 JBoss 응용 프로그램 서버를 시작합니다.
2. 다음 디렉터리로 이동합니다.
<MDM Hub 배포 디렉터리>/<운영 체제 이름>/mrncleanse
3. 다음 디렉터리로 이동합니다.
<MDM Hub 배포 디렉터리>/<운영 체제 이름>/mrncleanse
4. 클러스터 노드에서 처리 서버 설치 프로그램을 시작하려면 사용 중인 운영 체제에 맞는 명령을 실행합니다.
UNIX. `./hub_cleanse_install.bin`
Windows. `hub_cleanse_install.exe`

처리 서버 설치 마법사가 시작됩니다.



5. 온라인 지침을 따릅니다. 메시지가 표시되면 수집한 설치 값을 입력합니다.
6. **다음 단계:** 설치가 완료되면 설치가 있는 모든 노드에 처리 서버를 수동으로 배포합니다.
참고: 기본 노드에 대한 자동 배포를 수행한 경우 기본 노드에서 수동으로 배포할 필요가 없습니다.

제 8 장

처리 서버 설치 후 태스크

이 장에 포함된 항목:

- [설치 로그 파일 복사, 78](#)
- [버전 및 빌드 번호 확인, 79](#)
- [처리 서버 응용 프로그램 배포\(조건부\), 79](#)
- [처리 서버에 대한 보안 통신 활성화, 81](#)
- [Elasticsearch 설치 및 구성, 81](#)
- [일치 인구집단 구성, 82](#)
- [처리 서버와 정리 엔진 구성, 83](#)

설치 로그 파일 복사

설치 로그 파일은 처리 서버 설치 프로세스의 문제를 해결하는 데 유용합니다. 로그 파일을 설치 기록 디렉터리에 복사합니다. Informatica 글로벌 고객 지원 센터는 고객이 설치 문제와 관련하여 문의할 경우 로그 파일 사본을 요청할 수 있습니다.

다음 테이블에는 다른 유형의 설치 로그 파일이 설명되어 있습니다.

로그 파일 유형	설명
설치 로그	<ul style="list-style-type: none">- 파일 이름: Informatica_MDM_Cleanse_Match_Server_Install_<타임스탬프>.xml- 위치: <MDM Hub 설치 디렉터리>/hub/cleanse/UninstallerData/Logs- 콘텐츠: 생성된 디렉터리, 설치된 파일과 실행된 명령의 이름 및 설치된 각 파일의 상태입니다.
설치 선행 조건 로그	<ul style="list-style-type: none">- 파일 이름: installPrereq.log- 위치: <MDM Hub 설치 디렉터리>/hub/cleanse/Logs- 콘텐츠: 설치 프로그램이 수행한 선행 조건 검사 로그입니다.
디버그 로그	<ul style="list-style-type: none">- 파일 이름: infamdm_installer_debug.txt- 위치: <MDM Hub 설치 디렉터리>/hub/cleanse/- 콘텐츠: 설치 중 수행된 선택 항목 및 설치 프로그램이 수행한 작업에 대한 자세한 정보입니다.
설치 후 설정 로그	<ul style="list-style-type: none">- 파일 이름: postInstallSetup.log- 위치: <MDM Hub 설치 디렉터리>/hub/cleanse/logs- 콘텐츠: 설치 후 프로세스 중 설치 프로그램이 수행한 작업 및 설치 후 프로세스 오류의 요약입니다.

로그 파일 유형	설명
처리 서버 로그	<ul style="list-style-type: none"> - 파일 이름: cmxserver.log - 위치: <MDM Hub 설치 디렉터리>/hub/cleanse/logs - 콘텐츠: 처리 서버 작업의 요약입니다.
JBoss 로그	<ul style="list-style-type: none"> - 파일 이름: server.log - 위치: <JBoss 설치 디렉터리>/standalone/log - 콘텐츠: JBoss 이벤트 로그 및 오류 메시지를 포함합니다.

버전 및 빌드 번호 확인

올바른 버전 및 빌드 번호의 처리 서버가 설치되어 있는지 확인합니다.

1. 명령 프롬프트를 열고 다음 디렉터리로 이동합니다. <MDM Hub 설치 디렉터리>/hub/cleanse/bin
2. 처리 서버 버전 및 빌드 번호를 확인하려면 다음 명령을 실행합니다.

UNIX의 경우. versionInfo.sh

Windows의 경우. versionInfo.bat

참고: AIX 시스템의 경우 <>/jre/bin 디렉터리에서 Java를 실행하도록 Java 홈versionInfo.sh 스크립트를 변경합니다.

처리 서버 응용 프로그램 배포(조건부)

처리 서버 응용 프로그램을 배포해야 하는 시나리오에서는 처리 서버 응용 프로그램을 배포합니다.

다음과 같은 시나리오에서는 처리 서버 응용 프로그램을 배포해야 합니다.

- 설치가 응용 프로그램 서버 다중 노드 환경 또는 클러스터 환경에서 수행됩니다.
- 설치가 완료되었지만 설치 중 실행한 postInstallSetup 스크립트가 실패합니다.
- 설치 중 postInstallSetup 스크립트를 건너뛰었습니다.

처리 서버 응용 프로그램을 배포하려면 다음 단계를 수행하십시오.

1. 처리 서버가 Hub 서버와 동일한 응용 프로그램 서버 인스턴스에 설치되지 않은 경우 데이터 소스를 생성합니다.
2. 처리 서버 응용 프로그램 siperian-mrm-cleanse.ear을 배포합니다.

1단계. 데이터 소스 생성(조건부)

Hub 서버와 동일한 응용 프로그램 서버 인스턴스에 처리 서버가 배포되지 않은 경우 응용 프로그램 서버의 데이터 소스를 구성합니다.

처리 서버에 대한 데이터 소스를 생성하도록 응용 프로그램 서버를 구성하려면 다음 태스크를 수행합니다.

1. MDM Hub 마스터 데이터베이스 및 각 연산 참조 저장소에 대한 <datasources> 정의를 Hub 서버의 standalone-full.xml 파일에서 처리 서버 시스템의 standalone-full.xml에 복사합니다. standalone-full.xml 파일은 다음 위치에 있습니다.
<JBoss 설치 디렉터리>/standalone/configuration
2. 처리 서버 시스템의 <JBoss 설치 디렉터리>\modules 아래에 다음 디렉터리 구조를 추가합니다.
com\informatica\mdm\jdbc\main
3. Hub 서버 시스템의 <JBoss 설치 디렉터리>\modules에서 다음 파일을 복사합니다.
 - module.xml
 - 지원되는 JDBC 드라이버 버전
4. 처리 서버 시스템에서 복사한 파일을 다음 디렉터리에 붙여 넣습니다. <JBoss 설치 디렉터리>\modules\com\informatica\mdm\jdbc\main

2단계. 처리 서버 응용 프로그램 배포(조건부)

응용 프로그램 서버 다중 노드 환경 또는 클러스터 환경에서 설치를 수행하는 경우나 postInstallSetup 스크립트가 생략되거나 실패한 경우 처리 서버 응용 프로그램을 배포합니다.

처리 서버를 설치한 동일한 시스템에 처리 서버 응용 프로그램을 배포합니다. 처리 서버 응용 프로그램에서 응용 프로그램과 연결된 처리 서버 설치를 찾을 수 있어야 합니다. 그러므로 다른 시스템의 배포에 대한 응용 프로그램 EAR 파일을 복사하지 마십시오. 예를 들어 테스트 시스템에 처리 서버를 설치한 다음 프로덕션 시스템에 응용 프로그램을 배포합니다. 이 경우 프로덕션 시스템에 배포하는 응용 프로그램에서 테스트 시스템에 있는 설치를 찾을 수 없습니다.

다음 절차 중 하나를 사용하여 처리 서버 응용 프로그램을 배포합니다.

배포에 스크립트 사용

postInstallSetup 스크립트를 실행하여 처리 서버 응용 프로그램을 배포합니다.

수동 배포

처리 서버 응용 프로그램을 수동으로 배포합니다.

처리 서버 응용 프로그램 배포에 스크립트 사용(조건부)

PostInstallSetup 스크립트를 실행하여 처리 서버 응용 프로그램을 배포할 수 있습니다.

중요: 응용 프로그램 서버 다중 노드 또는 클러스터 환경에서 설치를 수행하는 경우 한 노드에 처리 서버 응용 프로그램을 먼저 배포하고 다른 노드에 처리 서버 응용 프로그램을 배포합니다. 처리 서버를 설치한 동일한 시스템에 처리 서버 응용 프로그램을 배포해야 합니다.

1. 명령 프롬프트를 열고 다음 디렉터리로 변경합니다.
<MDM Hub 설치 디렉터리>/hub/cleanse
2. PostInstallSetup 스크립트를 실행합니다.
UNIX의 경우. ./postInstallSetup.sh
Windows의 경우. postInstallSetup.bat

처리 서버 응용 프로그램 수동 배포(조건부)

처리 서버 응용 프로그램을 수동으로 배포할 수 있습니다. 처리 서버 설치 디렉터리에서 처리 서버 응용 프로그램을 배포해야 합니다.

중요: 응용 프로그램 서버 다중 노드 또는 클러스터 환경에서 설치를 수행하는 경우 한 노드에 처리 서버 응용 프로그램을 먼저 배포하고 다른 노드에 처리 서버 응용 프로그램을 배포합니다. 처리 서버를 설치한 동일한 시스템에 처리 서버 응용 프로그램을 배포해야 합니다.

1. JBoss 응용 프로그램 서버를 중지합니다.
2. 기존 배포가 있는 경우 다음 JBoss 배포 디렉터리에서 `siperian-mrm-cleanse.ear` 배포 파일을 삭제합니다.
<JBoss 설치 디렉터리>\standalone\deployments
3. `siperian-mrm-cleanse.ear` 배포 파일을 처리 서버 설치 디렉터리에서 JBoss 배포 디렉터리로 복사합니다.
4. JBoss 응용 프로그램 서버를 시작합니다.

응용 프로그램 배포에 대한 자세한 내용은 JBoss 설명서를 참조하십시오.

처리 서버에 대한 보안 통신 활성화

각 처리 서버에는 서명된 인증서가 필요합니다. Hub 콘솔을 사용하여 HTTPS 프로토콜을 활성화하고 각 처리 서버에 대한 보안 포트를 지정합니다.

1. 처리 서버에 대한 서명된 인증서를 인증서 저장소에 생성합니다.
2. 응용 프로그램 서버에서 인증서 저장소에 액세스할 수 있는지 확인합니다.
3. Hub 콘솔에 로그인합니다.
4. 연산 참조 저장소 데이터베이스를 선택합니다.
5. 쓰기 잠금을 획득합니다.
6. 유틸리티 작업 영역에서 **처리 서버**를 선택합니다.
7. 처리 서버를 선택하고 **처리 서버 편집** 아이콘을 클릭합니다.
처리 서버 추가/편집 대화 상자가 열립니다.
8. **포트**가 보안 포트인지 확인합니다.
9. **보안 연결(HTTPS)** 확인란을 선택합니다.
10. **확인**을 클릭합니다.
11. 목록에 나타나는 다른 처리 서버를 확인합니다.

Elasticsearch 설치 및 구성

검색을 사용하려면 MDM Hub용 Elasticsearch를 설치 및 설정하십시오.

검색을 설치 및 구성하는 방법에 대한 자세한 내용은 *Multidomain MDM 구성 가이드*의 "Elasticsearch로 검색" 장을 참조하십시오.

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.

일치 인구집단 활성화

일치 규칙에 사용할 일치 인구집단을 활성화해야 합니다.

1. *<population>.ysp* 파일을 다음 위치에 복사합니다.

UNIX의 경우. *<infamdm_install_directory>/hub/cleanse/resources/match*

Windows의 경우. *<infamdm_install_directory>\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 테이블에 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 사용합니다.

- a. 인구집단을 추가할 연산 참조 저장소 스키마에 연결합니다.

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

UNIX의 경우. *<infamdm_install_directory>/server/resources/database/custom_scripts/oracle*

Windows의 경우. *<infamdm_install_directory>\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.0001 입니다.

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;
```

4. 인구집단을 활성화한 후에 처리 서버를 다시 시작합니다.

5. Hub 콘솔에 로그인하여 인구집단이 활성화되었는지 확인합니다.

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

처리 서버와 정리 엔진 구성

처리 서버를 설치한 후 처리 서버에서 정리 엔진을 구성할 수 있습니다.

정리 엔진 구성에 대한 자세한 내용은 *Multidomain MDM 정리 어댑터 가이드*를 참조하십시오.

제 9 장

응용 프로그램 서버에 대한 ActiveVOS 설치 후 태스크

이 장에 포함된 항목:

- [JBoss에서 ActiveVOS 보안 도메인 편집, 84](#)
- [JBoss 환경에서 트러스트된 사용자 생성, 84](#)

JBoss에서 ActiveVOS 보안 도메인 편집

JBoss에서 컨테이너 기반 인증을 활성화하십시오.

1. 다음 디렉터리에서 `standalone-full.xml` 파일을 엽니다.

<JBoss 설치 디렉터리>/standalone/configuration

2. `security-domain name="ActiveVOS"` 요소에서 `login-module` 구성을 다음 구성으로 바꿉니다.

```
<login-module code="RealmUsersRoles" flag="required">
  <module-option name="usersProperties" value="{jboss.server.config.dir}/application-
users.properties"/>
  <module-option name="rolesProperties" value="{jboss.server.config.dir}/application-
roles.properties"/>
  <module-option name="realm" value="ApplicationRealm"/>
  <module-option name="unauthenticatedIdentity" value="anonymous"/>
</login-module>
```

3. JBoss 응용 프로그램 서버가 실행 중인 경우 서버를 다시 시작합니다.

JBoss 환경에서 트러스트된 사용자 생성

ActiveVOS 워크플로우 엔진을 사용하려면 응용 프로그램 서버에서 `abTrust`, `abServiceConsumer` 및 `abTaskClient` 역할을 가진 트러스트된 사용자를 생성합니다. 트러스트된 사용자는 MDM Hub 및 ActiveVOS에서 트러스트됩니다. 해당 사용자는 MDM Hub와 ActiveVOS 간의 보안 통신을 원활하게 합니다.

트러스트된 사용자는 Hub 콘솔의 ActiveVOS 워크플로우 어댑터 사용자와 같은 사용자입니다. 트러스트된 사용자의 이름은 응용 프로그램 서버 관리자의 이름과 같을 수 없습니다.

1. 명령 프롬프트에서 `add-user.bat`를 실행합니다.

- UNIX: <JBoss 설치 디렉터리>/bin/add-user.sh

- Windows: <JBoss 설치 디렉터리>\bin\add-user.bat

2. 나타나는 프롬프트에 응답합니다.

참고: 프롬프트에 대괄호로 묶인 기본값이 표시됩니다. **Enter** 키를 눌러 기본값을 사용하고 다음 프롬프트로 이동합니다.

어떤 사용자 유형을 추가하시겠습니까? a) 관리 사용자 또는 b) 응용 프로그램 사용자

b를 입력하여 응용 프로그램 사용자를 선택합니다.

영역(ApplicationRealm)

standalone-full.xml 파일에 추가한 login-module에서 지정한 같은 영역 이름을 입력합니다.

사용자 이름

트러스트된 사용자의 사용자 이름을 입력합니다.

암호

JBoss 암호 표준을 준수하는 암호를 입력합니다.

이 사용자에게 어떤 역할을 지정하시겠습니까?

abTrust,abServiceConsumer,abTaskClient를 입력합니다.

영역 <영역 이름>에 대한 사용자 <사용자 이름>을(를) 추가하려고 합니다. 맞습니까?

사용자를 추가하려면 yes를 입력합니다.

하나의 AS 프로세스가 다른 AS 프로세스에 연결하는 데 이 새로운 사용자가 사용됩니까?

yes를 입력합니다.

3. 응용 프로그램 서버를 다시 시작합니다.

제 10 장

비즈니스 항목 어댑터에 대한 ActiveVOS 설치 후 태스크

이 장에 포함된 항목:

- [ActiveVOS 웹 응용 프로그램, 86](#)
- [비즈니스 항목 워크플로우 어댑터의 ActiveVOS URN 구성, 87](#)
- [ActiveVOS 프로토콜을 HTTPS로 설정, 87](#)
- [기본 워크플로우 엔진 구성, 88](#)
- [ActiveVOS의 MDM ID 서비스 구성, 88](#)
- [태스크 구성, 89](#)

ActiveVOS 웹 응용 프로그램

ActiveVOS Server의 번들 라이선스 버전을 설치하면 두 개의 ActiveVOS 웹 응용 프로그램도 사용하도록 허가됩니다. 사용자를 응용 프로그램 서버 컨테이너에 추가하면 이러한 응용 프로그램을 사용할 수 있습니다.

다음과 같이 다양한 용도로 웹 응용 프로그램을 사용합니다.

ActiveVOS Console

관리자는 ActiveVOS Console을 사용하여 배포된 프로세스, 알림 시스템 및 끝점 위치를 관리합니다. 성능 모니터링 및 관리에 대한 엔진도 구성할 수 있습니다.

ActiveVOS Central

비즈니스 사용자는 ActiveVOS Central을 사용하여 태스크, 요청 및 보고서를 관리할 수 있습니다. 그러나 일반적으로 비즈니스 사용자는 태스크 관리자에서 항목을 열어 검토할 수 있기 때문에 IDD(Data Director) 응용 프로그램을 사용하여 태스크를 관리합니다.

ActiveVOS Central을 사용하려면 MDM Hub 사용자를 응용 프로그램 서버 컨테이너에 추가해야 합니다.

웹 응용 프로그램에 대한 자세한 내용은 Informatica ActiveVOS 설명서를 참조하십시오.

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.

기본 워크플로우 엔진 구성

기본 워크플로우 엔진을 구성하려면 비즈니스 항목을 기반으로 ActiveVOS 워크플로우에 대한 워크플로우 엔진을 추가합니다. 보조 워크플로우 엔진은 더 이상 사용되지 않는 워크플로우 엔진을 사용하여 기존 태스크를 처리하려는 기존 고객을 위한 것입니다.

1. Hub 콘솔에서 구성 작업 영역의 **워크플로우 관리자**를 클릭합니다.
2. 쓰기 잠금을 획득합니다.
3. **워크플로우 엔진** 탭을 선택한 후 **추가** 단추를 클릭합니다.
4. **워크플로우 추가** 대화 상자에 워크플로우 엔진 속성을 입력합니다.

다음 테이블에는 워크플로우 엔진 속성이 설명되어 있습니다.

필드	설명
워크플로우 엔진	워크플로우 엔진의 표시 이름입니다.
어댑터 이름	비즈니스 항목을 기반으로 하는 ActiveVOS 워크플로우 어댑터의 경우 BE ActiveVOS를 선택하십시오.
호스트	Informatica ActiveVOS 인스턴스의 호스트 이름입니다.
포트	Informatica ActiveVOS 인스턴스의 게시 이름입니다.
사용자 이름	트러스트된 사용자의 사용자 이름입니다.
암호	트러스트된 사용자의 암호입니다.
프로토콜	MDM Hub와 ActiveVOS 간에 통신하는 프로토콜입니다. http 또는 https를 프로토콜로 사용할 수 있습니다.

5. **확인**을 클릭합니다.

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 admin인 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.

태스크 구성

Informatica Data Director에서 태스크 워크플로우 사용을 시작하기 전에 프로비저닝 도구에서 태스크 템플릿, 태스크 트리거 및 태스크 유형을 구성합니다.

자세한 내용은 *Multidomain MDM 프로비저닝 도구 가이드*를 참조하십시오.

제 11 장

ActiveVOS 사용자 지정

`build.properties` 파일에서 ActiveVOS 속성을 구성하여 ActiveVOS 설치를 사용자 지정할 수 있습니다. 속성에는 기본값이 있습니다. 기본값이 사용자 환경에서 작동하지 않을 경우 기본값을 설정할 필요가 없습니다.

다음 테이블에는 구성할 수 있는 ActiveVOS 속성이 설명되어 있습니다.

속성	설명
<code>activevos.install.console</code>	ActiveVOS 콘솔을 서버에 배포할지 여부를 제어합니다. 기본값은 <code>true</code> 입니다. <code>false</code> 로 설정하면 ActiveVOS 콘솔이 설치되지 않습니다.
<code>activevos.secure.https.only</code>	모든 ActiveVOS HTTP 트래픽을 HTTPS로 강제 리디렉션합니다. 기본값은 <code>false</code> 입니다.

ActiveVOS 속성 추가

ActiveVOS 속성을 활성화하려면 Hub 콘솔에서 `build.properties` 파일에 속성을 추가하고 설치 후 스크립트 실행을 다시 실행하여 변경 사항을 적용해야 합니다.

1. 텍스트 편집기에서 다음 `build.properties` 디렉터리를 엽니다.
`<MDM Hub installation directory>\hub\server\bin\build.properties`
2. ActiveVOS 속성을 `build.properties` 파일에 추가합니다.
3. 명령 프롬프트를 엽니다.
4. 다음 디렉터리에서 `postInstallSetup` 스크립트로 이동합니다.
 - UNIX의 경우. `<MDM Hub 설치 디렉터리>\hub\cleanse/`
 - Windows의 경우. `<MDM Hub 설치 디렉터리>\hub\cleanse`
5. `PostInstallSetup` 스크립트를 실행합니다.
 - UNIX의 경우. `postinstallsetup.sh`.
 - Windows의 경우. `postinstallsetup.bat`
6. 응용 프로그램 서버를 다시 시작합니다.

제 12 장

리소스 키트 설치

이 장에 포함된 항목:

- [MDM Hub 샘플 연산 참조 저장소 설정, 91](#)
- [Informatica MDM Hub 샘플 연산 참조 저장소 등록, 94](#)
- [그래픽 모드에서 리소스 키트 설치, 96](#)
- [콘솔 모드에서 리소스 키트 설치, 98](#)
- [자동 모드에서 리소스 키트 설치, 100](#)

MDM Hub 샘플 연산 참조 저장소 설정

MDM Hub 샘플 연산 참조 저장소를 사용하려면 먼저 설정해야 합니다. 리소스 키트를 설치하기 전에 MDM Hub 샘플 연산 참조 저장소를 설정합니다. MDM Hub 샘플 연산 참조 저장소를 설정하려면 연산 참조 저장소를 생성한 후 `mdm_sample`을 가져옵니다.

1. 배포 디렉터리의 다음 위치로 이동합니다.

UNIX의 경우. <배포 디렉터리>/database/bin

Windows의 경우. <배포 디렉터리>\database\bin

2. 다음 명령을 실행합니다.

UNIX의 경우. `./sip_ant.sh create_ors`

Windows의 경우. `sip_ant.bat create_ors`

3. 나타나는 프롬프트에 응답합니다.

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

데이터베이스 유형을 입력합니다(ORACLE, MSSQL, DB2).

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 **SERVICE**입니다.

연산 참조 저장소 데이터베이스 호스트 이름을 입력합니다. [localhost]

데이터베이스를 호스팅하는 시스템의 이름입니다. 기본값은 localhost입니다.

연산 참조 저장소 데이터베이스 포트 번호를 입력합니다. [1521]

데이터베이스 수신기에서 사용하는 포트 번호입니다. 기본값은 1521입니다.

연산 참조 저장소 데이터베이스 서비스 이름을 입력합니다. [orcl]

Oracle 서비스의 이름입니다. 선택한 Oracle 연결 유형이 SERVICE이면 이 프롬프트가 나타납니다.

연산 참조 저장소 데이터베이스 SID를 입력합니다. [orcl]

Oracle 시스템 식별자의 이름입니다. 선택한 Oracle 연결 유형이 SID이면 이 프롬프트가 나타납니다.

Oracle Net 연결 식별자(TNS 이름) 입력 [orcl]

Oracle TNS 이름입니다. 기본값은 orcl입니다.

연결 URL입니다. [jdbc:oracle:thin:@//<호스트 이름>:<포트>/<service_name 또는 SID>]

데이터베이스 연결용 연결 URL입니다.

연산 참조 저장소 데이터베이스 사용자 이름을 입력합니다. [cmx_ors]

MDM Hub 샘플 연산 참조 저장소 데이터베이스의 사용자 이름입니다. 기본값은 cmx_ors입니다.

연산 참조 저장소 데이터베이스 사용자 암호를 입력합니다.

MDM Hub 샘플 연산 참조 저장소 사용자의 암호입니다.

목록 de, en_US, fr, ja, ko, zh_CN에서 로캘 이름을 입력합니다. [en_US]

운영 체제 로캘입니다. 기본값은 en_US입니다.

DBA 사용자 이름을 입력합니다. [SYS]

관리자의 사용자 이름입니다. 기본값은 SYS입니다.

DBA 암호를 입력합니다.

관리자의 암호입니다.

MDM 인덱스 테이블스페이스의 이름을 입력합니다. [CMX_INDX]

연산 참조 저장소에 대한 인덱스 구성 요소를 포함할 테이블스페이스의 이름입니다. 기본값은 CMX_INDX입니다.

MDM 임시 테이블스페이스의 이름을 입력합니다. [CMX_TEMP]

연산 참조 저장소에 대한 임시 구성 요소를 포함할 테이블스페이스의 이름입니다. 기본값은 CMX_TEMP입니다.

Oracle 임시 테이블스페이스의 이름을 입력합니다. [TEMP]

Oracle 임시 테이블스페이스의 이름입니다. 기본값은 TEMP입니다.

4. 샘플 연산 참조 저장소를 생성한 후 다음 디렉터리에서 sip_ant.log를 검토합니다.

UNIX의 경우. <빠른 디렉터리>/database/bin

Windows의 경우. <빠른 디렉터리>\database\bin

sip_ant.log 파일은 sip_ant 스크립트를 실행하여 연산 참조 저장소를 생성할 때 발생할 수 있는 모든 오류를 기록합니다.

5. mdm_sample을 가져오려면 다음 명령을 실행합니다.

UNIX의 경우. ./sip_ant.sh import_schema

Windows의 경우. sip_ant.bat import_schema

6. 나타나는 프롬프트에 응답합니다.

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

데이터베이스 유형을 입력합니다. (ORACLE, MSSQL, DB2)

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 이름을 입력합니다. [localhost]

데이터베이스를 호스팅하는 시스템의 이름입니다. 기본값은 localhost입니다.

연산 참조 저장소 데이터베이스 포트 번호를 입력합니다. [1521]

데이터베이스 수신기에서 사용하는 포트 번호입니다. Default is 1521.

연산 참조 저장소 데이터베이스 서비스 이름을 입력합니다. [orcl]

Oracle 서비스의 입력하십시오. 선택한 Oracle 연결 유형이 SERVICE이면 이 프롬프트가 나타납니다.

연산 참조 저장소 데이터베이스 SID를 입력합니다. [orcl]

Oracle 시스템 식별자의 이름입니다. 선택한 Oracle 연결 유형이 SID이면 이 프롬프트가 나타납니다.

Oracle Net 연결 식별자(TNS 이름) 입력 [orcl]

Oracle TNS 이름입니다. 기본값은 orcl입니다.

연결 URL입니다. [jdbc:oracle:thin:@//<호스트 이름>:<포트>/<service_name 또는 SID>]

데이터베이스 연결용 연결 URL입니다.

연산 참조 저장소 데이터베이스 사용자 이름을 입력합니다. [cmx_ors]

MDM Hub 샘플 연산 참조 저장소 데이터베이스의 이름입니다. 기본값은 cmx_ors입니다.

연산 참조 저장소 데이터베이스 사용자 암호를 입력하십시오.

MDM Hub 샘플 연산 참조 저장소 데이터베이스 사용자의 이름입니다.

목록 de, en_US, fr, ja, ko, zh_CN에서 로캘 이름을 입력합니다. [en_US]

운영 체제 로캘입니다. 기본값은 en_US입니다.

ZIP 덤프 파일 경로를 입력합니다. [<배포 디렉터리>\resources\database]

mdm_sample.zip 파일 경로입니다.

ZIP 덤프 파일 이름을 입력합니다. [mdm_sample.zip]

ZIP 덤프 파일의 이름입니다. 기본값은 mdm_sample.zip입니다.

Informatica MDM Hub 샘플 연산 참조 저장소 등록

MDM Hub 샘플 연산 참조 저장소를 설정했으면, 등록해야 합니다. Hub 콘솔을 통해 MDM Hub 샘플 연산 참조 저장소를 등록합니다.

1. Hub 콘솔을 시작합니다.

데이터베이스 변경 대화 상자가 표시됩니다.

2. MDM Hub 마스터 데이터베이스를 선택하고 **연결**을 클릭합니다.
3. 구성 작업 영역에서 **데이터베이스** 도구를 시작합니다.
4. **쓰기 잠금 > 잠금 획득**을 클릭합니다.
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 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.

연결 속성 페이지가 표시됩니다.

- b. 선택한 연결 유형에 대한 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	키에 고정된 접두사로서, Hub 저장소 인스턴스에서 레코드를 고유하게 식별합니다.
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 SERVICE의 이름입니다. 서비스 필드는 서비스 연결 유형을 선택한 경우에 표시됩니다.

속성	설명
포트	Oracle 데이터베이스 서버에서 실행 중인 Oracle 수신기의 TCP 포트입니다. 기본값은 1521입니다.
Oracle TNS 이름	응용 프로그램 서버의 TNSNAMES.ORA 파일에서 정의된 네트워크의 데이터베이스 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. mdm_sample를 지정합니다.
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.

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 마법사는 기본적으로 연결 URL을 생성합니다. 다음 목록에는 Oracle 연결 유형에 대한 연결 URL의 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 날짜 소스 생성	등록 후 응용 프로그램 서버에 대해 날짜 소스를 생성하려면 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 MDM Hub 샘플 연산 참조 저장소를 등록합니다.

10. 등록된 MDM Hub 샘플 연산 참조 저장소를 선택하고 **데이터베이스 연결 테스트** 단추를 클릭하여 데이터베이스 설정을 테스트합니다.

데이터베이스 테스트 대화 상자에 데이터베이스 연결 테스트 결과가 표시됩니다.

11. **확인**을 클릭합니다.

연산 참조 저장소가 등록되고 데이터베이스에 대한 연결이 테스트됩니다.

그래픽 모드에서 리소스 키트 설치

그래픽 모드에서 리소스 키트를 설치할 수 있습니다.

리소스 키트를 설치하기 전에 **MDM Hub**을 설치 및 구성해야 합니다.

1. 응용 프로그램 서버를 시작합니다.
2. 명령 프롬프트를 열고 리소스 키트 설치 프로그램으로 이동합니다. 기본적으로 설치 프로그램은 다음 디렉터리에 있습니다.

UNIX의 경우. <배포 디렉터리>/<operating system name>/mrresourcekit

Windows의 경우. <배포 디렉터리>\windows\mrresourcekit

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 옵션을 선택할 수 있습니다.

샘플 스키마

MDM Hub 샘플 스키마 리소스를 설치합니다. 샘플 응용 프로그램을 설치하려면 먼저 샘플 스키마를 생성하여 Hub 서버에 등록해야 합니다.

샘플 및 유틸리티

샘플 응용 프로그램 및 유틸리티를 설치합니다.

배포된 샘플 응용 프로그램 목록은 다음 디렉터리의 build.properties 파일에 저장되어 있습니다.

<Resourcekit_Home>\samples

SIF SDK 및 Javadocs

SIF SDK와 관련된 javadocs, 라이브러리 및 리소스를 설치합니다.

BPM SDK

BPM SDK와 관련된 리소스를 설치합니다.

Jaspersoft

리소스 키트 홈에 Jaspersoft 설치 프로그램을 복사합니다.

SSA-NAME3

리소스 키트 홈에 SSA-NAME3 설치 프로그램을 복사합니다.

MDM Hub에서 샘플 스키마를 생성하고 등록하는 데 대한 요구 사항 관련 메시지가 표시됩니다.

8. **확인**을 클릭합니다.

설치 폴더 선택 창이 표시됩니다.

9. 리소스 키트 설치 위치를 선택합니다.

- 기본 위치를 선택하려면 **다음**을 클릭합니다.
- 경로를 입력하려면 설치 폴더 경로를 입력하고 **다음**을 클릭합니다.

참고: 디렉터리 또는 폴더 이름에 공백이 있는 경로를 지정하면 설치가 실패합니다.

- 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.

10. UNIX의 경우 링크 폴더를 선택하거나 링크를 생성하지 않는 옵션을 선택한 후 **다음**을 클릭합니다.
Windows의 경우 제품 아이콘을 생성할 위치를 선택하거나 제품 아이콘을 생성하지 않는 옵션을 선택합니다.

구성 선택 창이 표시됩니다.

11. 구성 옵션을 선택하고 **다음**을 클릭합니다.

다음 옵션 중 하나를 선택할 수 있습니다.

샘플 구성

샘플을 설치 및 구성합니다.

소스만

샘플 소스를 설치하되 샘플은 구성하지 않습니다.

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

12. **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.

13. 응용 프로그램 서버 설정을 구성합니다.

- a. 응용 프로그램 서버 설치 디렉터리를 지정한 다음 **다음**을 클릭합니다.

JBoss 응용 프로그램 서버 구성 이름 창이 표시됩니다.

- b. 구성 이름을 지정한 다음 **다음**을 클릭합니다.

기본값은 독립 실행형입니다.

- c. 원격 포트를 지정합니다.

Informatica MDM Hub 서버 창이 표시됩니다.

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

다음 필드에 값을 입력합니다.

서버 이름

Hub 서버를 호스팅하는 서버의 이름입니다.

서버 HTTP 포트

Hub 서버의 포트 번호입니다.

Informatica MDM 관리 암호

MDM Hub에 액세스하는 데 사용하는 암호입니다.

MDM Hub 홈 디렉터리

Hub 서버 설치를 위한 디렉터리입니다.

리소스 키트 ORS ID 창이 표시됩니다.

15. 목록에서 리소스 키트 ORS ID를 선택한 후 **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 스키마에 대한 연산 참조 저장소 ID가 표시되지 않습니다. 샘플 연산 참조 저장소를 등록한 다음 설치를 다시 시작합니다.

배포 선택 창이 표시됩니다.

16. 다음 옵션 중 하나를 선택하고 **다음**을 클릭합니다.

예, 설치 중에 실행합니다.

설치하는 동안 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 구성해야 합니다. 이 단계에서 리소스 키트를 배포하지 않으면 리소스 키트에 제공된 **postInstallSetup** 스크립트를 사용하여 샘플을 변경하고 재배포할 수 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 재배포해야 합니다.

설치 전 요약 창이 표시됩니다.

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

설치가 완료되면 **설치 완료** 창이 표시됩니다.

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

콘솔 모드에서 리소스 키트 설치

콘솔 모드에서 리소스 키트를 설치할 수 있습니다.

리소스 키트를 설치하기 전에 **MDM_SAMPLE** 스키마를 등록했는지 확인합니다.

1. 응용 프로그램 서버를 시작합니다.

2. MDM Hub 배포의 다음 디렉터리로 이동합니다.

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

Windows의 경우. <MDM Hub 배포 디렉터리>/windows/resourcekit

3. 명령 프롬프트에서 다음 명령을 실행합니다.

UNIX의 경우. `./hub_resourcekit_install.bin -i console`

Windows의 경우. `hub_resourcekit_install.exe -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. 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. 설치할 리소스 키트 기능의 번호를 점표로 구분하여 입력하고 **Enter** 키를 누릅니다.

샘플 스키마 설치를 묻는 메시지가 표시됩니다.

9. 리소스 키트 설치 폴더를 선택합니다.

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

10. 설치 폴더의 위치를 확인합니다. 설치 폴더를 확인하려면 **확인**을 입력하고, 설치 폴더를 변경하려면 **취소**를 입력합니다.

11. **Enter** 키를 누릅니다.

링크 위치 옵션 목록이 표시됩니다.

12. 링크 위치 옵션의 번호를 입력합니다.

링크 파일 위치를 묻는 메시지가 표시됩니다.

13. 링크 파일의 절대 경로를 입력하고 **Enter** 키를 누릅니다.

소스 샘플 구성 옵션이 나타납니다.

14. 구성 옵션을 입력하고 **Enter** 키를 누릅니다.

옵션	설명
1	샘플 설치 및 구성
2	샘플 소스를 설치하되 샘플은 구성하지 않음

1을 입력하면 응용 프로그램 서버 옵션 목록이 표시됩니다. **2**를 입력하면 설치 전 요약이 표시됩니다.

15. **1**을 입력한 경우 선택할 응용 프로그램 서버의 번호를 입력하고 **Enter** 키를 누릅니다.

응용 프로그램 서버 정보 프롬프트가 표시됩니다.

16. JBoss 설정을 구성합니다.

- a. 응용 프로그램 서버 설치 디렉터리를 지정한 다음 **Enter** 키를 누릅니다.

JBoss 응용 프로그램 서버 구성 이름에 대한 정보가 표시됩니다.

- b. 구성 이름을 지정합니다. 기본값은 독립 실행형입니다.

- c. **Enter** 키를 누릅니다.

- d. 원격 포트를 지정합니다.

참고: JBoss 포트가 기본 데이터베이스 포트와 충돌하는 경우 포트 중 하나를 변경하여 충돌을 해결합니다.

Hub 서버 정보 프롬프트가 표시됩니다.

17. Hub 서버 설치에 대한 정보를 입력하고 **Enter** 키를 누릅니다.

다음 테이블에는 **Hub** 서버 설치 정보에 대한 프롬프트가 설명되어 있습니다.

프롬프트	설명
서버 이름	Hub 서버를 호스팅하는 서버의 이름.
서버 HTTP 포트	Hub 서버의 포트 번호.
Informatica MDM 관리 암호	MDM Hub에 액세스하는 데 필요한 암호.
MDM Hub 홈 디렉터리	Hub 서버 설치를 위한 디렉터리입니다.

MDM Hub ORS ID 목록이 표시됩니다.

18. MDM 샘플 스키마의 연산 참조 저장소 ID를 입력하고 **Enter** 키를 누릅니다.
 샘플 스키마를 등록하지 않은 경우에는 샘플 스키마에 대한 연산 참조 저장소 ID가 표시되지 않습니다. 샘플 연산 참조 저장소를 등록한 다음 설치를 다시 시작합니다.
 배포 선택 프롬프트가 표시됩니다.
19. `postInstallSetup` 스크립트를 설치의 일부로 실행할지 아니면 나중에 수동으로 실행할지를 선택합니다.
20. **Enter** 키를 누릅니다.
 설치 선택 항목의 요약이 표시됩니다.
21. 설치 전 요약에서 정보를 확인합니다. 정보가 올바르면 **Enter** 키를 눌러 설치를 시작합니다.
 제공한 구성 정보에 따라 리소스 키트가 설치됩니다. 프로세스가 완료되면 설치 완료 정보가 표시됩니다.
22. **Enter** 키를 눌러 설치 프로그램을 종료합니다.

자동 모드에서 리소스 키트 설치

자동 모드에서 사용자 상호 작용 없이 리소스 키트를 설치할 수 있습니다. 다중 설치가 필요한 경우나 시스템 클러스터에 설치해야 하는 경우 자동 설치를 수행하고자 할 수 있습니다. 자동 설치에서는 어떠한 진행률 또는 실패 메시지도 표시하지 않습니다.

리소스 키트에 대한 자동 설치를 실행하기 전에 자동 설치에 대한 속성 파일을 구성해야 합니다. 설치 프로그램은 이 파일을 읽어 설치 옵션을 확인합니다. 자동 설치 프로세스는 잘못된 설정(예: 잘못된 응용 프로그램 서버 경로 또는 포트)을 입력한 경우에도 완료될 수 있습니다. 따라서 속성 파일에 올바른 설정을 입력했는지 확인해야 합니다.

리소스 키트를 설치할 시스템의 하드 디스크에 리소스 키트 설치 파일을 복사합니다. 자동 모드에서 설치하려면 다음 태스크를 완료합니다.

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

속성 파일 구성

Informatica는 설치 프로그램에 필요한 매개 변수가 포함된 샘플 속성 파일을 제공합니다. 샘플 속성 파일을 사용자 지정하여 설치 옵션을 지정할 수 있습니다. 그런 다음 자동 설치를 실행합니다.

자동 설치 프로그램은 속성 파일 설정의 유효성을 검사하지 않습니다. 자동 설치 프로그램을 실행하기 전에 올바른 설정을 지정하고 해당 설정을 확인해야 합니다.

1. 다음 디렉터리에서 `silentInstallResourceKit_sample.properties` 파일을 찾습니다.

UNIX의 경우. `/silent_install/mrmresourcekit`

Windows의 경우. `\silent_install\mrmresourcekit`

파일을 사용자 지정한 후 저장합니다. 파일의 이름을 바꾸고 시스템에서 원하는 곳에 저장할 수 있습니다.

참고: 자동 속성 파일에서 슬래시와 백슬래시는 특수 문자입니다. 파일에 정보를 입력하는 경우(예: 설치 경로를 입력하는 경우) 이러한 각 문자를 두 개씩 입력해야 합니다. 예를 들어 서버 디렉터리 경로를 입력하려면 `\\u1\\infamdm\\hub\\resourcekit`를 입력해야 합니다.

2. `silentInstallResourceKit_sample.properties` 파일의 백업 사본을 생성합니다.
3. 텍스트 편집기를 사용하여 파일을 열고 설치 매개 변수 값을 변경합니다.
4. `silentInstallresourcekit.properties`와 같은 새 이름으로 속성 파일을 저장합니다.

다음 표에는 변경할 수 있는 설치 매개 변수가 설명되어 있습니다.

속성 이름	설명
INSTALLER_UI	설치 모드를 지정합니다. <code>silent</code> 로 설정합니다.
SIP.INSTALL.TYPE	설치 유형을 지정합니다. <code>SIPERIAN_SAMPLE_INSTALL</code> 로 설정합니다.
SIP.INSTALL.SAMPLE.SCHEMA	샘플 스키마를 설치할지 여부를 지정합니다. 다음 값 중 하나를 지정합니다. - 0. 샘플 스키마를 설치하지 않음 - 1. 샘플 스키마 설치
SIP.INSTALL.SAMPLES	샘플 및 유틸리티를 설치할지 여부를 지정합니다. 다음 값 중 하나를 지정합니다. - 0. 샘플 및 유틸리티를 설치하지 않음 - 1. 샘플 및 유틸리티 설치
SIP.INSTALL.SIF.SDK	SIF(서비스 통합 프레임워크) SDK를 설치할지 여부를 지정합니다. 다음 값 중 하나를 지정합니다. - 0. SIF SDK를 설치하지 않음 - 1. SIF SDK 설치
SIP.INSTALL.BPM.SDK	BPM SDK를 설치할지 여부를 지정합니다. 다음 값 중 하나를 지정합니다. - 0. BPM SDK를 설치하지 않음 - 1. SIF SDK 설치
SIP.INSTALL.JASPERSOFT	Jaspersoft 보고 도구를 설치할지 여부를 지정합니다. 다음 값 중 하나를 지정합니다. - 0. BPM SDK를 설치하지 않음 - 1. SIF SDK 설치

속성 이름	설명
SIP.INSTALL.SSANAME3	SSA-NAME3을 설치할지 여부를 지정합니다. 다음 값 중 하나를 지정합니다. - 0. SSA-NAME3를 설치하지 않음 - 1. SSA-NAME3 설치
USER_INSTALL_DIR	리소스 키트를 설치할 디렉터리로, 예를 들면 C:\\ <infamd_install_directory>\hub\resourcekit와 같습니다.
RUN_CONFIGURE_FLAG	샘플을 구성할지 여부를 지정합니다. - 0. 샘플을 구성하지 않음 - 1. 샘플 설치 및 구성 기본값은 1입니다. RUN_CONFIGURE_FLAG 속성을 1로 설정한 경우 RUN_CONFIGURE_SETUP 속성을 주석 처리하거나 0으로 설정합니다. 샘플을 구성하려면 응용 프로그램 서버 및 Hub 서버가 시작되었고, 샘플 스키마가 Hub 콘솔에 등록되어 있는지 확인합니다.
RUN_CONFIGURE_SETUP	소스 샘플만 구성할지 여부를 지정합니다. - 0. 샘플 소스를 설치하지 않음 - 1. 샘플 소스 설치 RUN_CONFIGURE_SETUP 속성을 1로 설정한 경우 RUN_CONFIGURE_FLAG 속성을 주석 처리하거나 0으로 설정합니다. RUN_CONFIGURE_SETUP 속성을 1로 설정하면 이후에 샘플을 구성할 수도, 배포할 수도 없습니다.
SIP.AS.CHOICE	응용 프로그램 서버의 이름입니다. JBoss를 지정합니다. 를 지정합니다.
SIP.AS.HOME	JBoss 설치 디렉터리의 경로입니다.
JBoss.AS.PORT_1	JBoss 포트 번호입니다.
SIP.JBOSS.CONFIGURATION_NAME	JBoss 구성 이름입니다. 기본값은 독립 실행형입니다.
SIP.SERVER.NAME	Hub 서버가 배포된 서버의 이름입니다.
SIP.SERVER.HTTP.PORT	Hub 서버가 수신하는 포트입니다.
SIP.ADMIN.PASSWORD	Hub 서버에 액세스하는 데 사용하는 암호입니다.
HUB_SERVER_HOME	Hub 서버 설치를 위한 디렉터리입니다.
SIP.ORS.ID	MDM Hub 샘플 스키마의 연산 참조 저장소 ID입니다.
RUN_DEPLOYMENT_FLAG	자동 설치의 일부로 postInstallSetup 스크립트를 실행합니다. - 0. postInstallSetup 스크립트를 실행하지 않음 - 1. postInstallSetup 스크립트 실행

자동 설치 프로그램 실행

속성 파일을 구성한 후 자동 설치를 시작할 수 있습니다.

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

3. 다음 명령을 실행합니다.

UNIX의 경우. `./hub_resourcekit_install.bin -f <location_of_silent_properties_file_for_resourcekit>`

Windows의 경우. `.\hub_resourcekit_install.exe -f <location_of_silent_properties_file_for_resourcekit>`

자동 설치 프로그램은 백그라운드에서 실행됩니다. 이 프로세스에는 다소 시간이 걸릴 수 있습니다.

`postinstallSetup.log` 파일을 확인하여 설치에 성공했는지 확인합니다.

로그 파일은 다음 디렉터리에서 사용할 수 있습니다.

UNIX의 경우. `<infadm_install_directory>/logs/postInstall.log`

Windows의 경우. `<infadm_install_directory>\logs\postInstall.log`

제 13 장

MDM Hub 문제 해결

이 장에 포함된 항목:

- [Troubleshooting the Installation Process, 104](#)

Troubleshooting the Installation Process

If the installation fails, use the following information to troubleshoot the failure.

Cannot launch the Hub Console

After installation, the Hub Console fails to launch and generates the following error in the log file:

```
SIP-09131: General Decryption failure and [ERROR] com.delos.util.StringUtil: Unable to decrypt
```

Encrypt and update the MDM Hub Master Database password or Operational Reference Store password.

1. To encrypt a database schema password, run the following command from a command prompt:

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

The results are echoed to the terminal window.

2. To update your Master Database password or Operational Reference Store password, connect as the `cmx_system` user and run the following statement:

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

3. Run the `postInstallSetup` script.

Hub Server EAR file deployment fails with `concurrent.RejectedExecutionException`

The Hub Server EAR file deployment might fail with the exception `concurrent.RejectedExecutionException`.

To resolve the issue, increase the length of the task queue in the `standalone-full.xml` file.

The following sample shows the `queue-length` count parameter to modify:

```
<long-running-threads>  
  <core-threads count="50"/>  
  <queue-length count="300"/>  
  <max-threads count="50"/>  
  <keepalive-time time="10" unit="seconds"/>  
</long-running-threads>
```


Cannot start JBoss or launch the Hub Console after installation

After installation, JBoss might not restart after you stop JBoss. When you launch the Hub Console, the Hub Console fails to launch and generates the following error:

```
SIP-09070: Failed to initialize repository layer.
```

To resolve the issue, run the following code in a batch file to restart JBoss:

```
rmdir 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
```

MDM Hub users cannot login

If you re-create the CMX_SYSTEM schema after installation of the Hub Server, the MDM Hub cannot recognize the hashed passwords. As a result, users cannot log in to the MDM Hub.

To resolve the issue, run the `postInstallSetup` script again manually. This script ensures that the passwords of the MDM Hub users are hashed again and users can log in.

Alternatively, if you do not want to run the `postInstallSetup` script again, run the following commands to migrate the user passwords to hashed passwords and create application users.

On UNIX.

```
cd <MDM Hub 설치 디렉터리>/server/bin
./sip_ant.sh hash_users_passwords
./sip_ant.sh add_application_users
```

On Windows.

```
cd <MDM Hub 설치 디렉터리>/server/bin
sip_ant.bat hash_users_passwords
sip_ant.bat add_application_users
```

PostInstallSetUp script fails because the ActiveVOS server deployment times out

When you install the Hub Server, the post-installation setup process might fail after trying to deploy the ActiveVOS server.

To resolve the issue, increase the value of the `deploy.wait.time` property in the `build.properties` file in the following directory:

On UNIX. `<infamdm installation directory>/hub/server/bin`

On Windows. `<infamdm installation directory>\hub\server\bin`

Inaccurate start or stop timestamp for Operational Reference Store

When you export an Operational Reference Store that you create, the MDM Hub creates a `.dmp` file. You can reuse the `.dmp` file of the Operational Reference Store to create a second Operational Reference Store. The start date and end date columns in the `C_REPOS_DB_VERSION` table of the Operational Reference Store will contain incorrect start date and end date. You can ignore this error as it does not affect any functionality.

Operational Reference Store registration fails because the connection times out

When you register the Operational Reference Store (ORS) in a JBoss cluster node, the ORS registration fails with a connection timeout error similar to the following:

```
SIP-10313: SIP-09061: Failed connecting with provider url abc03.
java.net.ConnectException: JBAS012144: Could not connect to remote://abc03:9999. The connection timed out.
```

To resolve the issue, when you register an ORS, disable the JBoss administrative security, and then enable the JBoss administrative security after ORS registration.

Hub Server deployment fails because JBoss composite operation fails

When you deploy the Hub Server in the JBoss application server, the Hub Server deployment fails with the following error:

```
JBAS014653: Composite operation failed and was rolled back. Steps that failed:" => {"Operation step-2" =>
{
    "JBAS014671: Failed services" => {"jboss.web.deployment.default-host./cmx" =>
"org.jboss.msc.service.StartException in service jboss.web.deployment.default-host./cmx:
    org.jboss.msc.service.StartException in anonymous service: JBAS018040: Failed to start context
    Caused by: org.jboss.msc.service.StartException in anonymous service: JBAS018040: Failed to start context"
}
```

To resolve the issue, perform the following steps:

1. Stop the JBoss application server.
2. Use a text editor to open the standalone.xml file in the following directory:
On UNIX. <JBoss Installation Directory>/standalone/configuration
On Windows. <JBoss Installation Directory>\standalone\configuration

3. Find the following code:

```
<subsystem xmlns="urn:jboss:domain:deployment-scanner:1.1">
    <deployment-scanner path="deployments" relative-to="jboss.server.base.dir" scan-
interval="5000"/>
</subsystem>
```

4. Add the deployment-timeout="300" attribute as shown in the following sample code:

```
<subsystem xmlns="urn:jboss:domain:deployment-scanner:1.1">
    <deployment-scanner path="deployments" relative-to="jboss.server.base.dir" scan-
interval="5000" deployment-timeout="300"/>
</subsystem>
```

5. Save changes, and start the JBoss application server.
The default deployment timeout value is 60, which is changed to 300.
6. Deploy the Hub Server.

The Hub Server cannot connect to the cmx_system schema

To verify that the Hub Server cannot connect to the cmx_system schema, review the application server log.

For example, you might find the following error, which confirms that the database is not accessible:

```
Caused by: java.sql.SQLException: ORA-28000: the account is locked
DSRA0010E: SQL State = 99999, Error Code = 28,000
```

To resolve the issue, resolve the database connection issue. Use the application server console to test the connection to the database. If you cannot resolve the connection to the cmx_system schema, re-create the cmx_system schema.

Failed to verify the need to tokenize records

When you run the match process, you might receive the following error:

```
SIP-16062: Failed to verify the need to tokenize records.
```

Verify the following environment variable settings:

- The library path environment variable must contain the following path:

On UNIX. <infamdm_install_directory>/hub/cleanse/lib

On Windows. <infamdm_install_directory>\hub\cleanse\lib

The library path environment variable depends on the operating system:

- AIX. LIBPATH

- Suse or RedHat Linux. LD_LIBRARY_PATH
- Windows. PATH
- The SSAPR environment variable must include the following path for all users:
On UNIX. <infadm_install_directory>/server_install_dir/cleanse/resources
On Windows. <infadm_install_directory>\server_install_dir\cleanse\resources

major.minor version errors when loading the Process Server

If you see multiple major.minor console errors when you try to load the 처리 서버, check that the system has the correct Java version installed.

Java XA Exception

When you restart JBoss after you install the Hub Server, the following exception error might appears:

```
javax.transaction.xa.XAException
```

Multidomain MDM users from cmx_system or cmx_ors do not have all the required XA database privileges. This happens because you might not have access to the system database user, to set up the Multidomain MDM schema.

Ensure that all the user accounts in cmx_system and cmx_ors are created properly and have all the necessary database privileges.

If the users are not created properly and do not have all the database privileges, the following error appears:

```
WARN [com.arjuna.ats.jta] (Periodic Recovery) ARJUNA016027: Local XARecoveryModule.xaRecovery got XA
exception XAException.XAER_RMERR: javax.transaction.xa.XAException
at oracle.jdbc.xa.OracleXAResource.recover(OracleXAResource.java:730)
at org.jboss.jca.adapters.jdbc.xa.XAManagedConnection.recover(XAManagedConnection.java:362)
at org.jboss.jca.core.tx.jbossts.XAResourceWrapperImpl.recover(XAResourceWrapperImpl.java:185)
at
com.arjuna.ats.internal.jta.recovery.arjunacore.XARecoveryModule.xaRecoveryFirstPass(XARecoveryModule.java
:597)
at
com.arjuna.ats.internal.jta.recovery.arjunacore.XARecoveryModule.periodicWorkFirstPass(XARecoveryModule.ja
va:194)
at
com.arjuna.ats.internal.jta.recovery.arjunacore.XARecoveryModule.periodicWorkFirstPass(XARecoveryModule.ja
va:140)
at com.arjuna.ats.internal.arjuna.recovery.PeriodicRecovery.doWorkInternal(PeriodicRecovery.java:765)
at com.arjuna.ats.internal.arjuna.recovery.PeriodicRecovery.run(PeriodicRecovery.java:377)
```

Watch for the following errors when running the create_system or create_ors sip_ant targets:

```
[exec] GRANT SELECT ON SYS.PENDING_TRANS$ TO cmx_system
[exec] DBA user should have grantable privilege for PENDING_TRANS$
[exec] ERROR CODE: -1031
[exec] DECLARE
[exec] *
[exec] ERROR at line 1:
[exec] ORA-01031: insufficient privileges
[exec] ORA-06512: at line 24
[exec] ORA-06512: at line 27
```

Assign the following database privileges to the system user, so that this system database user can run the create_system and create_ors ant targets:

```
GRANT SELECT ON sys.pending_trans$ TO <DBA user> with grant option;
GRANT SELECT ON sys.dbp_pending_transactions TO <DBA user> with grant option;
GRANT SELECT ON sys.dbp_pending TO <DBA user> with grant option;
GRANT EXECUTE ON sys.dbms_xa TO <DBA user> with grant option;
```

Alternatively, to avoid the XA exception error in the log, you can assign the select and execute permissions to the cmx_system and cmx_ors users:

```
GRANT EXECUTE ON sys.dbms_lock TO <DBA user> with grant option;
```

Informatica 주소 확인 exception during certification

Informatica 주소 확인 generates an exception during certification. Ensure that the stack size for the JVM is sufficient.

1. Navigate to the bin directory under the JBoss root directory.
2. Open the following file in a text editor.
On UNIX. run.sh
On Windows. run.bat

3. Set the JAVA_OPTS variable as follows:

```
JAVA_OPTS=${JAVA_OPTS} -Xss2000k
```

4. Save and close the file.

Operational Reference Store does not have a workflow engine configured

If you install the MDM Hub and then import an 연산 참조 저장소 (ORS) from a previous version, you see a fatal error that indicates that the ORS does not have a workflow engine configured. This error occurs because the Siperian BPM workflow engine is not registered by default. Use the Workflow Manager to register the Siperian BPM workflow engine with the name that the ORS expects to find.

After you deploy the Process Server .ear files, an error occurs

On a Linux environment with WebSphere 8.5.5.9, after you deploy the 처리 서버, the following error occurs:

```
Too many open files. Unable to start cleanse ear.
```

Increase the value of the ulimit parameter in Linux, and deploy the 처리 서버.

Cannot add the application users using the sip_ant script

The following applications are affected: Business Process Manager (formerly known as ActiveVOS), Data Director, MDM Hub Console and Provisioning Tool.

To add the application users, run the sip_ant command, using the add_app_users parameter.

You can fetch the connect URL from the following location:

```
hub/server/bin/build.properties (masterdatabase.jdbc.url))
```

Run the following command:

- On Windows.

```
sip_ant.bat add_app_users -Ddatabase.password=<cmx system password> -Dmaster.  
connecturl="jdbc:oracle:thin:@<Database Host name>:<DB Port>:<SID>" -Dmaster.username=cmx_system
```

- On Unix.

```
./sip_ant.sh add_app_users -Ddatabase.password=<cmx system password> -Dmaster.  
connecturl="jdbc:oracle:thin:@<Database Host name>:<DB Port>:<SID>" -Dmaster.username=cmx_system
```

Run the following command on the SQL Server:

```
./sip_ant.sh add_app_users -Ddatabase.password=<cmx system password> -  
Dmaster.connecturl="jdbc:sqlserver://<Database Host Name>:<DB Port>;DatabaseName=cmx_system" -  
Dmaster.username=cmx_system
```

The Certificates folder is not created

The Certificates folder is not created as expected, after you back up the existing Certificates folder from `\infamdm\hub\server\resources\`, and run the `sip_ant.sh hash_users_passwords` script. This issue occurs when the `sip_ant.sh hash_users_passwords` script updates the `C_REPOS_USER` table in `cmx_system`.

To resolve this issue, restart the application server. As a result, the Certificates folder is created in the expected location: `\infamdm\hub\server\resources\`.

The folder will contain only the `MDM_KEYSTORE_FILE_JKS` keystore.

To generate other certificates, run the following command:

```
./sip_ant.sh add_app_users
```

제 14 장

제거

이 장에 포함된 항목:

- [제거 개요, 110](#)
- [Hub 저장소 제거, 110](#)
- [그래픽 모드에서 처리 서버 제거, 111](#)
- [그래픽 모드에서 Hub 서버 제거, 112](#)
- [그래픽 모드에서 리소스 키트 제거, 112](#)
- [콘솔 모드에서 처리 서버 제거, 113](#)
- [콘솔 모드에서 Hub 서버 제거, 113](#)
- [콘솔 모드에서 리소스 키트 제거, 114](#)
- [수동으로 처리 서버 배포 해제, 114](#)
- [수동으로 Hub 서버 배포 해제, 114](#)

제거 개요

MDM Hub를 제거하려면 MDM Hub 구현에서 처리 서버, Hub 서버 및 Hub 저장소를 제거해야 합니다.

다음 단계를 사용하여 MDM Hub를 제거합니다.

1. Hub 저장소를 제거합니다.
2. 처리 서버를 제거합니다.
3. Hub 서버를 제거합니다.

Hub 저장소 제거

Hub 저장소 스키마를 삭제하고 Hub 저장소 스키마에 대한 사용자 로그인을 제거하여 Hub 저장소를 제거할 수 있습니다. Hub 저장소 스키마를 삭제하기 전에 Hub 콘솔을 사용하여 Hub 저장소 스키마의 등록을 해제합니다.

Hub 저장소 스키마를 삭제하려면 관리자 권한이 있어야 합니다.

1. Hub 콘솔을 시작합니다.
2. 구성 작업 영역에서 **데이터베이스** 도구를 클릭합니다.
데이터베이스 정보 페이지가 표시됩니다.

3. 쓰기 잠금 > 잠금 획득을 클릭합니다.
4. 데이터베이스 목록에서 등록을 해제할 연산 참조 저장소를 선택합니다.
5. 데이터베이스 등록 해제 단추를 클릭합니다.
데이터베이스 도구에서 연산 참조 저장소의 등록 해제를 확인하라는 메시지가 표시됩니다.
6. 예를 클릭합니다.
7. SQL*Plus를 사용하여 Oracle 인스턴스에 연결합니다.
8. 각 Hub 저장소 스키마에 대해 다음 명령을 사용하여 스키마를 삭제합니다.

```
drop user <user name> cascade;
```


계단식 배열 옵션을 사용하여 스키마를 삭제하는 경우 연결된 스키마도 삭제됩니다.

그래픽 모드에서 처리 서버 제거

그래픽 모드에서 처리 서버를 제거할 수 있습니다.

UNIX의 그래픽 모드에서 처리 서버 제거

MDM Hub를 제거하려면 처리 서버를 제거해야 합니다. MDM Hub 구현의 각 처리 서버에 대해 처리 서버를 제거하는 단계를 수행해야 합니다.

1. 응용 프로그램 서버를 중지합니다.
2. 다음 디렉터리로 이동합니다.

```
<infamdm_install_directory>/hub/cleanse/UninstallerData
```
3. 제거 프로그램을 실행합니다.

```
./"Uninstall Informatica MDM Hub Cleanse Match Server"
```
4. 제거를 클릭합니다.
제거 프로세스가 완료되면 제거 완료 창이 표시됩니다.
5. 완료를 클릭합니다.

Windows의 그래픽 모드에서 처리 서버 제거

MDM Hub를 제거하려면 처리 서버를 제거해야 합니다. MDM Hub 구현의 각 처리 서버에 대해 처리 서버를 제거하는 단계를 수행해야 합니다.

1. 응용 프로그램 서버를 중지합니다.
2. 시작을 클릭한 다음 프로그램 > Infamdm > Hub > 정리 > 제거 프로그램 데이터 > Informatica MDM Hub 정리 일치 서버 제거를 클릭합니다.
제거 소개 창이 표시됩니다.
3. 제거를 클릭합니다.
제거 프로세스가 완료되면 제거 완료 창이 표시됩니다.
4. 완료를 클릭합니다.

그래픽 모드에서 Hub 서버 제거

그래픽 모드에서 Hub 서버를 제거할 수 있습니다.

UNIX의 그래픽 모드에서 Hub 서버 제거

MDM Hub를 제거하려면 MDM Hub 구현에서 Hub 서버를 제거해야 합니다.

1. 응용 프로그램 서버를 중지해야 합니다.
2. 다음 디렉터리로 이동합니다.
`<infamdm_install_directory>/hub/server/UninstallerData`
3. 제거 프로그램을 실행합니다.
`./"Uninstall Informatica MDM Hub Server"`
제거 소개 창이 표시됩니다.
4. **제거**를 클릭합니다.
제거 프로세스가 완료되면 제거 완료 창이 표시됩니다.
5. **완료**를 클릭합니다.

Windows의 그래픽 모드에서 Hub 서버 제거

MDM Hub를 제거하려면 MDM Hub 구현에서 Hub 서버를 제거해야 합니다.

1. 응용 프로그램 서버를 중지해야 합니다.
2. 시작을 클릭한 다음 **프로그램 > Infamdm > Hub > 서버 > UninstallerData > Informatica MDM Hub 서버 제거**를 클릭합니다.
제거 소개 창이 표시됩니다.
3. **제거**를 클릭합니다.
제거 프로세스가 완료되면 제거 완료 창이 표시됩니다.
4. **완료**를 클릭합니다.

그래픽 모드에서 리소스 키트 제거

그래픽 모드에서 리소스 키트를 제거할 수 있습니다.

UNIX의 그래픽 모드에서 리소스 키트 제거

리소스 키트를 제거하려면 MDM Hub 구현에서 리소스 키트를 제거해야 합니다.

1. 응용 프로그램 서버를 중지합니다.
2. 다음 디렉터리로 이동합니다.
`<infamdm_install_directory>/hub/resourcekit/UninstallerData`
3. 다음 명령을 실행합니다.
`./"Uninstall Informatica MDM Hub Resource Kit"`
Informatica MDM Hub 리소스 키트 제거 창이 나타납니다.

4. **제거**를 클릭합니다.
제거할 수 없는 항목 목록과 함께 **제거 완료** 창이 나타납니다.
5. **완료**를 클릭합니다.
6. 다음 디렉터리를 수동으로 제거합니다.
`<infamdm_install_dir>/hub/resourcekit`

Windows의 그래픽 모드에서 리소스 키트 제거

리소스 키트를 제거하려면 MDM Hub 구현에서 리소스 키트를 제거해야 합니다.

1. 응용 프로그램 서버를 중지합니다.
2. 다음 디렉터리로 이동합니다.
`<ResourceKit_install_dir>\deploy\UninstallerData`
3. Uninstall Informatica MDM Hub Resource Kit.exe를 두 번 클릭합니다.
Informatica MDM Hub 리소스 키트 제거 창이 나타납니다.
4. **제거**를 클릭합니다.
제거할 수 없는 항목 목록과 함께 **제거 완료** 창이 나타납니다.
5. **완료**를 클릭합니다.
6. 다음 디렉터리를 수동으로 제거합니다.
`<infamdm_install_dir>\hub\resourcekit`

콘솔 모드에서 처리 서버 제거

UNIX에서 콘솔 모드로 처리 서버를 제거할 수 있습니다. 콘솔 모드에서 처리 서버를 설치한 경우 콘솔 모드에서 처리 서버를 제거하십시오.

1. 다음 디렉터리로 이동합니다.
`<infamdm_install_dir>/hub/cleanse/UninstallerData`
2. 다음 명령을 입력하여 제거 프로그램을 실행합니다.
`./"Uninstall Informatica MDM Hub Cleanse Match Server"`

콘솔 모드에서 Hub 서버 제거

UNIX에서 콘솔 모드로 Hub 서버를 제거할 수 있습니다. 콘솔 모드에서 Hub 서버를 설치한 경우 콘솔 모드에서 Hub 서버를 제거하십시오.

1. 다음 디렉터리로 이동합니다.
`<infamdm_install_dir>/hub/server/UninstallerData`
2. 다음 명령을 입력하여 제거 프로그램을 실행합니다.
`./"Uninstall Informatica MDM Hub Server"`

콘솔 모드에서 리소스 키트 제거

콘솔 모드에서 리소스 키트를 제거할 수 있습니다. 콘솔 모드에서 리소스 키트를 설치한 경우 콘솔 모드에서 리소스 키트를 제거합니다.

1. 다음 디렉터리로 이동합니다.

UNIX의 경우. `<infadm_install_dir>/hub/resourcekit/UninstallerData`

Windows의 경우. `<infadm_install_dir>\hub\resourcekit\UninstallerData`

2. 명령 프롬프트에서 다음 명령을 실행합니다.

UNIX의 경우. `"Uninstall Informatica MDM Hub Resource Kit.bin" -i console`

Windows의 경우. `"Uninstall Informatica MDM Hub Resource Kit.exe" -i console`

수동으로 처리 서버 배포 해제

JBoss 응용 프로그램 서버에서 처리 서버를 수동으로 배포 해제해야 할 수 있습니다.

1. 처리 서버를 수동으로 배포 해제하려면 다음 디렉터리에서 `siperian-mrm-cleanse.ear` 파일을 삭제합니다.

UNIX의 경우. `<JBoss_install_dir>/standalone/deployments`

Windows의 경우. `<JBoss_install_dir>\standalone\deployments`

2. JBoss 자동 배포 프로세스에서 응용 프로그램을 제거하지 않으면 JBoss 응용 프로그램 서버를 다시 시작합니다.

자세한 내용은 JBoss 설명서를 참조하십시오.

수동으로 Hub 서버 배포 해제

JBoss 응용 프로그램 서버에서 Hub 서버 수동으로 배포 해제해야 할 수 있습니다.

1. JBoss 배포 디렉터리에서 다음 배포 파일을 제거합니다.

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

배포 파일은 다음 디렉터리에 있습니다.

UNIX의 경우.

- `<JBoss 설치 디렉터리>/<구성 이름>/deployments`

Windows의 경우.

- `<JBoss 설치 디렉터리>\<구성 이름>\deployments`

2. JBoss 자동 배포 프로세스에서 응용 프로그램을 제거하지 않으면 JBoss 응용 프로그램 서버를 다시 시작합니다.

자세한 내용은 JBoss 설명서를 참조하십시오.

인덱스

A

ActiveVOS
URN, 설정 [87](#)
ActiveVOS 콘솔 관리자
abAdmin 역할 [25](#)
생성 [25](#)

H

HTTPS
처리 서버용 [81](#)
Hub 서버
JBoss 클러스터에 배포 [45](#)
postInstallSetup 스크립트 [52](#)
마법사로 설치 [43](#)
명령줄에서 설치 [44](#)
배포 [51](#)
배포 스크립트 [51](#)
버전 정보 [48](#)
빌드 번호 [48](#)
설치 [96](#)
설치 로그 파일 [48](#)
수동 배포 [51](#), [53](#), [57](#)
수동으로 재배포 [57](#)
자동 설치 [100](#)
자동으로 설치 [45](#)
제거 [113](#), [114](#)
Hub 서버 속성 파일
구성 [27](#)
Hub 저장소
제거 [110](#)
테이블스페이스, 생성 [15](#)
Hub 콘솔
시작 [58](#), [62](#)
Hub 콘솔 클라이언트
build.properties
구성 [49](#)
구성 [49](#)
다중 노드 환경 [49](#)
클러스터 환경 [49](#)

I

Infinispan
구성 [61](#), [62](#)
Informatica ActiveVOS
스키마 생성 [18](#)

J

JBoss
JBoss 클러스터 노드에서 시작 [26](#)
독립 실행형 JBoss 인스턴스에서 시작 [26](#)
설정 [50](#)
JBoss 구성
MDM Hub용 [20](#)
JBoss 보안
활성화 [67](#)
JBoss 클러스터
Hub 서버 배포 [45](#)
처리 서버 배포 [76](#)
JBoss 클러스터 노드에서 시작
독립 실행형 인스턴스에서 시작 [26](#)
JMS 메시지 대기열
구성 [55](#), [58](#)

M

Master Database
creating [28](#)
MDM Hub
JDK(Java 개발 키트) 요구 사항 [13](#)
구성 요소 [9](#)
설치 디자인 [11](#)
설치 태스크 [11](#)
설치 토폴로지 [11](#)
소개 [9](#)
시스템 요구 사항 [13](#)
운영 체제 로컬 설정 [13](#)
환경 변수 설정 [13](#)

N

대상 데이터베이스
선택 [62](#)
데이터베이스
대상 데이터베이스 [62](#)
연결 테스트 [15](#)
로그 파일
Hub 서버 로그 파일 [48](#)
JBoss 로그 파일 [48](#), [78](#)
디버그 로그 파일 [48](#), [78](#)
설치 로그 파일 [48](#), [78](#)
설치 선행 조건 로그 파일 [48](#), [78](#)
설치 후 설정 로그 파일 [48](#), [78](#)
처리 서버 로그 파일 [78](#)
리소스 키트
설치 [96](#), [98](#)
자동 속성 파일 [101](#)
제거 [113](#)

- 마스터 데이터베이스
 - 메타데이터 가져오기 [31](#)
- 보안 통신
 - 활성화, 처리 서버 [81](#)
- 샘플 스키마
 - 등록 [94](#)
 - 설치 [91](#)
- 서버 속성 구성 [23](#)
- 설치
 - Hub 서버
 - 마법사 [43](#)
 - 명령줄 [44](#)
 - 자동으로 [45](#)
 - 리소스 키트 [96, 98](#)
 - 처리 서버
 - 마법사 [74](#)
 - 명령줄 [75](#)
 - 자동으로 [76](#)
- 설치 프로그램
 - 워크플로우 [38, 71](#)
- 연산 참조 저장소
 - 등록 [63](#)
 - 메타데이터 가져오기 [33](#)
 - 생성 [30](#)
- 워크플로우 엔진
 - 추가 [88](#)
- 일치 인구집단
 - 활성화 [82](#)
- 일치 채우기
 - 활성화 [82](#)
- 전체 프로파일 구성 [23](#)
- 제거
 - Hub 서버 [113, 114](#)
 - Hub 저장소 [110](#)
 - 처리 서버 [113](#)
- 처리 서버
 - HTTPS, 활성화 [81](#)
 - JBoss 클러스터에 배포 [76](#)
 - postInstallSetup 스크립트 [80](#)
 - 데이터 소스 생성 [79](#)
 - 마법사로 설치 [74](#)
 - 명령줄에서 설치 [75](#)
 - 배포 [79-81](#)
 - 배포 스크립트 [79, 80](#)
 - 버전 정보 [79](#)
 - 빌드 번호 [79](#)
 - 설치 로그 파일 [78](#)
 - 수동 배포 [79, 80](#)

- 처리 서버 (계속)
 - 자동으로 설치 [76](#)
 - 제거 [113](#)
- 처리 서버 속성 파일
 - 구성 [27](#)
- 테이블스페이스
 - 생성 [15](#)

O

- Oracle 구성
 - init.ora 매개 변수 설정 [15](#)
 - MDM Hub용 [15](#)
 - Oracle 휴지통 비활성화 [15](#)

P

- postinstallsetup 스크립트
 - 실행 [80](#)
- postInstallSetup 스크립트
 - Hub 서버용 [52](#)
 - 실행 [52](#)
 - 처리 서버용 [80](#)

R

- Remoting-Connector 보안
 - 활성화 [67](#)

T

- TLS
 - 구성 [25](#)
- TNS 이름
 - 추가 [15](#)
- troubleshooting
 - post-installation process [104](#)

U

- URN
 - ActiveVOS 설정 [87](#)