



Informatica® Multidomain MDM
10.4

Oracle Database with IBM WebSphere **에 대한 설치 가이드**

Informatica Multidomain MDM Oracle Database with IBM WebSphere 에 대한 설치 가이드

10.4

2020년3월

© 저작권 Informatica LLC 2001, 2020

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: 2020-06-04

목차

서문	8
Informatica 리소스	8
Informatica 네트워크	8
Informatica 기술 자료	8
Informatica 설명서	8
Informatica Product Availability Matrix	9
Informatica Velocity	9
Informatica Marketplace	9
Informatica 글로벌 고객 지원 센터	9
장 1: 설치 개요	10
Multidomain MDM 설치	10
설치 토폴로지	12
설치 태스크	12
장 2: 설치 전 태스크	13
설치 준비	13
환경 준비	14
데이터베이스 환경 설정	16
1단계. Oracle 설치 및 구성	16
2단계. 데이터베이스 및 테이블스페이스 생성	17
3단계. 데이터베이스 권한 및 연결 설정	19
4단계. ActiveVOS 스키마 생성	19
응용 프로그램 서버 환경 설정	20
Java Virtual Machine 구성	21
TLS(Transport Layer Security) 구성	24
보안 HTTP 응답 헤더	24
MDM Hub 환경에서 암호 암호화	24
WebSphere 환경에서 보안 프로필 생성	24
JAAS 응용 프로그램 로그인 구성	25
ActiveVOS 콘솔 관리자 생성	26
MDM Hub 배포를 위한 SOAP 요청 제한 시간 구성	26
추가 응용 프로그램 서버 구성(선택 사항)	27
독립 실행형 처리 서버 인스턴스에 대한 WebSphere 구성	27
여러 MDM Hub 마스터 데이터베이스에 대한 WebSphere 구성	30
HTTPS 프로토콜 구성	30
Informatica Data Director에 대한 WebSphere 구성	31
자동 설치를 위한 속성 파일 구성	31
Informatica Platform 속성 파일 구성	32

Hub 서버 속성 파일 구성.....	32
처리 서버 속성 파일 구성.....	32
장 3: Hub 저장소 설치.....	33
MDM Hub 마스터 데이터베이스 생성.....	33
연산 참조 저장소 생성.....	35
MDM Hub 마스터 데이터베이스로 메타데이터 가져오기.....	36
연산 참조 저장소로 메타데이터 가져오기.....	38
장 4: Hub 저장소 설치 후 태스크.....	40
Oracle 구성 요소에 대한 액세스 확인.....	40
장 5: Hub 서버 설치.....	42
Hub 서버 설치.....	42
설치 프로그램 워크플로우 검토.....	43
설치 값 수집.....	44
설치 마법사에서 Hub 서버 설치.....	49
명령줄에서 Hub 서버 설치(UNIX만 해당).....	49
자동으로 Hub 서버 설치.....	50
클러스터의 노드에 Hub 서버 설치.....	51
장 6: Hub 서버 설치 후 태스크.....	52
설치 로그 파일 복사.....	53
버전 및 빌드 번호 확인.....	53
Elasticsearch 설치 및 구성.....	54
Hub 콘솔 클라이언트 구성(조건부).....	54
MDM Hub 마스터 데이터베이스 이름 구성.....	54
Hub 서버 EAR 파일 재배포.....	55
클래스 로더 구성.....	55
응용 프로그램 서버 설정 확인 및 구성(조건부).....	56
응용 프로그램 서버 설정 편집.....	56
WebSphere 다중 노드 또는 클러스터 환경에 대한 Hub 서버 구성.....	56
Hub 서버 응용 프로그램 배포(조건부).....	57
스크립트를 사용하여 Hub 서버 응용 프로그램 배포(조건부).....	58
수동으로 Hub 서버 응용 프로그램 배포(조건부).....	59
1단계. 데이터 소스 생성.....	59
2단계. JMS 메시지 대기열 구성.....	63
3단계. Hub 서버 EAR 파일 다시 패키지.....	65
4단계. Hub 서버 응용 프로그램 배포.....	66
5단계. 클래스 로더 구성.....	67
6단계. Hub 서버에 JMS 메시지 대기열 구성.....	67
7단계. Informatica Data Director의 서버 리소스 구성.....	69

메타데이터 캐시 구성(선택 사항).....	69
Infinispan 특성 편집.....	70
Hub 콘솔 시작.....	71
연산 참조 저장소 등록.....	72
장 7: 처리 서버 설치.....	74
처리 서버 설치.....	74
설치 프로그램 워크플로우 검토.....	75
설치 값 수집.....	76
설치 마법사에서 처리 서버 설치.....	78
명령줄에서 처리 서버 설치(UNIX만 해당).....	79
자동으로 처리 서버 설치.....	80
클러스터의 노드에 처리 서버 설치.....	80
장 8: 처리 서버 설치 후 태스크.....	82
설치 로그 파일 복사.....	82
버전 및 빌드 번호 확인.....	83
WebSphere 다중 노드 또는 클러스터 환경에 대한 처리 서버 구성.....	83
처리 서버 EAR 파일 재배포.....	84
클래스 로더 구성.....	84
처리 서버 응용 프로그램 배포(조건부).....	85
1단계. 데이터 소스 생성(조건부).....	85
2단계. 처리 서버 응용 프로그램 배포(조건부).....	88
3단계. 클래스 로더 구성.....	90
처리 서버에 대한 보안 통신 활성화.....	90
Elasticsearch 설치 및 구성.....	91
일치 인구집단 구성.....	92
일치 인구집단 활성화.....	92
처리 서버와 정리 엔진 구성.....	93
장 9: 응용 프로그램 서버에 대한 ActiveVOS 설치 후 태스크.....	94
WebSphere 클러스터 환경에서 ActiveVOS 설치 및 배포.....	94
WebSphere 작업 관리자 구성.....	94
WebSphere 시간 관리자 구성.....	95
JAAS 응용 프로그램 로그인 구성.....	95
ActiveVOS 서버 및 ActiveVOS Central 설치 완료.....	96
ActiveVOS 설치 파일 편집.....	96
ActiveVOS 및 Identity Resolution 배포.....	97
WebSphere 환경에서 트러스트된 사용자 생성.....	97
사용자 및 그룹을 보안 프로필에 추가.....	98

장 10: 비즈니스 항목 어댑터에 대한 ActiveVOS 설치 후 태스크.....	99
ActiveVOS 웹 응용 프로그램.....	99
비즈니스 항목 워크플로우 어댑터의 ActiveVOS URN 구성.....	100
ActiveVOS URL의 프로토콜 구성.....	100
ActiveVOS 프로토콜을 HTTPS로 설정.....	101
기본 워크플로우 엔진 구성.....	101
ActiveVOS의 MDM ID 서비스 구성.....	102
태스크 구성.....	103
 장 11: 리소스 키트 설치.....	 104
MDM Hub 샘플 연산 참조 저장소 설정.....	104
Informatica MDM Hub 샘플 연산 참조 저장소 등록.....	107
그래픽 모드에서 리소스 키트 설치.....	109
콘솔 모드에서 리소스 키트 설치.....	111
자동 모드에서 리소스 키트 설치.....	113
속성 파일 구성.....	114
자동 설치 프로그램 실행.....	116
 장 12: 리소스 키트 설치 후 태스크.....	 117
sip_ant 스크립트 편집.....	117
수동으로 postInstall 스크립트 실행.....	117
 장 13: MDM Hub 문제 해결.....	 119
설치 프로세스 문제 해결.....	119
 장 14: 제거.....	 123
제거 개요.....	123
Hub 저장소 제거.....	123
그래픽 모드에서 처리 서버 제거.....	124
UNIX의 그래픽 모드에서 처리 서버 제거.....	124
Windows의 그래픽 모드에서 처리 서버 제거.....	124
그래픽 모드에서 Hub 서버 제거.....	125
UNIX의 그래픽 모드에서 Hub 서버 제거.....	125
Windows의 그래픽 모드에서 Hub 서버 제거.....	125
그래픽 모드에서 리소스 키트 제거.....	125
UNIX의 그래픽 모드에서 리소스 키트 제거.....	125
Windows의 그래픽 모드에서 리소스 키트 제거.....	126
콘솔 모드에서 처리 서버 제거.....	126
콘솔 모드에서 Hub 서버 제거.....	126
콘솔 모드에서 리소스 키트 제거.....	127
수동으로 처리 서버 배포 해제.....	127

수동으로 Hub 서버 배포 해제.....	127
인덱스.....	128

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, 10](#)
- [설치 토폴로지, 12](#)
- [설치 태스크, 12](#)

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 거치도록 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.
Informatica Platform	소스 데이터를 정리하고 MDM Hub로 전송할 때 사용하는 Informatica 서비스 및 Informatica 클라이언트로 구성되는 환경입니다. MDM Hub에서 제공되는 정리 기능 대신 Informatica Platform 기능을 사용하여 데이터를 정리할 수 있습니다. Hub 서버 설치의 일부로 Informatica 플랫폼을 설치하면 데이터 통합 서비스, 모델 리포지토리 서비스 및 Informatica Developer(Developer tool)가 설치됩니다.
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, 13](#)
- [환경 준비, 14](#)
- [데이터베이스 환경 설정, 16](#)
- [응용 프로그램 서버 환경 설정, 20](#)
- [추가 응용 프로그램 서버 구성\(선택 사항\), 27](#)
- [자동 설치를 위한 속성 파일 구성, 31](#)

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.

태스크	설명
설치 프로그램 파일 다운로드 및 추출	<p>Informatica Electronic Software Download 사이트에서 설치 파일을 시스템의 디렉터리에 다운로드합니다. 압축된 파일을 추출하려면 빈 폴더도 추출하는 추출 유틸리티를 사용합니다.</p> <p>다음 설치 파일을 다운로드 및 추출합니다.</p> <ul style="list-style-type: none"> - 운영 체제에 대한 MDM Hub 설치 프로그램 - 데이터베이스 파일 - 운영 체제에 대한 ActiveVOS Server 설치 프로그램 <p>Informatica 플랫폼을 설치하려는 경우 다음 파일을 다운로드합니다.</p> <ul style="list-style-type: none"> - 운영 체제에 대한 Informatica 플랫폼 서버 설치 파일 - Informatica 플랫폼 클라이언트 설치 파일
라이선스 키 확인	<p>Informatica의 전자 메일 메시지에 있는 라이선스 키가 있는지 확인합니다. 제품을 설치하는 사용자 계정에서 액세스할 수 있는 디렉터리에 라이선스 키 파일을 복사합니다.</p> <p>라이선스 키가 없으면 Informatica 글로벌 고객 지원 센터에 문의하십시오.</p>
설치 기록 디렉터리 생성	<p>유효성 검사 결과, 환경 보고서, 데이터베이스 디버그 로그 및 로그 파일과 같은 설치 파일의 사본을 저장하기 위한 디렉터리를 생성합니다.</p> <p>예를 들어 install_doc 디렉터리를 생성합니다. 설치 문제를 해결하려면 분석을 위해 설치 기록 디렉터리의 보관 파일을 생성하여 Informatica 글로벌 고객 지원 센터로 보낼 수 있습니다.</p>

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

태스크	설명
Visual Studio 2015용 Visual C++ 재배포 가능 패키지를 설치 Windows만 해당	Windows 시스템에서 Multidomain MDM를 사용하려면 Visual Studio 2015용 Visual C++ 재배포 가능 패키지를 통해 이름 검색 기능 및 일치 기능을 지원해야 합니다.
환경 변수 설정	<p>MDM Hub 설치를 위한 환경 변수를 설정합니다.</p> <p>올바른 JDK를 사용하려면 JDK 디렉터리를 가리키도록 다음과 같은 환경 변수를 설정합니다.</p> <ul style="list-style-type: none"> - JAVA_HOME. 필수 - PATH. 필수 - INFA_JDK_HOME. 선택 사항입니다. AIX에 Informatica Platform을 설치하려는 경우에 필요합니다. <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> <p>도메인에 대해 보안이 활성화된 Informatica 플랫폼을 설치하려는 경우 다음과 같은 디렉터리를 가리키도록 INFA_TRUSTSTORE 환경 변수를 설정합니다.</p> <p><Informatica platform installation directory>/Client/clients/shared/security</p>
운영 체제 로컬 설정	Hub 서버, MDM Hub 마스터 데이터베이스, 연산 참조 저장소 및 Hub 콘솔에 대한 동일한 운영 체제 로컬을 설정합니다.
UNIX에서 X Window System 설치	UNIX에서 그래픽 모드로 설치 프로그램을 실행할 경우 X Window System을 설치합니다. X Window System은 그래픽 디스플레이 서버입니다. X Window System 설치에 대한 자세한 내용은 운영 체제용 설명서를 참조하십시오.
응용 프로그램 서버의 루트 시작 페이지에 대한 액세스 비활성화	보안을 개선하려면 응용 프로그램 서버의 루트 시작 페이지에 대한 액세스를 비활성화합니다. 지침은 응용 프로그램 서버에 대한 설명서를 참조하십시오.
보안되지 않은 TLS 암호 그룹 비활성화	<p>보안을 개선하려면 Multidomain MDM을 사용하는 Java Runtime Environment에서 TLS 암호 그룹을 비활성화합니다.</p> <ol style="list-style-type: none"> 1. 다음 파일을 엽니다. ../jdk<버전>/jre/lib/security/java.security 2. jdk.tls.disabledAlgorithms 속성을 찾고 다음의 보안되지 않은 암호 그룹 목록을 포함하도록 값을 업데이트합니다. <p>jdk.tls.disabledAlgorithms = SSLv3, RC4, MD5withRSA, DH keySize < 1024, EC keySize < 224, DES40_CBC, RC4_40, 3DES_EDE_CBC, EDH-RSA-DES-CBC3-SHA, ECDHE-RSA-DES-CBC3-SHA, DES-CBC3-SHA</p> <p>속성에 대한 자세한 내용은 JDK 설명서를 참조하십시오.</p>

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. <ul style="list-style-type: none"> - 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 로드에는 데이터베이스 관리를 간소화하려면 큰 파일 테이블스페이스를 생성합니다. 그러나 필요한 경우 작은 파일 테이블스페이스를 생성할 수 있습니다.

1. Oracle에 SYSTEM과 같은 데이터베이스 관리자 로깅합니다.
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 로드와 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.dba_pending_transactions TO <DBA user> with grant option; GRANT SELECT ON sys.dba_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

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

WebSphere 설치 디렉터리 경로에 공백이 없는지 확인합니다.

참고: 데이터베이스 서버와 동일한 시간대로 응용 프로그램 서버를 설치합니다.

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

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

Java Virtual Machine 구성

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

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

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

다음 테이블에는 Java 옵션 설정이 설명되어 있습니다.

Java 옵션	설명
-server	시작은 느리지만 후속 작업은 빨라집니다.
-Djava.net.preferIPv4Stack	Java가 IPv4(인터넷 프로토콜 버전 4)를 사용할지 여부를 지정합니다. 운영 체제에서 IPv6(인터넷 프로토콜 버전 6)을 사용하는 경우 true로 설정합니다.
-Dcom.ibm.crypto.provider.DoRSATypeChecking	Java에서 개인 키를 사용한 데이터의 RSA 유형 암호화 및 공개 키를 사용한 암호 해독을 허용할지 여부를 지정합니다. MDM Hub 설치 프로그램에서 라이선스 인증서를 읽을 수 있고, MDM Hub에서 암호 해시가 작동하도록 하려면 필요합니다. false로 설정합니다. -Dcom.ibm.crypto.provider.DoRSATypeChecking을 false로 설정하지 않으면 Hub 서버가 시작되지 않고 라이선스 오류가 발생할 수 있습니다.
-Djgroups.bind_addr	JGroup이 메시지를 보내고 받아야 하는 인터페이스입니다. 다중 노트 또는 클러스터된 환경에서 필요합니다. 각 노트가 고유한 네트워크 인터페이스에 바인딩되어야 합니다.

Java 옵션	설명
-De360.mdm.host -De360.mdm.port -De360.connection.channel	<p>응용 프로그램 서버 통신 프로토콜, 호스트 및 포트입니다. 2809 이외의 Bootstrap 포트에 MDM Hub 응용 프로그램을 배포하려면 다음 Java 옵션을 설정합니다.</p> <ul style="list-style-type: none"> -De360.connection.channel. 사용할 통신 프로토콜로 설정합니다. 유효한 값은 HTTP 및 HTTPS입니다. 기본값은 HTTP입니다. -De360.mdm.host. WebSphere 호스트의 IP 주소로 설정합니다. <p>환경에서 HTTPS 통신 프로토콜을 사용하고 보안 인증서가 FQDN(정규화된 도메인 이름)으로 발행되는 경우 FQDN으로 설정합니다.</p> <ul style="list-style-type: none"> -De360.mdm.port. 2809 대신 구성된 WebSphere Bootstrap 포트에 설정합니다. <p>이 매개 변수를 구성하지 않으면 Entity 360 프레임워크에 기반한 Data Director 화면이 예상대로 작동하지 않을 수 있습니다.</p>
-Didd.mdm.host -Didd.mdm.port -Didd.protocol	<p>제목 영역이 있는 Data Director에 필요합니다. 제목 영역이 있는 Data Director를 배포하려면 다음 Java 옵션을 설정합니다.</p> <ul style="list-style-type: none"> -Didd.mdm.host. 응용 프로그램 서버 호스트의 호스트 이름 또는 IP 주소로 설정합니다. -Didd.mdm.port. 기본값이 아닌 HTTP 또는 HTTPS 수신기 포트를 사용하여 배포하는 데 필요합니다. 이 속성은 서버 초기화 중 제목 영역 응용 프로그램을 배포하는 데 사용됩니다. 응용 프로그램용 JVM에 사용되는 HTTP 또는 HTTPS 수신기 포트를 지정합니다. 기본값은 8080입니다. -Didd.protocol. 서버 초기화 중 제목 영역 응용 프로그램을 배포하는 데 사용되는 필수 속성입니다. 사용할 통신 프로토콜을 HTTP 또는 HTTPS로 지정합니다. 기본값은 HTTP입니다.
-DFrameworksLogConfigurationPath	log4j.xml 파일의 경로입니다.
-Doracle.jdbc.J2EE13Compliant	<p>Oracle 드라이버의 시스템 변수가 J2EE와 완전하게 호환될지 여부를 지정합니다. true로 설정합니다.</p> <p>매개 변수를 true로 설정하지 않으면 JDBC(Java Database Connectivity) 문제가 발생할 수 있습니다.</p>
-Dmdm.node.groupid	MDM Hub 구현에서 Java Virtual Machine에 대한 그룹 ID를 지정합니다. Hub 서버 및 처리 서버를 논리적으로 그룹화하려는 경우에만 필요합니다.
-Dfile.encoding -Dclient.encoding.override	<p>Informatica Data Director를 사용하고 REST API를 사용하여 레코드를 검색하려는 경우에 필요합니다.</p> <p>UTF-8 문자가 포함된 레코드를 찾아서 저장하려면 두 Java 옵션을 모두 UTF-8로 설정합니다.</p>
-Dstricttransportsecurity.flag	웹 브라우저에서 HTTP 요청을 사용한 모든 Data Director 액세스 시도를 HTTPS 요청으로 변환할지 여부를 지정합니다. true로 설정합니다.
-XX:codecachetotal	JIT 코드 캐시 크기입니다. MDM Hub 환경의 성능을 개선하려면 512m으로 설정합니다.

Java 옵션	설명
-Xmx	최대 JVM 힙 크기입니다. 6GB 이상으로 설정합니다. 예를 들어 -Xmx를 6144m로 설정하려면 다음과 같은 JAVA_OPTIONS 환경 변수 설정을 사용합니다. set "JAVA_OPTIONS=-server ... -Xmx6144m"
-Xms	초기 힙 크기입니다. 2048m로 설정합니다.
-Xmso	처리 서버 JVM에 필요합니다. 운영 체제 스레드에 대한 초기 스택 크기입니다. 낮은 시스템 스레드 스택 크기로 인해 응용 프로그램 서버가 예기치 않게 종료되는 것을 방지합니다. 4096k로 설정합니다.
-Xss	초기 스택 크기입니다. 2000k로 설정합니다.
XX:+UseCodeCacheFlushing	코드 캐시가 가득 찬 경우 컴파일된 코드의 JVM 처리 여부를 지정합니다.
-Dtask.pageSize=<최대 태스크 수>	각 요청에서 검색되는 최대 ActiveVOS 태스크 수를 지정합니다. 기본값은 5000입니다. 환경에 많은 수의 태스크가 있는 경우 수를 늘리십시오.

Java Virtual Machine 논리적 그룹화의 예

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

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

다음 테이블에는 논리적 JVM 그룹의 예제가 나와 있습니다.

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

JVM1의 경우 시작 스크립트에 다음 Java 옵션을 추가합니다.

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

JVM2의 경우 시작 스크립트에 다음 Java 옵션을 추가합니다.

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

JVM3의 경우 시작 스크립트에 다음 Java 옵션을 추가합니다.

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

JVM4의 경우 시작 스크립트에 다음 Java 옵션을 추가합니다.

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

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

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

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.

MDM Hub 환경에서 암호 암호화

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

1. 다음 디렉터리에서 wsadmin.properties 파일을 엽니다.

```
<WebSphere 설치 디렉터리>/profiles/<응용 프로그램 서버 프로파일 이름>/properties
```

2. com.ibm.ws.scripting.echoparams Java 속성을 false로 설정합니다.

WebSphere 환경에서 보안 프로파일 생성

WebSphere에서 Multidomain MDM 및 Informatica ActiveVOS에 사용할 보안 프로 파일을 구성합니다.

1. 명령줄에서 다음 샘플 코드에 표시된 대로 보안 프로 파일을 생성합니다.

Windows의 경우

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

UNIX의 경우

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

2. WebSphere 콘솔에서 보안 전송 유형을 SSL 지원됨 유형으로 변경합니다.

- a. 보안을 확장하고 글로벌 보안을 클릭합니다.
- b. 인증 아래에서 RMI/IIOP 보안을 확장하고 CSiv2 인바운드 통신을 클릭합니다.
- c. CSiv2 전송 계층 아래의 전송 목록에서 SSL 지원됨을 선택합니다.

- d. **적용**을 클릭한 다음 **저장**을 클릭합니다.
 - e. **CSiv2 아웃바운드 통신**을 클릭합니다.
 - f. CSiv2 전송 계층 아래의 전송 목록에서 **SSL 지원됨**을 선택합니다.
 - g. **적용**을 클릭한 다음 **저장**을 클릭합니다.
3. WebSphere 콘솔에서 응용 프로그램 보안이 설정되었는지 확인합니다.
 - a. **보안**을 확장하고 **글로벌 보안**을 클릭합니다.
 - b. 응용 프로그램 보안 아래에서 **응용 프로그램 보안 활성화**를 선택합니다.
 - c. **적용**을 클릭한 다음 **저장**을 클릭합니다.
4. 페더레이션된 리포지토리를 설정합니다.
 - a. **보안**을 확장하고 **글로벌 보안**을 클릭합니다.
 - b. 사용자 계정 리포지토리 아래의 사용 가능한 영역 정의 목록에서 **페더레이션된 리포지토리**를 선택합니다.
 - c. **구성**을 클릭합니다.
 - d. 영역의 리포지토리 아래에서 **기본 제공 리포지토리 사용**을 클릭합니다.
 - e. 관리자에 대한 암호를 지정합니다.
 - f. **적용**을 클릭한 다음 **저장**을 클릭합니다.
5. WebSphere 프로필을 다시 시작합니다.

JAAS 응용 프로그램 로그인 구성

WebSphere 클러스터 환경에 응용 프로그램 보안을 활성화했다면 JAAS 응용 프로그램 로그인을 구성하십시오. JAAS 응용 프로그램 로그인 설정은 ActiveVOS를 시작하는 데 필요합니다.

1. WebSphere 콘솔을 시작합니다.
2. JAAS 로그인 모듈을 구성합니다.
 - a. **보안(Security)**을 확장하고 **글로벌 보안(Global security)** > **JAAS - 응용 프로그램 로그인(JAAS - Application logins)**을 클릭합니다.
 - b. ActiveVOS에서 제공된 사용자의 응용 프로그램 로그인을 추가합니다.
 - c. 별칭을 ActiveVOSProvidedUser로 지정합니다.
 - d. **JAAS 로그인 모듈(JAAS login modules)** 섹션에서 다음 모듈 클래스를 추가합니다.
com.activee.rt.websphere.trustvalidation.AeBasicLoginModule
 - e. 변경 내용을 저장합니다.
3. **일반 속성(General properties)** 섹션에서 모듈 속성을 설정합니다.
 - a. 인증 전략을 **필수(Required)**로 설정합니다.
 - b. 모듈의 사용자 이름 및 암호를 입력합니다.
 - c. 변경 내용을 저장합니다.
4. ActiveVOS 사용자를 생성합니다.
 - a. **사용자 및 그룹(Users and Groups)**을 확장하고 **사용자 관리(Manage Users)**를 클릭합니다.
 - b. **생성(Create)**을 클릭하고 이름 및 암호 같은 사용자 세부 정보를 입력합니다.
 - c. **생성(Create)**을 클릭합니다.
ActiveVOS 사용자가 생성됩니다.

5. ActiveVOS 사용자의 관리 역할을 추가합니다.
 - a. 사용자 및 그룹(**Users and Groups**) 아래에서 관리 사용자 역할(**Administrative user roles**) > 사용자(**User**)를 클릭합니다.
 - b. 사용자에 대해 **모니터(Monitor)** 역할을 선택합니다.
 - c. 사용자를 **사용 가능(Available)** 목록에서 **역할에 매핑됨(Mapped to role)** 목록으로 이동합니다.
 - d. **확인(OK)**을 클릭합니다.
6. JAAS 응용 프로그램 로그인을 사용하도록 ActiveVOS를 구성합니다.
 - a. ActiveVOS 구성 마법사를 시작합니다.
 - b. **글로벌 보안 구성(Global Security Configuration)** 페이지에서 클러스터 커뮤니케이션 **JAAS 로그인 사용(Cluster Communications Use JAAS Login)**을 선택합니다.
 - c. 보안 **JAAS 로그인 이름(Security JAAS Login Name)** 목록에서 **ActiveVOSProvidedUser**를 선택합니다.
 - d. **JAAS 로그인 구성(JAAS Login Configuration)** 페이지에서 **ActiveVOS에서 제공된 사용자(ActiveVOS Provided User)**를 선택합니다.
 - e. ActiveVOS에서 제공된 사용자에 대해 사용자 이름 및 암호를 지정합니다.
 - f. 변경 내용을 저장합니다.

ActiveVOS 콘솔 관리자 생성

ActiveVOS를 사용하려는 경우 응용 프로그램 서버 컨테이너에서 **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.

- ▶ WebSphere 콘솔에 로그인하고 ActiveVOS 콘솔 관리자를 생성합니다.

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

MDM Hub 배포를 위한 SOAP 요청 제한 시간 구성

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

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

추가 응용 프로그램 서버 구성(선택 사항)

MDM Hub 환경의 요구 사항을 기반으로 추가 WebSphere 구성을 수행합니다.

다음 테이블에는 수행할 수 있는 구성이 설명되어 있습니다.

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

독립 실행형 처리 서버 인스턴스에 대한 WebSphere 구성

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

적절한 데이터 소스를 사용하도록 WebSphere를 구성하려면 다음과 같은 태스크를 수행합니다.

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

1단계. JDBC 드라이버 설치

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

지원되는 버전의 JDBC 드라이버를 가져오려면 Oracle에 문의합니다.

- ▶ JDBC 드라이버를 다음 디렉터리에 복사합니다.

<WebSphere installation directory>/lib

2단계. MDM Hub 마스터 데이터베이스 데이터 소스 생성

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

1. WebSphere 응용 프로그램 서버 관리 콘솔을 시작합니다.
2. 드라이버 라이브러리의 위치를 지정합니다.
 - a. 콘솔 탐색 트리에서 **환경**을 확장합니다.
 - b. **WebSphere 변수** 링크를 클릭합니다.

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

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

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

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

- d. **다음**을 클릭한 다음 **마침**을 클릭합니다.
- 5. MDM Hub 마스터 데이터베이스 데이터 소스를 생성합니다.
 - a. 생성한 JDBC 공급자를 클릭합니다.
구성 페이지가 표시됩니다.
 - b. **추가 속성** 아래에서 **데이터 소스**를 클릭합니다.
데이터 소스 페이지가 표시됩니다.
 - c. **새로 만들기**를 클릭합니다.

- d. 다음 데이터 소스 속성을 지정합니다.

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

- e. 다음을 클릭한 다음 **마침**을 클릭합니다.

3단계. 연산 참조 저장소 데이터 소스 생성

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

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

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

- e. **확인**을 클릭합니다.

- JDBC 공급자를 생성합니다.
 - 리소스 > JDBC**를 확장한 다음 **JDBC 공급자**를 클릭합니다.
JDBC 공급자 페이지가 표시됩니다.
 - 응용 프로그램이 데이터 소스를 사용할 범위를 선택합니다.

- c. **새로 만들기**를 클릭하고 다음과 같은 속성을 지정합니다.

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

- d. **다음**을 클릭한 다음 **마침**을 클릭합니다.

5. 연산 참조 저장소 데이터 소스를 생성합니다.

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

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

- e. **다음**을 클릭한 다음 **마침**을 클릭합니다.

여러 MDM Hub 마스터 데이터베이스에 대한 WebSphere 구성

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

HTTPS 프로토콜 구성

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

- SSL 활성화 WebSphere 포트를 생성합니다.
- 자체 서명된 인증서를 허용하도록 WebSphere를 구성합니다.

3. 다음과 같은 사용자 지정 JVM 속성을 구성합니다.

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

HTTPS 프로토콜 구성에 대한 자세한 내용은 WebSphere 설명서를 참조하십시오.

Informatica Data Director에 대한 WebSphere 구성

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

다음 구성을 수행해야 합니다.

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

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.

- Informatica 플랫폼. MDM Hub 설치의 일부로 Informatica 플랫폼을 설치하는 데 필요합니다.
- 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.

Informatica Platform 속성 파일 구성

MDM Hub 설치의 일부로 Informatica 플랫폼을 설치하려면 Informatica 플랫폼 속성 파일을 구성합니다. 속성 파일에서 설치 옵션을 지정하고 SilentInput.properties라는 이름으로 파일을 저장합니다.

1. 다음과 같은 디렉터리에서 SilentInput.properties 파일을 찾습니다. <배포 디렉터리>/Informatica 플랫폼>
2. 파일의 백업 사본을 생성합니다.
3. 텍스트 편집기에서 SilentInput.properties 파일을 엽니다.
4. 설치 매개 변수의 값을 구성하고 파일을 저장합니다.

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.properties와 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.properties와 같은 이름으로 속성 파일을 저장합니다.

제 3 장

Hub 저장소 설치

이 장에 포함된 항목:

- [MDM Hub 마스터 데이터베이스 생성, 33](#)
- [연산 참조 저장소 생성, 35](#)
- [MDM Hub 마스터 데이터베이스로 메타데이터 가져오기, 36](#)
- [연산 참조 저장소로 메타데이터 가져오기, 38](#)

MDM Hub 마스터 데이터베이스 생성

Oracle을 설치한 후 MDM Hub 마스터 데이터베이스를 생성합니다. 여러 MDM Hub 마스터 데이터베이스를 생성하려는 경우 각 MDM Hub 마스터 데이터베이스를 서로 다른 스키마에 생성해야 합니다. MDM Hub 마스터 데이터베이스의 기본 이름은 `CMX_SYSTEM`이지만 사용자 지정 이름을 사용할 수 있습니다.

참고: 배포 디렉터리에서 폴더 이름을 변경하면 메타데이터 가져오기 시 오류가 발생합니다.

Oracle 멀티테넌트 기능을 사용하려는 경우 PDB(플러깅 가능한 데이터베이스)에 MDM Hub 마스터 데이터베이스를 생성합니다.

1. 명령 프롬프트를 열고 다음 디렉터리로 이동합니다.
`<MDM Hub 배포 디렉터리>/database/bin`
2. MDM Hub 마스터 데이터베이스를 생성하려면 다음 명령을 실행합니다.
UNIX의 경우. `./sip_ant.sh create_system`
Windows의 경우. `sip_ant.bat create_system`
3. 다음 테이블에 설명되어 있는 프롬프트에 응답합니다.

참고: 프롬프트에 대괄호로 묶인 기본 텍스트가 표시됩니다. **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 실행하는 호스트의 이름입니다. 기본값은 localhost입니다. 중요: 클러스터 환경에서는 절대 호스트 이름 또는 IP 주소를 지정하여 캐싱 문제를 방지합니다.
데이터베이스 포트 번호 입력 [1521]	데이터베이스 수신기에서 사용하는 포트 번호입니다. 기본값은 1521입니다.
데이터베이스 서비스 이름 입력 [orcl]	Oracle 서비스의 이름입니다. 선택한 Oracle 연결 유형이 SERVICE이면 이 프롬프트가 표시됩니다. 기본값은 orcl입니다.
데이터베이스 SID 입력 [orcl]	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.
DBA 사용자 이름 입력 [SYS]	관리자의 사용자 이름입니다. 기본값은 SYS입니다.
DBA 암호 입력	관리자의 암호입니다.
MDM 인덱스 테이블스페이스의 이름 입력 [CMX_INDX]	MDM Hub 마스터 데이터베이스에 대한 인덱스 구성 요소를 포함할 테이블스페이스의 이름입니다. 기본값은 CMX_INDX입니다.

프롬프트	설명
MDM 임시 테이블스페이스(Oracle 임시 테이블스페이스 아님)의 이름 입력 [CMX_TEMP]	MDM Hub 마스터 데이터베이스에 대한 임시 구성 요소를 포함할 테이블스페이스의 이름입니다. 기본값은 CMX_TEMP입니다.
Oracle 임시 테이블스페이스의 이름 입력 [TEMP]	Oracle 임시 테이블스페이스의 이름입니다. 기본값은 TEMP입니다.

4. MDM Hub 마스터 데이터베이스가 성공적으로 생성되었는지 확인하려면 다음 디렉터리에서 sip_ant.log를 검토합니다.

<MDM Hub 배포 디렉터리>/database/bin

sip_ant.log 파일은 sip_ant 스크립트를 실행하여 MDM Hub 마스터 데이터베이스를 생성할 때 발생할 수 있는 모든 오류를 기록합니다.

연산 참조 저장소 생성

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

참고: 배포 디렉터리에서 폴더 이름을 변경하면 메타데이터 가져오기 시 오류가 발생합니다.

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: 배포 디렉터리에서 폴더 이름을 변경하면 메타데이터 가져오기 시 오류가 발생합니다.

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_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: 배포 디렉터리에서 폴더 이름을 변경하면 메타데이터 가져오기 시 오류가 발생합니다.

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, 40](#)

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, 42](#)
- [설치 프로그램 워크플로우 검토, 43](#)
- [설치 값 수집, 44](#)
- [설치 마법사에서 Hub 서버 설치, 49](#)
- [명령줄에서 Hub 서버 설치\(UNIX만 해당\), 49](#)
- [자동으로 Hub 서버 설치, 50](#)
- [클러스터의 노드에 Hub 서버 설치, 51](#)

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.

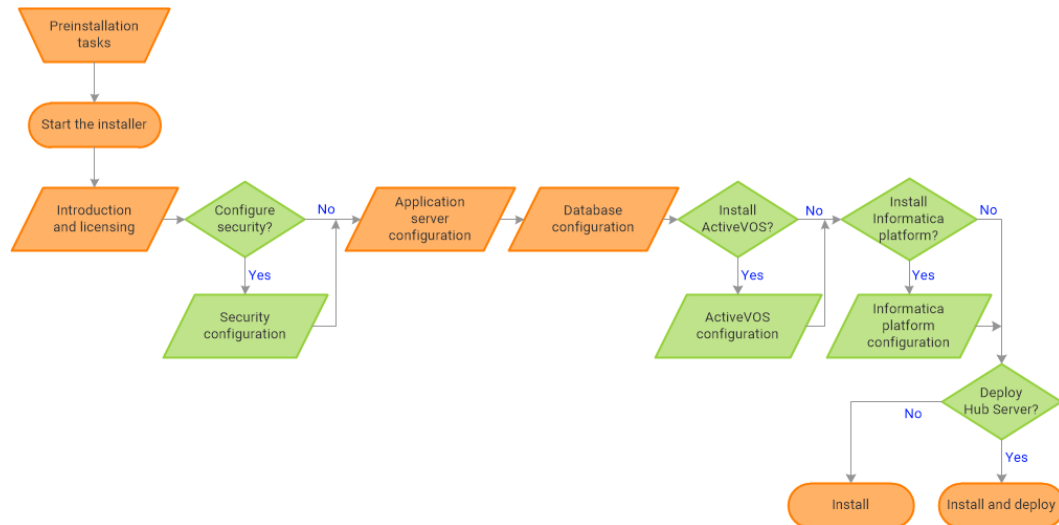
1. 설치 워크플로우를 검토합니다.

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. **Informatica Platform을 설치합니까?** 관리자는 Informatica Platform을 사용하여 소스 시스템의 데이터를 MDM에 매핑합니다. MDM의 랜딩 테이블을 사용할 수도 있습니다.

참고: Informatica Platform에 대한 설치를 이미 완료했다면 Hub 서버 설치 중에 이를 다시 설치할 필요가 없습니다. 나중에 Informatica Platform을 독립적으로 설치할 수도 있습니다.

4. **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: IBM WebSphere

설치에 필요한 WebSphere 세부 정보를 수집하려면 다음 테이블을 사용합니다.

속성	설명	기본값	서버 인스턴스 1 값
WebSphere 설치 디렉터리	WebSphere가 설치된 위치입니다.	-	
WebSphere 보안이 활성화되어 있습니까?	WebSphere 보안이 활성화되어 있는 경우 예를 선택하고 사용자 이름과 암호를 제공합니다.	아니요	
서버 이름	Hub 서버를 배포할 WebSphere 응용 프로그램 서버의 이름입니다. 클러스터된 환경에서는 하나의 클러스터 서버 이름과 해당 Bootstrap 포트 및 SOAP 커넥터 포트 정보를 입력합니다.	-	
Bootstrap 포트	지정한 서버에서 사용하는 Bootstrap 포트 번호입니다. 팁: 포트 정보를 찾으려면 WebSphere 관리 콘솔로 이동한 다음 응용 프로그램 서버 > <서버 이름> > 포트 를 클릭합니다.	2809	
SOAP 커넥터 포트	지정한 서버에서 사용하는 SOAP 커넥터 포트 번호입니다. 팁: 포트 정보를 찾으려면 WebSphere 관리 콘솔로 이동한 다음 응용 프로그램 서버 > <서버 이름> > 포트 를 클릭합니다.	8880	
프로필 이름	Hub 서버를 배포할 WebSphere 응용 프로그램 서버가 포함되어 있는 WebSphere 프로필의 이름입니다.	-	
사용자 이름	관리 권한이 있는 WebSphere 사용자의 이름입니다.	admin	
암호	WebSphere 관리자의 암호입니다.	-	

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

속성	서버 인스턴스 2 값	서버 인스턴스 3 값	서버 인스턴스 4 값
WebSphere 설치 디렉터리			
WebSphere 보안이 활성화되어 있습니까?			
서버 이름			
Bootstrap 포트			
SOAP 커넥터 포트			
프로필 이름			

속성	서버 인스턴스 2 값	서버 인스턴스 3 값	서버 인스턴스 4 값
사용자 이름			
암호			

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:2809	
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.	-	

Informatica Platform - 선택 사항

Informatica Platform을 설치하려는 경우 다음 위치를 기록합니다.

속성	설명	기본값	설치값
설치 속성 파일	Informatica Platform 배포 디렉터리의 SilentInstall.properties 파일에 대한 정규화된 경로입니다. Informatica Platform의 자동 설치를 위한 설치 속성이 포함되어 있습니다.	-	
Informatica Platform 설치 프로그램 파일	Informatica Platform 배포 디렉터리의 설치 프로그램 파일에 대한 정규화된 경로입니다.	-	

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	설치값
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.	-	

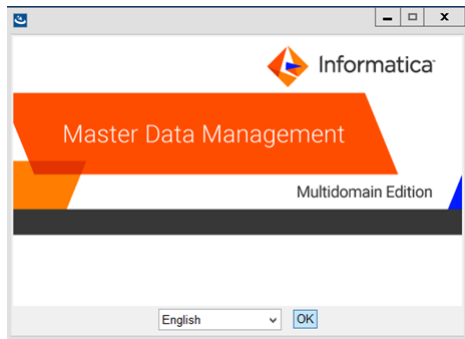
설치 마법사에서 Hub 서버 설치

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

동일한 사용자 이름을 사용하여 **Hub** 서버와 처리 서버를 설치해야 합니다.

1. 응용 프로그램 서버를 시작합니다.
2. 다음 디렉터리로 이동합니다.
<MDM Hub 배포 디렉터리>/<운영 체제 이름>/mrmsserver
3. 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 시작해야 합니다.

예를 들어 WebSphere 클러스터에 host1, host2, host3 및 host4에서 실행되고 RMI 포트 2812, 2813, 2814 및 2815를 사용하는 4개의 서버가 있습니다. 각 서버에 노드가 있습니다. node1, node2, node3 및 node4에 Hub 서버를 설치해야 합니다. 노드 순서에 관계없이 Hub 서버를 설치할 수 있습니다. 노드 하나(예: node2)에서 Hub 서버 설치를 완료한 이후에 다른 노드(예: node1 또는 node4)에서 설치를 시작합니다.

Hub 서버 설치의 디렉터리 구조가 모든 노드에서 동일한지 확인합니다.

1. WebSphere 클러스터를 시작하려면 다음 단계를 수행합니다.

- a. WebSphere 배포 관리자를 시작합니다.
- b. Hub 서버를 설치할 WebSphere 클러스터의 노드를 시작합니다.
- c. WebSphere 클러스터를 시작합니다.

2. 다음 디렉터리로 이동합니다.

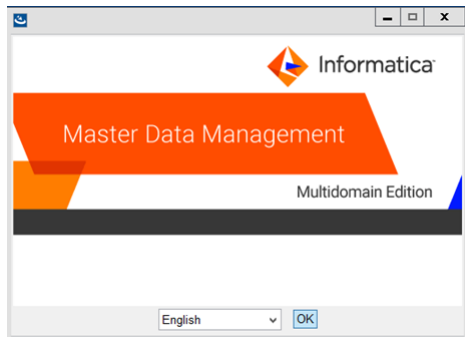
<MDM Hub 배포 디렉터리>/<운영 체제 이름>/mrmsserver

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

UNIX. ./hub_install.bin -DSIPERIAN_INSTALL_PREREQ_VALIDATION=false

Windows. hub_install.exe -DSIPERIAN_INSTALL_PREREQ_VALIDATION=false

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.

제 6 장

Hub 서버 설치 후 태스크

이 장에 포함된 항목:

- [설치 로그 파일 복사, 53](#)
- [버전 및 빌드 번호 확인, 53](#)
- [Elasticsearch 설치 및 구성, 54](#)
- [Hub 콘솔 클라이언트 구성\(조건부\), 54](#)
- [MDM Hub 마스터 데이터베이스 이름 구성, 54](#)
- [Hub 서버 EAR 파일 재배포, 55](#)
- [클래스 로더 구성, 55](#)
- [응용 프로그램 서버 설정 확인 및 구성\(조건부\), 56](#)
- [Hub 서버 응용 프로그램 배포\(조건부\), 57](#)
- [스크립트를 사용하여 Hub 서버 응용 프로그램 배포\(조건부\), 58](#)
- [수동으로 Hub 서버 응용 프로그램 배포\(조건부\), 59](#)
- [메타데이터 캐시 구성\(선택 사항\), 69](#)
- [Hub 콘솔 시작, 71](#)
- [연산 참조 저장소 등록, 72](#)

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.
WebSphere 로그	<ul style="list-style-type: none">- 파일 이름: startServer.log, stopServer.log, SystemErr.log 및 SystemOut.log- 위치: <WebSphere 설치 디렉터리>/profiles/<응용 프로그램 서버 프로파일 이름>/logs/<서버 이름>- 콘텐츠: WebSphere 서버 상태 및 성능 정보입니다.

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.

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.

`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.

Hub 서버 EAR 파일 재배포

수동으로 또는 Hub 서버 설치의 일부로 postInstallSetup 스크립트를 실행한 후 WebSphere 서버 관리 콘솔을 사용하여 Hub 서버 EAR 파일(siperian-mrm.ear)을 배포 해제 및 배포합니다. Hub 서버 설치 디렉터리에서 EAR 파일을 배포해야 합니다.

1. WebSphere 서버 관리 콘솔에 로그인합니다.
2. siperian-mrm.ear을 배포 해제합니다.
3. siperian-mrm.ear을 배포합니다.

EAR 파일은 다음 디렉터리에 있습니다.

<MDM Hub 설치 디렉터리> /hub/server

다음 배포 옵션을 구성합니다.

- **응용 프로그램 설치 준비** 패널에서 기본 바인딩 생성을 위한 배포 옵션을 활성화합니다.
- **모듈용 메타데이터** 패널에서 siperian-ejb.jar 모듈에 대한 metadata-complete 특성을 비활성화하여 모듈을 읽을 때마다 주석 기반 메타데이터를 검사합니다.

자세한 내용은 WebSphere 서버 설명서를 참조하십시오.

클래스 로더 구성

Hub 서버 응용 프로그램의 클래스 로더를 구성하려면 WebSphere 배포 관리자를 사용하면 됩니다.

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

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

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

- b. **엔터프라이즈 응용 프로그램** 페이지에서 응용 프로그램 EAR 파일의 이름을 클릭합니다.
- c. 응용 프로그램을 구성하는 페이지에서 **모듈 관리** 링크를 클릭합니다.
- d. 모듈 목록에서 웹 모듈의 링크를 클릭합니다.
- e. 웹 모듈 구성 페이지에서 클래스 로더 순서를 선택합니다.
- f. **적용**을 클릭한 다음 **확인**을 클릭합니다.

3. WebSphere를 다시 시작한 후 Hub 서버 응용 프로그램을 시작합니다.

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.
WebSphere 클러스터에 대한 Hub 서버 구성	WebSphere 클러스터에 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.

WebSphere 다중 노드 또는 클러스터 환경에 대한 Hub 서버 구성

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

예를 들어 WebSphere 클러스터 또는 다중 노드 환경에 `host1`과 `host2`에서 실행되고 RMI 포트 2812와 2813을 사용하는 2개의 서버가 있다고 가정해 보겠습니다. 두 서버에 WebSphere 속성을 구성해야 합니다.

1. 각 서버에서 다음과 같은 디렉터리의 `cmxserver.properties` 파일을 엽니다.

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

2. 다음 속성을 구성합니다.

속성	설명
cluster.flag	수동으로 추가해야 합니다. 클러스터링 활성화 여부를 지정합니다. 클러스터링을 활성화하려면 true로 설정합니다. 기본값은 false입니다.
cmx.appserver.hostname	모든 서버의 시스템 이름을 심표로 구분하여 지정합니다. 예를 들어 WebSphere 클러스터 또는 다중 노드 환경에 host1과 host2에서 실행되는 2개의 서버가 있는 경우 속성을 cmx.appserver.hostname=host1,host2로 설정합니다.
cmx.appserver.rmi.port	서버에서 사용하는 RMI 포트 번호를 심표로 구분하여 지정합니다. 예를 들어 WebSphere 클러스터 또는 다중 노드 환경에 포함된 서버가 RMI 포트 2812와 2813을 사용하는 경우 속성을 cmx.appserver.rmi.port=2812,2813으로 설정합니다.

속성 설명에서 첫 번째 서버의 호스트 이름 및 포트 번호는 host1 및 2812입니다. 두 번째 서버의 호스트 이름 및 포트 번호는 host2 및 2813입니다.

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.

WebSphere에서 보안이 활성화되어 있지 않으면 다음 명령을 실행합니다.

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>
```

WebSphere에서 보안을 활성화한 경우 다음 명령을 실행합니다.

UNIX의 경우.

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

Windows의 경우.

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

3. siperian-ejb.jar 모듈의 주식 기반 메타데이터에 대한 검사를 활성화하려면 WebSphere 서버 관리 콘솔을 사용하여 EAR 파일 siperian-mrm.ear을 배포 해제 및 배포합니다.

자세한 내용은 [“Hub 서버 EAR 파일 재배포” 페이지 55](#) 항목을 참조하십시오.

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 태스크를 수행합니다.

1. 데이터 소스 생성
2. JMS 메시지 대기열 구성
3. Hub 서버 응용 프로그램 다시 패키지
4. Hub 서버 응용 프로그램 배포
5. 클래스 로더 구성
6. Hub 서버에서 JMS 메시지 대기열 구성
7. 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. JDBC 드라이버를 설치합니다.
2. MDM Hub 마스터 데이터베이스 데이터 소스를 생성합니다.
3. 연산 참조 저장소 데이터 소스를 생성합니다.

1단계. JDBC 드라이버 설치

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

지원되는 버전의 JDBC 드라이버를 가져오려면 Oracle에 문의합니다.

- ▶ JDBC 드라이버를 다음 디렉터리에 복사합니다.

<WebSphere installation directory>/lib

2단계. MDM Hub 마스터 데이터베이스 데이터 소스 생성

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

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

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

- e. **확인**을 클릭합니다.
4. JDBC 공급자를 생성합니다.
 - a. **리소스 > JDBC**를 확장한 다음 **JDBC 공급자**를 클릭합니다.
JDBC 공급자 페이지가 표시됩니다.
 - b. 응용 프로그램이 데이터 소스를 사용할 범위를 선택합니다.

- c. **새로 만들기**를 클릭하고 다음과 같은 속성을 지정합니다.

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

- d. **다음**을 클릭한 다음 **마침**을 클릭합니다.

5. MDM Hub 마스터 데이터베이스 데이터 소스를 생성합니다.

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

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

- e. **다음**을 클릭한 다음 **마침**을 클릭합니다.

3단계. 연산 참조 저장소 데이터 소스 생성

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

- WebSphere 응용 프로그램 서버 관리 콘솔을 시작합니다.
- 드라이버 라이브러리의 위치를 지정합니다.
 - 콘솔 탐색 트리에서 **환경**을 확장합니다.
 - WebSphere 변수** 링크를 클릭합니다.
 - 다음과 같은 JDBC 드라이버 디렉터리를 가리키도록 JDBC 변수를 업데이트합니다.
<WebSphere 설치 디렉터리>/lib

3. 연산 참조 저장소가 사용할 보안 계정을 생성합니다.
 - a. 콘솔 탐색 트리에서 **보안**을 확장합니다.
 - b. **관리, 응용 프로그램 및 인프라 보안** 링크를 클릭합니다.
 - c. 인증에서 **Java Authentication and Authorization Service**를 확장하고 **J2C 인증 데이터**를 클릭합니다.
 - d. **새로 만들기**를 클릭하고 다음과 같은 속성을 설정합니다.

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

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

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

- d. **다음**을 클릭한 다음 **마침**을 클릭합니다.
5. 연산 참조 저장소 데이터 소스를 생성합니다.
 - a. 생성한 JDBC 공급자를 클릭합니다.
구성 페이지가 표시됩니다.
 - b. **추가 속성** 아래에서 **데이터 소스**를 클릭합니다.
데이터 소스 페이지가 표시됩니다.
 - c. **새로 만들기**를 클릭합니다.

d. 다음 데이터 소스 속성을 지정합니다.

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

e. 다음을 클릭한 다음 **마침**을 클릭합니다.

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. WebSphere 서버에 버스를 생성합니다.
2. 대기열 연결 팩터리를 구성합니다.
3. SIF 요청에 대한 JMS 메시지 대기열을 구성합니다.
4. 검색용 JMS 메시지 대기열을 구성합니다. (조건부)

1단계. WebSphere 서버에 버스 생성

JMS 메시지 대기열을 생성하려면 WebSphere 콘솔을 사용하여 버스를 생성합니다.

1. WebSphere 콘솔에서 **서비스 통합 > 버스**로 이동합니다.
2. SiperianBus라는 버스를 생성하고 변경 사항을 저장합니다.
3. 생성한 SiperianBus로 이동하고 **버스 멤버**를 클릭합니다.
4. **서버** 목록에 새 멤버를 추가합니다.
5. 응용 프로그램이 실행되고 있는 서버를 선택하고 **다음**을 클릭한 후 **데이터 저장소**를 선택합니다.
6. **다음**을 클릭하고 **생성된 JNDI 이름**으로 **기본 데이터 소스 생성**이 선택되어 있는지 확인합니다.
7. **다음**을 클릭한 다음 **마침**을 클릭합니다.
8. **저장**을 클릭합니다.
9. 생성한 SiperianBus로 이동하고 **대상**을 클릭합니다.
10. **새로 만들기**를 클릭하고 대상 유형으로 대기열을 선택한 후 **다음**을 클릭합니다.
11. 식별자로 이름 SiperianQueue를 사용하고 **다음**을 클릭합니다.
12. 생성한 버스 멤버를 선택하고 **다음**을 클릭합니다.
13. **마침**을 클릭한 다음 **저장**을 클릭합니다.

2단계. 대기열 연결 팩터리 구성

생성한 버스에 대한 대기열 연결 팩터리를 생성 및 구성합니다.

1. WebSphere 콘솔에서 리소스 > JMS > JMS 공급자로 이동합니다.
2. 목록에서 노드(예: **Node=<servername>Node01**)를 선택합니다.
3. 사용할 JMS 공급자를 선택합니다.
4. 대기열 연결 팩터리를 클릭합니다.
5. `siperian.mrm.jms.xaconnectionfactory`를 이름 및 JNDI 이름으로 지정합니다.
6. **SiperianBus**를 대기열 연결 팩터리의 버스 이름으로 선택합니다.
7. **저장**을 클릭합니다.
8. 대기열을 클릭하고 범위에 대해 동일한 노드(예: **Node=<servername>Node01**)를 선택합니다.

3단계. JMS 대기열 구성

버스 및 연결 팩터리를 구성한 후 시스템 JMS 메시지 대기열과 사용자 지정 JMS 메시지 대기열을 구성합니다. 그런 다음 대기열에 대한 활성화 사양을 생성합니다.

1. 대기열 페이지에서 **새로 만들기**를 클릭합니다.
2. JMS 공급자를 선택하고 **확인**을 클릭합니다.
3. 클러스터 환경에서는 **범위**를 하나 이상의 노드로 설정할 수 있습니다.
팁: 단일 노드에서 대기열을 구성한 다음 여러 노드로 확장하는 것이 좋습니다.
4. JMS 대기열에 대해 다음 옵션을 지정합니다.

옵션	시스템 대기열 값	사용자 지정 대기열 값
이름	<code>siperian.sif.jms.queue</code>	<code>siperian.sif.test.jms.queue</code>
JNDI 이름	<code>queue/siperian.sif.jms.queue</code>	<code>queue/siperian.sif.test.jms.queue</code>
설명	MDM JMS 시스템 대기열	MDM JMS 사용자 지정 대기열
버스 이름	SiperianBus	SiperianBus
대기열 이름	SiperianQueue	SiperianQueue

5. **확인**을 클릭합니다.
6. **JMS 활성화 사양**을 클릭하고 목록에서 범위를 선택합니다.
7. 활성화를 구성하려면 **새로 만들기**를 클릭합니다.
8. 다음 옵션을 지정합니다.

옵션	값
이름	SiperianActivation
JNDI 이름	SiperianActivation

옵션	값
대상 유형	대기열
대상 JNDI 이름	queue/siperian.sif.jms.queue
버스 이름	SiperianBus

9. **확인**을 클릭합니다.

4단계. 검색용 JMS 대기열 구성

검색 기능을 사용하여 Data Director 내에서 전체 텍스트 검색을 활성화하려면 JMS 대기열이 필요합니다. 검색 JMS 대기열을 활성화하거나 Hub 콘솔 대기열을 구성할 필요는 없습니다.

검색용 JMS 대기열을 구성하기 전에 Elasticsearch를 구성해야 합니다.

1. 대기열 페이지에서 **새로 만들기**를 클릭합니다.
2. JMS 공급자를 선택하고 **확인**을 클릭합니다.
3. 클러스터 환경에서는 **범위**를 하나 이상의 노드로 설정합니다.
팁: 단일 노드에서 대기열을 구성한 다음 여러 노드로 확장하는 것이 좋습니다.
4. 다음 옵션을 지정합니다.

옵션	값
이름	informatica.mdm.sss.jms.queue
JNDI 이름	queue/informatica.mdm.sss.jms.queue
설명	Siperian JMS 대기열 검색용
버스 이름	SiperianBus
대기열 이름	SiperianQueue
전송 모드	응용 프로그램

5. **확인**을 클릭합니다.

3단계. Hub 서버 EAR 파일 다시 패키지

cmxserver.properties 파일에서 cmx.home 속성을 편집하거나 응용 프로그램 서버 클러스터에서 설치한 경우 에 Hub 서버 EAR 파일을 다시 패키지합니다.

1. 이름이 EAR인 디렉토리를 생성합니다.
 - a. 다음 디렉터리로 이동합니다.
`<MDM Hub 설치 디렉터리>/hub/server/lib`
 - b. 다음 명령을 실행합니다.
`mkdir ear`

2. 사용자 지정 JAR 파일이 있는 경우 각 사용자 지정 JAR 파일을 이전 단계에서 생성한 EAR 디렉터리로 복사합니다.

사용자 지정 JAR 파일을 EAR 디렉터리로 복사하려면 다음 명령을 실행합니다.

```
copy <사용자 지정 JAR 파일 위치>/<사용자 지정 JAR 파일 이름>.jar ear
```

사용자 지정 사용자 종료를 위한 사용자 지정 JAR 파일이 필요할 수 있습니다.

3. 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
```

4단계. Hub 서버 응용 프로그램 배포

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

1. 기존 배포가 있는 경우 WebSphere 서버 관리 콘솔을 사용하여 다음 배포 파일을 배포 해제합니다.

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

2. WebSphere 관리 콘솔을 사용하여 이전 단계에서 나열된 파일을 배포합니다.

배포 파일은 다음 디렉터리에 있습니다.

```
<MDM Hub 설치 디렉터리> /hub/server
```

다음 배포 옵션을 구성합니다.

- **응용 프로그램 설치 준비** 패널에서 기본 바인딩 생성을 위한 배포 옵션을 활성화합니다.
- **모듈용 메타데이터** 패널에서 siperian-ejb.jar 모듈에 대한 metadata-complete 특성을 비활성화하여 모듈을 읽을 때마다 주석 기반 메타데이터를 검사합니다.
- 클러스터 노드에서 배포하는 경우 **설치 옵션 선택** 패널에서 응용 프로그램 배포를 위한 배포 옵션을 활성화합니다.

3. 클러스터 노드에 Hub 서버 응용 프로그램을 배포한 경우 다음 단계를 수행합니다.

- a. 클러스터, 배포 관리자 및 노드를 중지합니다.

- b. 노드, 배포 관리자 및 클러스터를 차례대로 시작합니다.

자세한 내용은 WebSphere 서버 설명서를 참조하십시오.

5단계. 클래스 로더 구성

Hub 서버 응용 프로그램을 수동으로 배포한 후 각 Hub 서버 응용 프로그램을 위한 클래스 로더를 구성합니다.

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

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

- 응용 프로그램 > 응용 프로그램 유형 > **WebSphere 엔터프라이즈 응용 프로그램**을 선택합니다.
 - 엔터프라이즈 응용 프로그램** 페이지에서 응용 프로그램 EAR 파일의 이름을 클릭합니다.
 - 응용 프로그램을 구성하는 페이지에서 **모듈 관리** 링크를 클릭합니다.
 - 모듈 목록에서 웹 모듈의 링크를 클릭합니다.
 - 웹 모듈 구성 페이지에서 클래스 로더 순서를 선택합니다.
 - 적용**을 클릭한 다음 **확인**을 클릭합니다.
- WebSphere를 다시 시작한 후 Hub 서버 응용 프로그램을 시작합니다.

6단계. 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.

- Hub 콘솔을 시작합니다.
- 메시지 대기열 서버를 추가합니다.
- 메시지 대기열을 추가합니다.

1단계. Hub 콘솔 시작

MDM Hub에 액세스하려면 Hub 콘솔을 시작합니다.

- 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.

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.

4. 사용자 이름과 암호를 입력하고 **확인**을 클릭합니다.

데이터베이스 변경 대화 상자가 표시됩니다.

5. 대상 데이터베이스를 선택합니다.

대상 데이터베이스는 MDM Hub 마스터 데이터베이스입니다.

6. 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. siperian.mrm.jms.xaconnectionfactory를 지정합니다.
표시 이름	Hub 콘솔에 표시되어야 하는 메시지 대기열 서버의 이름입니다. 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 획득을 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.

7단계. Informatica Data Director의 서버 리소스 구성

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

1. WebSphere 서버 관리 콘솔에서 **리소스 > URL**을 클릭합니다.
2. JNDI URL 리소스를 구성하려면 다음 속성을 설정합니다.

속성	값
범위	Hub 서버의 범위를 지정합니다.
이름	Hub 서버 홈 디렉터리
JNDI 이름	url/hubserver/home
사양	file:/// <Hub 서버 설치 디렉터리>

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

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.

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.

4. 사용자 이름과 암호를 입력하고 **확인**을 클릭합니다.

데이터베이스 변경 대화 상자가 표시됩니다.

5. 대상 데이터베이스를 선택합니다.

대상 데이터베이스는 MDM Hub 마스터 데이터베이스입니다.

6. 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 데이터베이스를 선택하고 연결을 클릭합니다.
3. 구성 작업 영역에서 데이터베이스 도구를 클릭합니다.
4. 쓰기 잠금 메뉴에서 잠금 획득을 클릭합니다.
5. 데이터베이스 창에서 데이터베이스 등록 단추를 클릭합니다.
Informatica MDM Hub 연결 마법사가 표시됩니다.
6. Oracle 데이터베이스 유형 옵션을 선택하고 다음을 클릭합니다.
7. 데이터베이스의 연결 속성을 구성합니다.
 - a. Oracle 연결 방법을 선택하고 다음을 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 다음을 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.

속성	설명
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 연결을 테스트하기 전에 응용 프로그램 서버를 다시 시작해야 합니다.

데이터베이스 테스트 대화 상자에 데이터베이스 연결 테스트 결과가 표시됩니다.

11. **확인**을 클릭합니다.

연산 참조 저장소가 등록되고 데이터베이스에 대한 연결이 테스트됩니다.

제 7 장



처리 서버 설치

이 장에 포함된 항목:

- [처리 서버 설치, 74](#)
- [설치 프로그램 워크플로우 검토, 75](#)
- [설치 값 수집, 76](#)
- [설치 마법사에서 처리 서버 설치, 78](#)
- [명령줄에서 처리 서버 설치\(UNIX만 해당\), 79](#)
- [자동으로 처리 서버 설치, 80](#)
- [클러스터의 노드에 처리 서버 설치, 80](#)

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.

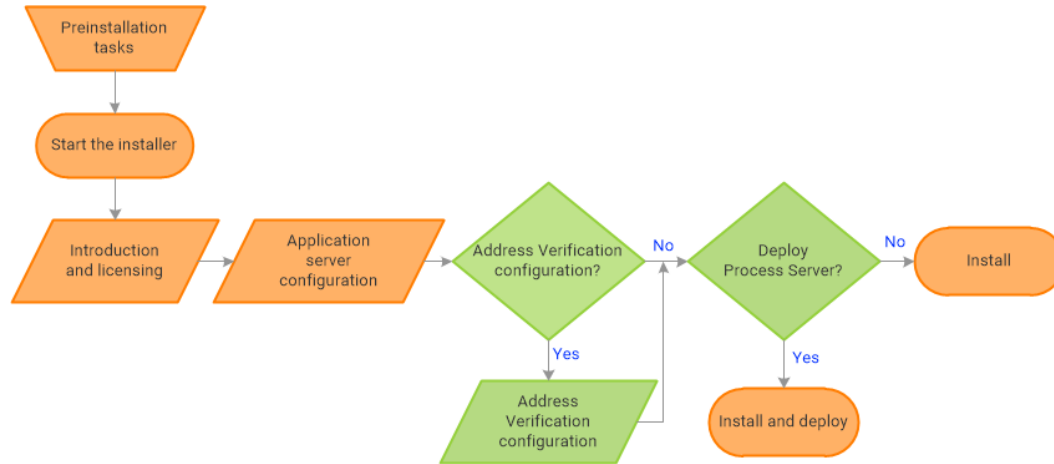
1. 설치 워크플로우를 검토합니다.

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 Address Verification을 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 Address Verification을 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: IBM WebSphere

설치에 필요한 WebSphere 세부 정보를 수집하려면 다음 테이블을 사용합니다.

속성	설명	기본값	서버 인스턴스 1 값
WebSphere 설치 디렉터리	WebSphere가 설치된 위치입니다.	-	
WebSphere 보안이 활성화되어 있습니까?	WebSphere 보안이 활성화되어 있는 경우 예를 선택하고 사용자 이름과 암호를 제공합니다.	아니요	
서버 이름	처리 서버를 배포할 WebSphere 응용 프로그램 서버의 이름입니다. 클러스터된 환경에서는 하나의 클러스터 서버 이름과 해당 Bootstrap 포트 및 SOAP 커넥터 포트 정보를 입력합니다.	-	
Bootstrap 포트	지정한 서버에서 사용하는 Bootstrap 포트 번호입니다. 팁: 포트 정보를 찾으려면 WebSphere 관리 콘솔로 이동한 다음 응용 프로그램 서버 > <서버 이름> > 포트 를 클릭합니다.	2809	
SOAP 커넥터 포트	지정한 서버에서 사용하는 SOAP 커넥터 포트 번호입니다. 팁: 포트 정보를 찾으려면 WebSphere 관리 콘솔로 이동한 다음 응용 프로그램 서버 > <서버 이름> > 포트 를 클릭합니다.	8880	
프로필 이름	처리 서버를 배포할 WebSphere 응용 프로그램 서버가 포함되어 있는 WebSphere 프로필의 이름입니다.	-	
사용자 이름	관리 권한이 있는 WebSphere 사용자의 이름입니다.	admin	
암호	WebSphere 관리자의 암호입니다.	-	

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

속성	서버 인스턴스 2 값	서버 인스턴스 3 값	서버 인스턴스 4 값
WebSphere 설치 디렉터리			
WebSphere 보안이 활성화되어 있습니까?			
서버 이름			
Bootstrap 포트			
SOAP 커넥터 포트			
프로필 이름			

속성	서버 인스턴스 2 값	서버 인스턴스 3 값	서버 인스턴스 4 값
사용자 이름			
암호			

Informatica Address Verification

참고: Address Verification의 이전 이름은 Informatica AddressDoctor입니다.

Informatica Address Verification을 설치하려는 경우 다음 속성을 기록합니다.

속성	설명	기본값 값	서버 인스턴스 1 값
구성 파일	Informatica Address Verification 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 Address Verification 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.	-	

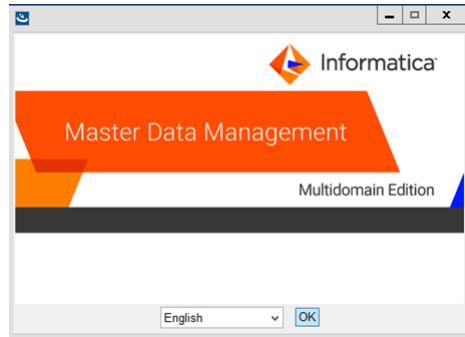
설치 마법사에서 처리 서버 설치

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 진행합니다.

처리 서버 설치의 디렉터리 구조가 모든 노드에서 동일한지 확인합니다.

1. **WebServer** 클러스터를 시작하려면 다음 단계를 수행합니다.
 - a. **WebSphere** 배포 관리자를 시작합니다.
 - b. 처리 서버를 설치할 **WebServer** 클러스터의 노드를 시작합니다.
 - c. **WebServer** 클러스터를 시작합니다.

2. 다음 디렉터리로 이동합니다.

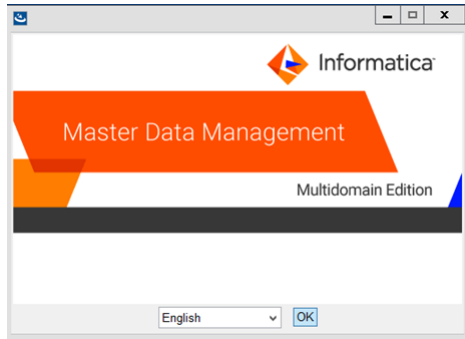
<MDM Hub 배포 디렉터리>/<운영 체제 이름>/mrncleanse

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

UNIX. `./hub_cleanse_install.bin -DSIPERIAN_INSTALL_PREREQ_VALIDATION=false`

Windows. `hub_cleanse_install.exe -DSIPERIAN_INSTALL_PREREQ_VALIDATION=false`

처리 서버 설치 마법사가 시작됩니다.



4. 온라인 지침을 따릅니다. 메시지가 표시되면 수집한 설치 값을 입력합니다.
5. **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, 82](#)
- [버전 및 빌드 번호 확인, 83](#)
- [WebSphere 다중 노드 또는 클러스터 환경에 대한 처리 서버 구성, 83](#)
- [처리 서버 EAR 파일 재배포, 84](#)
- [클래스 로더 구성, 84](#)
- [처리 서버 응용 프로그램 배포\(조건부\), 85](#)
- [처리 서버에 대한 보안 통신 활성화, 90](#)
- [Elasticsearch 설치 및 구성, 91](#)
- [일치 인구집단 구성, 92](#)
- [처리 서버와 정리 엔진 구성, 93](#)

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 검사의 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.

로그 파일 유형	설명
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.
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.
WebSphere 로그	<ul style="list-style-type: none"> - 파일 이름. startServer.log, stopServer.log, SystemErr.log 및 SystemOut.log - 위치. <WebSphere 설치 디렉터리>/profiles/AppSrv01/logs/<서버 이름> - 콘텐츠. 서버 시작 및 중지 및 성능에 대한 정보를 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.

WebSphere 다중 노드 또는 클러스터 환경에 대한 처리 서버 구성

WebSphere 다중 노드 또는 클러스터 환경에서 처리 서버를 설치한 경우 WebSphere 클러스터에 대한 처리 서버를 구성합니다. WebSphere 환경에 대한 Hub 서버를 구성하려면 cmxcleanse.properties 파일에서 cluster.flag 속성을 추가합니다.

1. 다음 디렉터리에서 cmxcleanse.properties 파일을 엽니다.

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

2. cluster.flag 속성을 추가합니다.

속성은 클러스터링을 활성화할지 여부를 지정합니다. 클러스터링을 활성화하려면 true로 설정합니다. 기본 값은 false입니다.

처리 서버 EAR 파일 재배포

수동으로 또는 처리 서버 설치의 일부로 `postInstallSetup` 스크립트를 실행한 후 WebSphere 서버 관리 콘솔을 사용하여 처리 서버 EAR 파일(`siperian-mrm-cleanse.ear`)을 배포 해제 및 배포합니다. 처리 서버 설치 디렉터리에서 EAR 파일을 배포해야 합니다.

1. WebSphere 서버 관리 콘솔에 로그인합니다.
2. `siperian-mrm-cleanse.ear`을 배포 해제합니다.
3. `siperian-mrm-cleanse.ear` 파일을 배포합니다.

EAR 파일은 다음 디렉터리에 있습니다.

<MDM Hub 설치 디렉터리>/hub/cleanse

다음 배포 옵션을 구성합니다.

- **응용 프로그램 설치 준비** 패널에서 기본 바인딩 생성을 위한 배포 옵션을 활성화합니다.
- **모듈용 메타데이터** 패널에서 `siperian-cleanse-ejb.jar` 모듈에 대한 `metadata-complete` 특성을 비활성화하여 모듈을 읽을 때마다 주석 기반 메타데이터를 검사합니다.

응용 프로그램 배포에 대한 자세한 내용은 WebSphere 서버 설명서를 참조하십시오.

클래스 로더 구성

처리 서버 응용 프로그램의 클래스 로더를 구성하려면 WebSphere 배포 관리자를 사용하면 됩니다.

1. 처리 서버 응용 프로그램 `siperian-mrm-cleanse.ear`의 클래스 로더는 상위 클래스 로더가 있는 클래스를 마지막으로 로드하도록 구성됩니다.

상위 클래스 로더가 있는 클래스를 맨 처음 로드하도록 클래스 로더가 구성되어 있으면 응용 프로그램의 클래스 로더를 구성합니다.

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

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

- a. **응용 프로그램 > 응용 프로그램 유형 > WebSphere 엔터프라이즈 응용 프로그램**을 선택합니다.
- b. **엔터프라이즈 응용 프로그램** 페이지에서 응용 프로그램 EAR 파일의 이름을 클릭합니다.

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

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.ear`을 배포합니다.
3. 클래스 로더를 구성합니다.

1단계. 데이터 소스 생성(조건부)

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

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

1단계. JDBC 드라이버 설치

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

지원되는 버전의 JDBC 드라이버를 가져오려면 Oracle에 문의합니다.

- ▶ JDBC 드라이버를 다음 디렉터리에 복사합니다.

`<WebSphere installation directory>/lib`

2단계. MDM Hub 마스터 데이터베이스 데이터 소스 생성

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

1. WebSphere 응용 프로그램 서버 관리 콘솔을 시작합니다.

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

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

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

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

- d. **다음**을 클릭한 다음 **마침**을 클릭합니다.
5. MDM Hub 마스터 데이터베이스 데이터 소스를 생성합니다.
 - a. 생성한 JDBC 공급자를 클릭합니다.
구성 페이지가 표시됩니다.
 - b. **추가 속성** 아래에서 **데이터 소스**를 클릭합니다.

데이터 소스 페이지가 표시됩니다.

- c. 새로 만들기를 클릭합니다.
- d. 다음 데이터 소스 속성을 지정합니다.

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

- e. 다음을 클릭한 다음 마침을 클릭합니다.

3단계. 연산 참조 저장소 데이터 소스 생성

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

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

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

- e. **확인**을 클릭합니다.

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

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

- d. **다음**을 클릭한 다음 **마침**을 클릭합니다.
5. 연산 참조 저장소 데이터 소스를 생성합니다.
 - a. 생성한 JDBC 공급자를 클릭합니다.
구성 페이지가 표시됩니다.
 - b. **추가 속성** 아래에서 **데이터 소스**를 클릭합니다.
데이터 소스 페이지가 표시됩니다.
 - c. **새로 만들기**를 클릭합니다.
 - d. 다음 데이터 소스 속성을 지정합니다.

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

- e. **다음**을 클릭한 다음 **마침**을 클릭합니다.

2단계. 처리 서버 응용 프로그램 배포(조건부)

응용 프로그램 서버 다중 노드 환경 또는 클러스터 환경에서 설치를 수행하는 경우나 `postInstallSetup` 스크립트가 실행되거나 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

참고: WebSphere에서 보안을 활성화한 경우 postInstallSetup.sh -Dwebsphere.password=<WebSpherePassword>를 실행합니다.

Windows의 경우. postInstallSetup.bat

참고: WebSphere에서 보안을 활성화한 경우 postInstallSetup.bat -Dwebsphere.password=<WebSphere Password>를 실행합니다.

3. siperian-cleanse-ejb.jar 모듈의 주석 기반 메타데이터에 대한 검사를 활성화하려면 WebSphere 서버 관리 콘솔을 사용하여 EAR 파일 siperian-mrm-cleanse.ear을 배포 해제 및 배포합니다.

자세한 내용은 [“처리 서버 EAR 파일 재배포” 페이지 84](#) 항목을 참조하십시오.

처리 서버 응용 프로그램 수동 배포(조건부)

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

1. 기존 배포가 있는 경우 WebSphere 서버 관리 콘솔을 사용하여 siperian-mrm-cleanse.ear을 배포 해제합니다.

2. WebSphere 서버 관리 콘솔을 사용하여 siperian-mrm-cleanse.ear 파일을 배포합니다.

배포 파일은 다음 디렉터리에 있습니다.

<MDM Hub 설치 디렉터리>/hub/cleanse

다음 배포 옵션을 구성합니다.

- **응용 프로그램 설치 준비** 패널에서 기본 바인딩 생성을 위한 배포 옵션을 활성화합니다.
- **모듈용 메타데이터** 패널에서 siperian-cleanse-ejb.jar 모듈에 대한 metadata-complete 특성을 비활성화하여 모듈을 읽을 때마다 주석 기반 메타데이터를 검사합니다.
- 클러스터 노드에서 배포하는 경우 **설치 옵션 선택** 패널에서 응용 프로그램 배포를 위한 배포 옵션을 활성화합니다.

3. 클러스터 노드에 배포한 경우 다음 단계를 수행합니다.

- 클러스터, 배포 관리자 및 노드를 중지합니다.
- 노드, 배포 관리자 및 클러스터를 차례대로 시작합니다.

응용 프로그램 배포에 대한 자세한 내용은 **WebSphere** 서버 설명서를 참조하십시오.

3단계. 클래스 로더 구성

처리 서버 응용 프로그램을 수동으로 배포한 후 응용 프로그램에 대한 클래스 로더를 구성합니다.

1. 처리 서버 응용 프로그램 **siperian-mrm-cleanse.ear**의 클래스 로더를 구성합니다.

상위 클래스 로더가 있는 클래스를 맨 처음 로드하도록 클래스 로더가 구성되어 있으면 응용 프로그램의 클래스 로더를 구성합니다.

- 응용 프로그램 > 응용 프로그램 유형 > WebSphere 엔터프라이즈 응용 프로그램**을 선택합니다.
- 엔터프라이즈 응용 프로그램** 페이지에서 응용 프로그램 중 하나를 클릭합니다.
- 응용 프로그램을 구성하는 페이지에서 **클래스 로드 및 업데이트 검색** 링크를 클릭합니다.
- 클래스 로더** 구성 페이지에서 **로컬 클래스 로더를 사용하여 로드된 클래스 먼저(상위 클래스는 마지막에)** 클래스 로더 순서 옵션을 선택합니다.
- 적용**을 클릭한 다음 **확인**을 클릭합니다.

2. 다음 응용 프로그램 **EAR** 파일의 웹 모듈에 대한 클래스 로더를 구성합니다.

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

- 응용 프로그램 > 응용 프로그램 유형 > WebSphere 엔터프라이즈 응용 프로그램**을 선택합니다.
- 엔터프라이즈 응용 프로그램** 페이지에서 응용 프로그램 **EAR** 파일의 이름을 클릭합니다.
- 응용 프로그램을 구성하는 페이지에서 **모듈 관리** 링크를 클릭합니다.
- 모듈 목록에서 웹 모듈의 링크를 클릭합니다.
- 웹 모듈 구성 페이지에서 클래스 로더 순서를 선택합니다.
- 적용**을 클릭한 다음 **확인**을 클릭합니다.

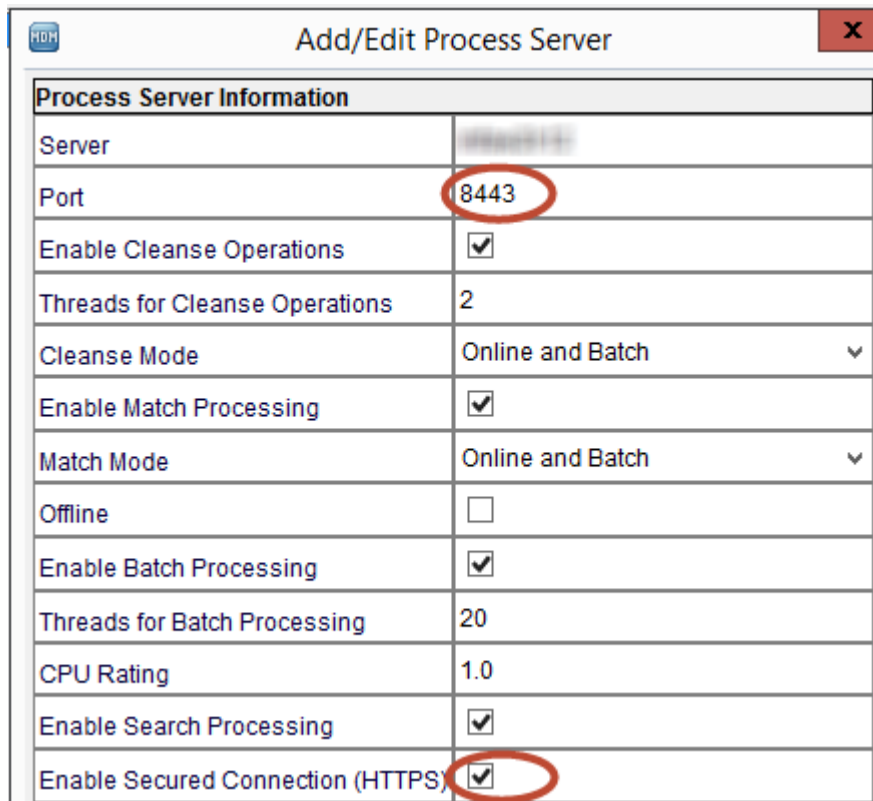
3. **WebSphere**를 다시 시작한 후 처리 서버 응용 프로그램을 시작합니다.

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.

- 처리 서버에 대한 서명된 인증서를 인증서 저장소에 생성합니다.
- 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 활성화(HTTPS)** 확인란을 선택합니다.



Process Server Information	
Server	
Port	8443
Enable Cleanse Operations	<input checked="" type="checkbox"/>
Threads for Cleanse Operations	2
Cleanse Mode	Online and Batch
Enable Match Processing	<input checked="" type="checkbox"/>
Match Mode	Online and Batch
Offline	<input type="checkbox"/>
Enable Batch Processing	<input checked="" type="checkbox"/>
Threads for Batch Processing	20
CPU Rating	1.0
Enable Search Processing	<input checked="" type="checkbox"/>
Enable Secured Connection (HTTPS)	<input checked="" type="checkbox"/>

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:

- [WebSphere 클러스터 환경에서 ActiveVOS 설치 및 배포, 94](#)
- [WebSphere 환경에서 트러스트된 사용자 생성, 97](#)
- [사용자 및 그룹을 보안 프로필에 추가, 98](#)

WebSphere 클러스터 환경에서 ActiveVOS 설치 및 배포

WebSphere 클러스터 환경에서 ActiveVOS 콘솔을 사용하여 MDM ID 서비스 공급자에 연결하려면 MDM Hub 설치 후 다음 단계를 수행합니다.

Hub 서버, 포함된 ActiveVOS 및 처리 서버를 설치한 후 다음 단계를 완료합니다.

1. WebSphere 작업 관리자를 구성합니다.
2. WebSphere 시간 관리자를 구성합니다.
3. JAAS 응용 프로그램 로그인을 구성합니다.
4. ActiveVOS 서버 및 ActiveVOS Central을 설치합니다.
5. ActiveVOS 설치 파일을 편집합니다.
6. ActiveVOS 및 Identity Resolution을 배포합니다.

참고: 설치 및 배포 절차는 WebSphere 버전에 따라 달라질 수 있습니다. 사용 중인 WebSphere 버전에 관련된 정보는 해당 버전의 WebSphere 설명서를 참조하십시오.

WebSphere 작업 관리자 구성

WebSphere 관리 콘솔을 사용하여 ActiveVOS용 엔터프라이즈 및 시스템 작업 관리자를 생성합니다. 작업 관리자는 비동기 Bean을 사용하는 ActiveVOS 응용 프로그램 구성 요소에 대한 스레드 풀 역할을 합니다.

1. WebSphere 관리 콘솔에서 리소스 > 비동기 Bean > 작업 관리자로 이동합니다.

2. 다음 속성과 함께 작업 관리자를 추가합니다.

속성	작업 관리자 1	작업 관리자 2
이름	ActiveVOS 엔터프라이즈 작업 관리자	ActiveVOS 시스템 작업 관리자
JNDI 이름	wm/ActiveVOS	wm/ActiveVOSSystem
최소 스레드 수	10	5
최대 스레드 수	150	50
범위	클러스터	클러스터
서비스 작업 영역 및 보안	활성화됨	활성화됨

WebSphere 시간 관리자 구성

WebSphere 관리 콘솔을 사용하여 ActiveVOS용 시간 관리자를 생성합니다. 시간 관리자는 비동기 Bean을 사용하는 ActiveVOS 응용 프로그램 구성 요소에 대한 스레드 풀 역할을 합니다.

1. WebSphere 관리 콘솔에서 리소스 > 비동기 Bean > 시간 관리자로 이동합니다.
2. 다음 속성과 함께 시간 관리자를 추가합니다.

속성	값
이름	ActiveVOS 엔터프라이즈 시간 관리자
JNDI 이름	tm/ActiveVOS
경보 스레드의 수	10
범위	클러스터
서비스 작업 영역 및 보안	활성화

JAAS 응용 프로그램 로그인 구성

WebSphere 관리 콘솔을 사용하여 ActiveBPELIdentityAssertion 및 ActiveBPELProvidedUser 응용 프로그램에 대한 응용 프로그램 로그인을 구성합니다.

1. ActiveBPEL ID 어설션에 대한 JAAS 로그인 모듈을 구성합니다.
 - a. WebSphere 관리 콘솔에서 보안 > 글로벌 보안 > JAAS 구성 > 응용 프로그램 로그인 구성으로 이동합니다.
 - b. 응용 프로그램 로그인을 추가하고 별칭을 ActiveBPELIdentityAssertion으로 지정합니다.
 - c. JAAS 로그인 모듈 섹션에서 다음 모듈 클래스를 지정된 순서로 추가합니다.
 1. com.activeee.rt.websphere.trustvalidation.AeIdentityAssertionLoginModule
 2. com.ibm.wsspi.security.common.auth.module.IdentityAssertionLoginModule
 - d. 변경 내용을 저장합니다.

2. ActiveBPEL 제공 사용자에게 대한 JAAS 로그인 모듈을 구성합니다.
 - a. 응용 프로그램 로그인을 추가하고 별칭을 ActiveBPELProvidedUser로 지정합니다.
 - b. **JAAS 로그인 모듈** 섹션에서 다음 로그인 모듈 클래스를 추가합니다.
com.activeee.rt.websphere.trustvalidation.AeBasicLoginModule
 - c. 모듈에 대해 다음 사용자 지정 속성을 추가합니다.

속성	설명
사용자 이름	ActiveVOS 관리자의 사용자 이름입니다.
암호	ActiveVOS 관리자의 암호입니다.

- d. 변경 내용을 저장합니다.
3. ActiveVOS 관리자의 관리 역할을 추가합니다.
 - a. **사용자 및 그룹**에서 **관리자 역할 > 사용자**를 클릭합니다.
 - b. 사용자에게 대해 **모니터링** 역할을 선택합니다.
 - c. 사용자를 **사용 가능** 목록에서 **역할에 매핑됨** 목록으로 이동합니다.
 - d. **확인**을 클릭합니다.

ActiveVOS 서버 및 ActiveVOS Central 설치 완료

WebSphere 환경을 구성한 후 ActiveVOS 서버 및 ActiveVOS Central 설치를 완료하려면 설치 프로그램 유틸리티를 실행합니다.

1. 명령 프롬프트를 열고 다음 디렉터리로 이동합니다.
<MDM Hub installation directory>/<ActiveVOS directory>/server-enterprise/websphere_config/bin
2. ActiveVOS 서버 및 ActiveVOS Central 설치를 완료하려면 다음 설치 프로그램 유틸리티를 실행합니다.
UNIX의 경우. ./config_deploy.sh
Windows의 경우. config_deploy.bat
3. 화면 지침을 따릅니다.

ActiveVOS 설치 파일 편집

ActiveVOS를 배포하기 전에 ActiveVOS 설치 파일을 편집합니다.

1. 다음 디렉터리로 이동합니다.
<MDM Hub 설치 디렉터리>/hub/server/conf
2. avos.install.properties 파일을 편집하여 install.web.application 속성을 false로 설정합니다.
3. 다음 디렉터리로 이동합니다.
<MDM Hub 설치 디렉터리>/avos/server/server-enterprise/websphere_config/bin
4. 다음 코드 블록을 주석 처리하여 deployer.xml 파일을 편집합니다.

```
<!-- <target name="deploy.resources" depends="deploy.timer.manager,deploy.work.manager"/>
<target name="deploy.timer.manager">
<echo message="{basedir}"/>
```



```

<run.wsadmin script="{basedir}
/scripts/timermanager.jacl"/>
</target>
<target name="deploy_work_manager">
<run.wsadmin script="{basedir}/scripts/workmanager.jacl"/>

<run.wsadmin script="{basedir}
/scripts/systemworkmanager.jacl"/>
</target>
<target name="deploy_jaas">
<run.wsadmin script="{basedir}/scripts/jaaslogin.jacl"/>

</target>

<target name="deploy_apps">

<run.wsadmin script="{basedir}
/scripts/installapp.jacl"/>
</target> -->

```

ActiveVOS 및 Identity Resolution 배포

ActiveVOS 설치 파일을 편집한 후 ActiveVOS 서버, ActiveVOS Central 및 MDM Identity Resolution을 배포합니다.

자세한 내용은 WebSphere 서버 설명서를 참조하십시오.

1. 다음 디렉터리로 이동합니다.
 <MDM Hub 설치 디렉터리>/hub/server/bin
2. ActiveVOS 서버 및 MDM Identity Resolution을 배포하려면 다음 명령을 실행합니다.

```

sip_ant.bat deploy_mdm_identity_resolution
sip_ant.bat deploy_avos_server

```
3. WebSphere 관리 콘솔에서 다음 ActiveVOS 응용 프로그램을 배포합니다.
 - ave_websphere.ear
 - activevos-central.war

ActiveVOS 응용 프로그램은 다음 디렉터리에 있습니다. <MDM Hub 설치 디렉터리> /hub/server

참고: 배포에서 기본 바인딩을 생성하도록 허용하는 옵션을 선택해야 합니다.

WebSphere 환경에서 트러스트된 사용자 생성

ActiveVOS 워크플로우 엔진을 사용하려면 트러스트된 사용자를 생성하여 abTrust, abServiceConsumer 및 abTaskClient 역할에 매핑합니다.

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

1. WebSphere 콘솔에서 ave_websphere EAR 응용 프로그램을 중지합니다.
2. 트러스트된 사용자를 생성합니다.
3. ave_websphere.ear 파일을 엽니다.
4. ave_websphere.ear 파일에서 트러스트된 사용자를 abTrust, abServiceConsumer 및 abTaskClient 역할에 매핑합니다.
5. WebSphere 프로필을 다시 시작합니다.

사용자 및 그룹을 보안 프로필에 추가

MDM Hub 관리자 및 사용자를 위한 사용자 및 그룹을 작성합니다. 사용자 및 그룹 작성 방법에 대한 자세한 내용은 WebSphere 설명서를 참조하십시오.

1. WebSphere 콘솔에서 ActiveVOS 서버로 인증할 각 MDM Hub 관리자 및 사용자에 대한 사용자를 작성합니다.
2. MDM Hub 관리자를 위한 그룹을 작성합니다.
3. MDM Hub 사용자를 위한 그룹을 작성합니다.
4. 관리자를 MDM Hub 관리자 그룹에 추가합니다.
5. 사용자를 MDM Hub 사용자 그룹에 추가합니다.

제 10 장

비즈니스 항목 어댑터에 대한 ActiveVOS 설치 후 태스크

이 장에 포함된 항목:

- [ActiveVOS 웹 응용 프로그램, 99](#)
- [비즈니스 항목 워크플로우 어댑터의 ActiveVOS URN 구성, 100](#)
- [ActiveVOS URL의 프로토콜 구성, 100](#)
- [ActiveVOS 프로토콜을 HTTPS로 설정, 101](#)
- [기본 워크플로우 엔진 구성, 101](#)
- [ActiveVOS의 MDM ID 서비스 구성, 102](#)
- [태스크 구성, 103](#)

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, 알람 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 URL의 프로토콜 구성

build.properties 파일에서 ActiveVOS URL의 프로토콜을 구성할 수 있습니다.

1. 다음 디렉터리에서 build.properties 파일을 찾습니다.
 - UNIX의 경우. <MDM Hub 설치 디렉터리>/hub/bin
 - Windows의 경우. <MDM Hub 설치 디렉터리>\hub\bin
2. 다음 매개 변수의 프로토콜을 http에서 https로 변경합니다.
 - activevos.mdm.sif.url
 - activevos.mdm.cs.url
3. build.properties 파일을 저장합니다.
4. 다음 디렉터리로 이동합니다.
 - UNIX의 경우. <MDM Hub 설치 디렉터리>/hub/server
 - Windows의 경우. <MDM Hub 설치 디렉터리>\hub\server
5. 다음 명령을 실행하여 Hub 서버 응용 프로그램을 배포하고 보안 구성에 변경 사항을 적용합니다.

UNIX의 경우

WebLogic

```
patchInstallSetup.sh -Dweblogic.password=<WebLogic 암호> -Ddatabase.password=<데이터베이스 암호>
```

WebSphere

```
patchInstallSetup.sh -Ddatabase.password=<데이터베이스 암호>
```

JBoss

```
patchInstallSetup.sh -Ddatabase.password=<데이터베이스 암호>
```

Windows의 경우

WebLogic

```
patchInstallSetup.bat -Dweblogic.password=<WebLogic 암호> -Ddatabase.password=<데이터베이스 암호>
```

WebSphere

```
patchInstallSetup.bat -Ddatabase.password=<데이터베이스 암호>
```

JBoss

```
patchInstallSetup.bat -Ddatabase.password=<데이터베이스 암호>
```

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

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 **admin**인 MDM Hub 사용자의 암호를 입력합니다.
참고: 나중에 관리 사용자의 암호를 변경할 경우에는 ActiveVOS ID 서비스 설정에 새 암호를 입력해야 합니다.
4. **업데이트**를 클릭합니다.
5. ActiveVOS가 MDM Hub에 admin사용자로 로그인할 수 있는지 그리고 **테스트할 사용자**로 지정된 사용자의 역할 목록을 검색할 수 있는지 테스트합니다.
 - a. **테스트** 탭을 선택합니다.
 - b. **테스트할 사용자** 필드에 역할에 할당된 MDM Hub 사용자를 입력합니다.
 - c. **테스트 설정**을 클릭합니다.

참고: 연산 참조 저장소가 구성되지 않았거나, 테스트 사용자가 역할에 속하지 않거나, 역할 이름에 공백이 포함되는 경우 테스트가 실패합니다.

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

리소스 키트 설치

이 장에 포함된 항목:

- [MDM Hub 샘플 연산 참조 저장소 설정, 104](#)
- [Informatica MDM Hub 샘플 연산 참조 저장소 등록, 107](#)
- [그래픽 모드에서 리소스 키트 설치, 109](#)
- [콘솔 모드에서 리소스 키트 설치, 111](#)
- [자동 모드에서 리소스 키트 설치, 113](#)

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

미리 알림 창이 표시됩니다.

- b. 선행 조건을 수행했는지 확인하고 **확인**을 클릭합니다.

WebSphere 보안 선택 창이 표시됩니다.

- c. WebSphere 보안 활성화 여부를 선택하고 **다음**을 클릭합니다.

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

WebSphere 응용 프로그램 서버의 서버 이름과 RMI 및 SOAP 포트를 설정합니다.

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

Informatica MDM Hub 서버 창이 표시됩니다.

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.

- 응용 프로그램 서버를 시작합니다.
- 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 쉼표로 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. WebSphere 설정을 구성합니다.
 - a. 응용 프로그램 서버 설치 디렉터리를 지정한 다음 **Enter** 키를 누릅니다.
 설치 프로그램에 JDBC 드라이버에 대한 WebSphere 선행 조건이 표시됩니다.
 - b. 데이터베이스 JDBC 드라이버 파일 위치를 확인하고 **Enter** 키를 누릅니다.
 데이터베이스 JDBC 드라이버 파일이 `<WebSphere_install_dir>/AppServer/lib` 디렉터리에 복사됩니다.
 WebSphere 보안 선택 정보가 표시됩니다.

- c. **아니요**를 선택하는 경우 WebSphere 응용 프로그램 서버 포트 정보가 표시됩니다. **예**를 선택하는 경우 WebSphere 응용 프로그램 서버 포트 및 사용자 자격 증명 정보가 표시됩니다.
- **아니요**를 선택한 경우 서버 이름, RMI 포트, SOAP 포트 및 프로필 이름을 입력하거나 기본값을 그대로 사용하고 **Enter** 키를 누릅니다.
 - **예**를 선택한 경우 서버 이름, RMI 포트, SOAP 포트, 프로필 이름, 사용자 이름 및 암호를 입력하거나 기본값을 그대로 사용하고 **Enter** 키를 누릅니다.
- 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 일부분으로 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_sample.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

속성 이름	설명
SIP.INSTALL.JASPERSOFT	Jaspersoft 보고 도구를 설치할지 여부를 지정합니다. 다음 값 중 하나를 지정합니다. - 0. BPM SDK를 설치하지 않음 - 1. SIF SDK 설치
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:\\ \<infadm_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 설정합 니다. 샘플을 구성하려면 응용 프로그램 서버 및 Hub 서버가 시작 되었고, 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 설정합니 다. RUN_CONFIGURE_SETUP 속성을 1로 설정하면 이후에 샘플 을 구성할 수도, 배포할 수도 없습니다.
SIP.AS.CHOICE	응용 프로그램 서버의 이름입니다. WebSphere를 지정합니다.
SIP.AS.HOME	WebSphere 설치 디렉터리의 경로입니다.
SIP.AS.SERVER	서버의 이름입니다.
SIP.AS.PROFILENAME	응용 프로그램 서버 프로필 이름입니다.
SIP.AS.PORT_2	RMI 포트 번호를 지정합니다.
SIP.AS.PORT_3	SOAP 포트 번호를 지정합니다.
SIP.WEBSPPHERE.SECURITY.ENABLED.Yes=1	WebSphere 보안이 활성화되어 있는 경우 이 속성을 설정합니다. SIP.WEBSPPHERE.SECURITY.ENABLED.Yes=1인 경우에는 다음 속성을 설정합니다. - SIP.APPSERVER.USERNAME - SIP.APPSERVER.PASSWORD
SIP.APPSERVER.USERNAME	WebSphere에 액세스하는 데 필요한 사용자 이름입니다.
SIP.APPSERVER.PASSWORD	WebSphere에 액세스하는 데 필요한 암호입니다.
SIP.SERVER.NAME	Hub 서버가 배포된 서버의 이름입니다.

속성 이름	설명
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 12 장

리소스 키트 설치 후 태스크

이 장에 포함된 항목:

- [sip_ant 스크립트 편집, 117](#)
- [수동으로 postInstall 스크립트 실행, 117](#)

sip_ant 스크립트 편집

설치 태스크를 수행한 후 sip_ant 스크립트를 편집합니다.

1. sip_ant 스크립트를 텍스트 편집기에서 엽니다.
sip_ant 스크립트는 다음 디렉터리에 있습니다.
UNIX의 경우. <리소스 키트 설치 디렉터리>/deploy/bin
Windows의 경우. <리소스 키트 설치 디렉터리>\deploy\bin
2. 다음과 유사한 행을 찾습니다.
UNIX의 경우. "\$JAVA_HOME/bin/java" \$USER_INSTALL_PROP -Xms128m -Xmx1024m -classpath "%WAS_CLASSPATH%";
Windows의 경우. "%JAVA_HOME%\bin\java" %USER_INSTALL_PROP% -Xms128m -Xmx1024m -classpath "%WAS_CLASSPATH%";
3. 다음과 유사한 코드로 바꿔서 JAVA_HOME을 설정합니다.
UNIX의 경우. "\$JAVA_HOME/bin/java" -Djava.endorsed.dirs="<WebSphere 설치 디렉터리>\endorsed_apis" \$USER_INSTALL_PROP -Xms128m -Xmx1024m -classpath "%WAS_CLASSPATH%";
Windows의 경우. "%JAVA_HOME%\bin\java" -Djava.endorsed.dirs="<WebSphere 설치 디렉터리>\endorsed_apis" %USER_INSTALL_PROP% -Xms128m -Xmx1024m -classpath "%WAS_CLASSPATH%";
4. 변경 내용을 저장하고 sip_ant 스크립트를 닫습니다.

수동으로 postInstall 스크립트 실행

모든 필수 배포 파일이 응용 프로그램 서버에 배포되었는지 확인하려면 postInstall 스크립트를 수동으로 실행합니다.

1. 명령 프롬프트를 엽니다.
2. 다음 디렉터리에서 postInstallSetup 스크립트로 이동합니다.

UNIX의 경우. <MDM Hub 설치 디렉터리>/hub/resourcekit/deploy

Windows의 경우. <MDM Hub 설치 디렉터리>\hub\resourcekit\deploy

3. 다음 명령을 실행합니다.

UNIX의 경우. `postInstall.sh -Ddatabase.password=<MDM Hub 마스터 데이터베이스 암호>`

참고: 암호에 느낌표(!)를 포함시킨 경우 느낌표 앞에 백슬래시를 포함시켜야 합니다. 예를 들어 암호가 !!cmx!!인 경우 암호를 \!!cmx\!! 같이 입력합니다.

Windows의 경우. `postInstall.bat -Ddatabase.password=<MDM Hub 마스터 데이터베이스 암호>`

참고: WebSphere에서 보안을 활성화한 경우 `-Dwebsphere.password=<Secure WebSphere 암호>` 옵션을 지정하여 `postInstallSetup` 스크립트를 실행합니다.

제 13 장

MDM Hub 문제 해결

이 장에 포함된 항목:

- [설치 프로세스 문제 해결, 119](#)

설치 프로세스 문제 해결

설치가 실패하는 경우 다음 정보를 사용하여 실패 문제를 해결합니다.

Hub 콘솔을 실행할 수 없음

설치 후 Hub 콘솔이 실행되지 않고 로그 파일에 다음 오류 메시지가 생성됩니다.

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

MDM Hub 마스터 데이터베이스 암호 또는 연산 참조 저장소 암호를 암호화하고 업데이트합니다.

1. 데이터베이스 스키마 암호를 암호화하려면 명령 프롬프트에서 다음 명령을 실행합니다.

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

결과가 터미널 창에 출력됩니다.

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

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

3. `postInstallSetup` 스크립트를 실행합니다.

기본 디렉터리에 응용 프로그램 서버 프로파일 설치되어 있지 않음

기본 디렉터리에 응용 프로그램 서버 프로파일을 설치하지 않은 경우 `postInstallSetup` 스크립트가 Hub 서버 및 처리 서버 응용 프로그램을 응용 프로그램 서버에 배포하지 못합니다.

다음 기본 디렉터리를 사용합니다.

UNIX의 경우. `<WebSphere_install_home>/profiles`

Windows의 경우. `<WebSphere_install_home>\profiles`

이 문제를 해결하려면 Hub 서버 및 처리 서버 EAR 파일을 다시 패키징한 다음, 응용 프로그램 서버를 설치한 사용자 지정 디렉터리에 수동으로 Hub 서버 및 처리 서버 응용 프로그램을 배포합니다.

디렉터리에 siperian-mrm.ear 파일이 포함되어 있어 PostInstallSetup 스크립트가 실패함

이름이 siperian-mrm.ear인 파일이 포함되어 있는 디렉터리에 Hub 서버 응용 프로그램을 배포하려고 하면 다음과 같은 오류가 표시됩니다.

```
[wsadmin] ADMA5016I: Installation of siperian-mrm.ear started.
```

```
[wsadmin] A composition unit with name siperian-mrm.ear already exists. Select a different application name.
```

이 문제를 해결하려면 siperian-mrm.ear 파일이 포함되어 있는 모든 디렉터리를 제거한 다음 postInstallSetup을 다시 실행하여 EAR 파일을 배포합니다.

참고: Hub 서버 응용 프로그램을 배포 해제해도 siperian-mrm.ear 파일이 여전히 응용 프로그램 서버 디렉터리에 존재할 수 있습니다.

처리 시간 초과로 인해 PostInstallSetup 스크립트가 실패함

WebSphere 환경에서 Hub 서버를 설치하는 경우 설치 후 설정 프로세스가 실패하고 다음 오류가 표시됩니다.

```
[wsadmin] Starting siperian-mrm.ear ...
[wsadmin] WASX7017E: Exception received while running file "wsinstall.jacl"; exception information:
com.ibm.websphere.management.exception.ConnectorException
[wsadmin] org.apache.soap.SOAPException: [SOAPException: faultCode=SOAP-ENV:Client; msg=Read timed out;
targetException=java.net.SocketTimeoutException: Read timed out]
```

이 문제는 SOAP 요청 시간이 초과될 때 발생합니다.

이 문제를 해결하려면 다음 단계를 수행합니다.

1. 다음 디렉터리로 이동합니다.
<Websphere 프로파일 루트 디렉터리>/properties
2. soap.client.props 파일에서 com.ibm.SOAP.requestTimeout 속성의 값을 늘립니다.
3. WebSphere를 다시 시작하고 postInstallSetup 스크립트를 다시 실행합니다.

MDM Hub 사용자가 로그인할 수 없음

Hub 서버를 설치한 후에 CMX_SYSTEM 스키마를 다시 생성하면 MDM Hub가 해시된 암호를 인식할 수 없습니다. 따라서 사용자가 MDM Hub에 로그인할 수 없습니다.

이 문제를 해결하려면 postInstallSetup 스크립트를 수동으로 다시 실행합니다. 이 스크립트는 MDM Hub 사용자의 암호가 다시 해시되고 사용자가 로그인할 수 있도록 합니다.

또는 postInstallSetup 스크립트를 다시 실행하고 싶지 않은 경우 다음 명령을 실행하여 사용자 암호를 해시된 암호로 마이그레이션한 후 응용 프로그램 사용자를 생성합니다.

UNIX의 경우.

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

Windows의 경우.

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

참고: WebSphere 환경에서 MDM Hub 사용자는 다음 디렉터리에 대한 액세스 권한 및 쓰기 권한이 있어야 합니다.

```
<MDM Hub 설치 디렉터리>/server/bin/resources/certificates
```

ActiveVOS 서버 배포 시간 초과로 인해 PostInstallSetup 스크립트에서 오류가 발생함

Hub 서버를 설치하면 ActiveVOS 서버 배포를 시도한 후에 설치 후 설정 절차에서 오류가 발생할 수도 있습니다.

문제를 해결하려면 다음 디렉터리의 `build.properties` 파일에서 `deploy.wait.time` 속성 값을 늘립니다.

UNIX의 경우. <infadm 설치 디렉터리>/hub/server/bin

Windows의 경우. <infadm 설치 디렉터리>\hub\server\bin

연산 참조 저장소의 시작 또는 중지 타임스탬프가 잘못되었음

생성한 연산 참조 저장소를 내보낼 때 MDM Hub에서는 .dmp 파일을 생성합니다. 연산 참조 저장소의 .dmp 파일을 재사용하여 두 번째 연산 참조 저장소를 생성할 수 있습니다. 연산 참조 저장소의 C_REPOS_DB_VERSION 테이블에 있는 시작 날짜 및 종료 날짜 열에 잘못된 시작 날짜 및 종료 날짜가 포함됩니다. 이 오류는 기능에 영향을 주지 않기 때문에 무시할 수 있습니다.

Hub 서버가 cmx_system 스키마에 연결할 수 없음

Hub 서버가 cmx_system 스키마에 연결할 수 없는지 확인하려면 응용 프로그램 서버 로그를 검토합니다.

예를 들어 다음 데이터베이스에 액세스할 수 없음을 나타내는 다음 오류가 표시될 수 있습니다.

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

이 문제를 해결하려면 데이터베이스 연결 문제를 해결합니다. 응용 프로그램 서버 콘솔을 사용하여 데이터베이스에 대한 연결을 테스트합니다. cmx_system 스키마에 대한 연결을 확인할 수 없는 경우 cmx_system 스키마를 다시 생성합니다.

레코드를 토큰화할 필요성을 확인하지 못함

일치 프로세스를 실행하면 다음과 같은 오류가 표시될 수 있습니다.

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

다음 환경 변수 설정을 확인합니다.

- 라이브러리 경로 환경 변수가 다음 경로를 포함해야 합니다.

UNIX의 경우. <infadm_install_directory>/hub/cleanse/lib

Windows의 경우. <infadm_install_directory>\hub\cleanse\lib

라이브러리 경로 환경 변수는 운영 체제에 따라 다릅니다.

- AIX. LIBPATH

- Suse 또는 RedHat Linux. LD_LIBRARY_PATH

- Windows. PATH

- SSAPR 환경 변수는 모든 사용자에게 다음 경로를 포함해야 합니다.

UNIX의 경우. <infadm_install_directory>/server_install_dir/cleanse/resources

Windows의 경우. <infadm_install_directory>\server_install_dir\cleanse\resources

처리 서버를 로드할 때 major.minor 버전 오류가 발생함

처리 서버를 로드하려고 할 때 여러 major.minor 콘솔 오류가 표시되는 경우 시스템에 올바른 Java 버전이 설치되어 있는지 확인합니다.

CORBA TRANSACTION_ROLLEDBACK 예외

데이터 관리자 또는 병합 관리자를 사용하는 경우 MDM Hub에서 CORBA TRANSACTION_ROLLEDBACK 예외를 생성할 수 있습니다. 이 예외가 생성된 경우, WebSphere 관리 콘솔을 열어 WebSphere 처리 서버 정의에 -

Djava.vendor=IBM Java 옵션을 수동으로 설정합니다.

인증하는 동안 Informatica Address Verification 예외가 발생함

인증하는 동안 Informatica Address Verification에서 예외가 발생합니다. JVM의 스택 크기가 충분하지 확인합니다.

1. WebSphere 콘솔을 엽니다.
2. 서버 > 응용 프로그램 서버 > <사용자의 서버> > 프로세스 정의 > **Java Virtual Machine**으로 이동합니다.
3. 일반 JVM 인수에 다음 인수를 추가합니다.
 - Xss2000k - Initializes the stack size to 2000k
 - Xms128m - Initializes the heap with at least 128 MB
 - Xmx1024m - Initializes the heap with a maximum of 1024 MB
4. 구성을 저장합니다.
5. WebSphere 서버를 다시 시작합니다.

연산 참조 저장소에 워크플로우 엔진이 구성되어 있지 않습니다.

MDM Hub을 설치한 다음 이전 버전에서 연산 참조 저장소(ORS)를 가져오면 ORS에 구성된 워크플로우 엔진이 없음을 표시하는 치명적인 오류가 표시됩니다. 이 오류는 Siperian BPM 워크플로우 엔진이 기본적으로 등록되어 있지 않기 때문에 발생합니다. 워크플로우 관리자를 사용하여 ORS에서 찾을 것으로 예상되는 이름으로 Siperian BPM 워크플로우 엔진을 등록합니다.

처리 서버 .ear 파일을 배포한 후 오류가 발생함

WebSphere 8.5.5.9가 있는 Linux 환경에서 처리 서버를 배포한 후 다음 오류가 발생합니다.

Too many open files. Unable to start cleanse ear.

Linux에서 ulimit 매개 변수의 값을 늘리고 처리 서버를 배포합니다.

제 14 장

제거

이 장에 포함된 항목:

- [제거 개요, 123](#)
- [Hub 저장소 제거, 123](#)
- [그래픽 모드에서 처리 서버 제거, 124](#)
- [그래픽 모드에서 Hub 서버 제거, 125](#)
- [그래픽 모드에서 리소스 키트 제거, 125](#)
- [콘솔 모드에서 처리 서버 제거, 126](#)
- [콘솔 모드에서 Hub 서버 제거, 126](#)
- [콘솔 모드에서 리소스 키트 제거, 127](#)
- [수동으로 처리 서버 배포 해제, 127](#)
- [수동으로 Hub 서버 배포 해제, 127](#)

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.
`<infadm_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. **시작**을 클릭한 다음 **프로그램 > Infadm > 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.
`<infadm_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

WebSphere 환경에서 처리 서버를 수동으로 배포 해제해야 할 수 있습니다.

- ▶ WebSphere 응용 프로그램 서버 관리 콘솔을 사용하여 `siperian-mrmcleanse.ear` 파일을 수동으로 배포 해제합니다.

자세한 내용은 WebSphere 설명서를 참조하십시오.

수동으로 Hub 서버 배포 해제

WebSphere 환경에서 Hub 서버를 수동으로 배포 해제해야 할 수 있습니다.

- ▶ WebSphere 응용 프로그램 서버 관리 콘솔을 사용하여 다음 배포 파일의 배포를 해제합니다.

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

자세한 내용은 WebSphere 설명서를 참조하십시오.

인덱스

A

ActiveVOS

- URN, 설정 [100](#)
- 배포 [94](#), [97](#)
- 설치 [94](#)
- 설치 파일 [96](#)
- 시간 관리자 구성 [95](#)
- 작업 관리자 구성 [94](#)

ActiveVOS Central

- 설치 [96](#)

ActiveVOS 서버

- 설치 [96](#)

ActiveVOS 콘솔 관리자

- abAdmin 역할 [26](#)
- 생성 [26](#)

ActiveVOS 파일

- deployer.xml [96](#)
- install.properties [96](#)

H

HTTPS

- 처리 서버용 [90](#)

Hub 서버

- EAR 파일 다시 패키지 [65](#)
- postInstallSetup 스크립트 [58](#)
- WebSphere 클러스터에 설치 [51](#)
- 마법사로 설치 [49](#)
- 명령줄에서 설치 [49](#)
- 배포 [57](#)
- 배포 스크립트 [57](#)
- 버전 정보 [53](#)
- 빌드 번호 [53](#)
- 사용자 지정 JAR 파일 다시 패키지 [65](#)
- 설치 [109](#)
- 설치 로그 파일 [53](#)
- 수동 배포 [57](#), [59](#)
- 자동 설치 [113](#)
- 자동으로 설치 [50](#)
- 재배포 [55](#)
- 제거 [126](#), [127](#)
- Hub 서버 속성 파일
- 구성 [32](#)
- Hub 저장소
- 제거 [123](#)
- 데이터스페이스, 생성 [16](#)
- Hub 콘솔
- 시작 [67](#), [71](#)
- Hub 콘솔 클라이언트
- build.properties
- 구성 [54](#)
- 구성 [54](#)
- 다중 노드 환경 [54](#)

- Hub 콘솔 클라이언트 (계속)
- 클러스터 환경 [54](#)

I

Identity Resolution

- 배포 [97](#)

Infinispan

- 구성 [69](#), [70](#)

Informatica ActiveVOS

- 스키마 생성 [19](#)

Informatica 플랫폼

- 속성 파일 [32](#)

J

JAAS 응용 프로그램 로그인

- ActiveVOS용 [95](#)
- 구성 [95](#)

Java 옵션

- 구성 [21](#)

JMS 메시지 대기열

- 구성 [63](#), [67](#)

JVM 매개 변수

- 구성 [21](#)

M

MDM Hub

- JDK(Java 개발 키트) 요구 사항 [14](#)
- 구성 요소 [10](#)
- 설치 디자인 [12](#)
- 설치 태스크 [12](#)
- 설치 토폴로지 [12](#)
- 소개 [10](#)
- 시스템 요구 사항 [14](#)
- 운영 체제 로컬 설정 [14](#)
- 환경 변수 설정 [14](#)

MDM Hub EAR 파일

- 다시 패키지 [65](#)

N

대상 데이터베이스

- 선택 [71](#)

데이터베이스

- 대상 데이터베이스 [71](#)

- 연결 테스트 [16](#)

로그 파일

- Hub 서버 로그 파일 [53](#)
- JBoss 로그 파일 [53](#), [82](#)

로그 파일 (계속)

- 디버그 로그 파일 [53, 82](#)
- 설치 로그 파일 [53, 82](#)
- 설치 선행 조건 로그 파일 [53, 82](#)
- 설치 후 설정 로그 파일 [53, 82](#)
- 처리 서버 로그 파일 [82](#)

리소스 키트

- 설치 [109, 111](#)
- 자동 속성 파일 [114](#)
- 제거 [126](#)

마스터 데이터베이스

- 메타데이터 가져오기 [36](#)
- 생성 [33](#)

문제 해결

- 설치 후 프로세스 [119](#)

보안 통신

- 활성화, 처리 서버 [90](#)

샘플 스카마

- 등록 [107](#)
- 설치 [104](#)

설치

Hub 서버

- 마법사 [49](#)
- 명령줄 [49](#)
- 자동으로 [50](#)

리소스 키트 [109, 111](#)

처리 서버

- 마법사 [78](#)
- 명령줄 [79](#)
- 자동으로 [80](#)

설치 프로그램

- 워크플로우 [43, 75](#)

연산 참조 저장소

- 등록 [72](#)
- 메타데이터 가져오기 [38](#)
- 생성 [35](#)

워크플로우 엔진

- 추가 [101](#)

일치 인구집단

- 활성화 [92](#)

일치 채우기

- 활성화 [92](#)

제거

- Hub 서버 [126, 127](#)
- Hub 저장소 [123](#)
- 처리 서버 [126](#)

처리 서버

- HTTPS, 활성화 [90](#)
- postInstallSetup 스크립트 [89](#)
- WebSphere 클러스터에 배포 [80](#)
- 데이터 소스 생성 [85](#)
- 마법사로 설치 [78](#)
- 명령줄에서 설치 [79](#)
- 배포 [85, 88, 89](#)
- 배포 스크립트 [85, 88](#)
- 버전 정보 [83](#)

처리 서버 (계속)

- 빌드 번호 [83](#)
- 설치 로그 파일 [82](#)
- 수동 배포 [85, 88](#)
- 자동으로 설치 [80](#)
- 재배포 [84, 89](#)
- 제거 [126](#)
- 처리 서버 속성 파일
- 구성 [32](#)
- 테이블스페이스
- 생성 [16](#)

O

Oracle 구성

- init.ora 매개 변수 설정 [16](#)
- MDM Hub용 [16](#)
- Oracle 휴지통 비활성화 [16](#)

P

postinstallsetup 스크립트

- 실행 [89](#)

postInstallSetup 스크립트

- Hub 서버용 [58](#)
- 실행 [58](#)
- 처리 서버용 [89](#)

T

TLS

- 구성 [24](#)
- TNS 이름
- 추가 [16](#)

U

URN

- ActiveVOS 설정 [100](#)

W

WebSphere

- 구성 [20](#)
- 설정 [56](#)

WebSphere 구성

- Informatica Data Director [31](#)

WebSphere 클러스터

- Hub 서버 설치 [51](#)
- 처리 서버 배포 [80](#)