



Informatica®
10.4.1

Developer tool 가이드

Informatica Developer tool 가이드

10.4.1

2020년6월

© 저작권 Informatica LLC 2009, 2022

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.

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

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.

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: 2022-05-05

목차

서문	13
Informatica 리소스	13
Informatica 네트워크	13
Informatica 기술 자료	13
Informatica 설명서	13
Informatica Product Availability Matrix	14
Informatica Velocity	14
Informatica Marketplace	14
Informatica 글로벌 고객 지원 센터	14
장 1: Informatica Developer	15
Informatica Developer 개요	15
Informatica Data Quality 및 프로파일링	15
Informatica Data Services	17
Informatica Developer 시작	18
로컬 시스템에서 Developer 도구 시작	18
원격 시스템에서 Developer 도구 시작	18
Informatica Developer 인터페이스	19
Informatica Developer의 보기	19
Informatica Developer 시작 페이지	21
치트 시트	21
Informatica Developer 온라인 도움말	21
Informatica 기본 설정	21
Informatica Marketplace	22
Informatica Developer 설정	22
1단계. 도메인 추가	22
2단계. 모델 리포지토리 추가	23
3단계. 기본 데이터 통합 서비스 선택	23
4단계. 기본 메타데이터 액세스 서비스 선택	23
도메인	24
프로젝트	24
프로젝트 작성	25
프로젝트 필터링	25
프로젝트 사용 권한	25
외부 개체에 대한 사용 권한	26
종속 개체 인스턴스에 대한 사용 권한	26
상위 개체 액세스	27
사용 권한 할당	27
폴더	28

폴더 작성.....	28
개체 작업 복사.....	29
개체 복사.....	29
개체 사본 저장.....	30
태그.....	30
태그 작성.....	30
태그 할당.....	31
태그 보기.....	31

장 2: 모델 리포지토리..... 32

모델 리포지토리 개요.....	32
Informatica Developer의 개체.....	33
개체 속성.....	35
리포지토리 개체 잠금.....	35
잠금 관리.....	35
잠금 관리 규칙 및 지침.....	36
버전 지정 개체를 사용하는 팀 기반 개발.....	36
버전 지정 개체 관리.....	37
개체의 기록 버전.....	39
체크 아웃된 개체 보기.....	39
버전 기록 보기.....	39
팀 기반 개발 문제 해결.....	40
모델 리포지토리에 연결.....	40
모델 리포지토리 서비스 새로 고침.....	40

장 3: Informatica Developer에서 검색..... 41

Informatica Developer에서 검색 개요.....	41
도메인 검색.....	41
개체 및 속성 검색.....	42
Business Glossary 검색.....	43
Business Glossary Desktop 조회.....	43
비즈니스 용어 조회.....	43
비즈니스 용어 조회를 위한 바로 가기 키 사용자 지정.....	43
편집기 검색.....	44

장 4: 연결..... 46

연결 개요.....	46
연결 유형.....	47
연결 탐색기 보기.....	48
연결 관리.....	48
연결 작성.....	49
연결 편집.....	49

연결 복사.....	50
연결 삭제.....	50
연결 목록 새로 고침.....	51
연결 전환.....	51
연결을 전환하기 전에.....	51
연결 전환.....	52
연결을 전환한 후에.....	53
타사 JDBC 드라이버.....	54
환경 SQL.....	54
연결 환경 SQL.....	55
트랜잭션 환경 SQL.....	55
환경 SQL 구성 지침.....	55

장 5: 실제 데이터 개체..... 56

실제 데이터 개체 개요.....	56
실제 데이터 개체 유형.....	57
관계형 데이터 개체.....	57
관계형 데이터 개체 가져오기.....	58
키 관계.....	59
사용자 지정된 데이터 개체.....	60
키 관계.....	61
사용자 지정된 데이터 개체 쓰기 속성.....	62
사용자 지정된 데이터 개체 작성.....	63
사용자 지정된 데이터 개체에 관계형 리소스 추가.....	64
사용자 지정된 데이터 개체에 관계형 데이터 개체 추가.....	64
사용자 지정된 데이터 개체에 키 작성.....	64
사용자 지정된 데이터 개체 내 관계 작성.....	65
대상 테이블 작성 또는 대체.....	65
대상 테이블 작성 또는 바꾸기에 대한 규칙 및 지침.....	66
디자인 타임에 DDL 생성 및 실행.....	66
런타임 시 DDL 생성 및 실행.....	67
DDL 생성 오류.....	67
사용자 지정 쿼리.....	68
사용자 지정 쿼리 최적화.....	68
기본 쿼리.....	68
힌트.....	69
고유 항목 선택.....	71
필터.....	71
정렬된 포트.....	72
사용자 정의 조인.....	73
외부 조인 지원.....	74
Informatica 조인 구문.....	74

PreSQL 및 PostgreSQL 명령.	78
사용자 지정 쿼리 작성.	79
비관계형 데이터 개체.	79
비관계형 데이터 개체 가져오기.	80
관계형 데이터 작업에서 읽기, 쓰기 또는 조회 변환 작성.	80
WSDL 데이터 개체.	81
WSDL 데이터 개체 개요 보기.	81
WSDL 데이터 개체 고급 보기.	82
WSDL 데이터 개체 가져오기.	82
WSDL 동기화.	82
인증서 관리.	83
동기화.	84
Informatica Developer의 플랫폼 파일 데이터 개체 동기화.	84
Informatica Developer의 관계형 데이터 개체 동기화.	85
실제 데이터 개체 문제 해결.	86
장 6: 플랫폼 파일 데이터 개체.	87
플랫폼 파일 데이터 개체 개요.	87
소스 파일 이름 생성.	88
플랫폼 파일 데이터 개체 개요 속성.	88
플랫폼 파일 데이터 개체 고급 속성.	89
형식 속성.	90
열 형식: 구분자로 분리된 속성.	90
열 형식: 고정 너비 속성.	91
런타임: 읽기 속성.	92
런타임: 쓰기 속성.	94
제어 파일.	95
런타임에 열 업데이트.	96
자동으로 런타임 열 이름 생성.	96
데이터 파일 헤더에서 런타임 열 이름 생성.	97
제어 파일에서 열 메타데이터 생성.	97
제어 파일 형식.	98
런타임 속성 매개 변수화.	98
제어 파일의 런타임 처리.	98
제어 파일에 대한 규칙 및 지침.	99
Excel에서 플랫폼 파일 데이터 개체로 복사.	99
Excel에서 플랫폼 파일 데이터 개체 편집.	99
메타데이터를 플랫폼 파일 데이터 개체에 복사.	100
예: Excel에서 데이터 개체 편집.	100
플랫폼 파일 데이터 개체 작성.	101
비어 있는 플랫폼 파일 데이터 개체 작성.	101
기존 플랫폼 파일에서 플랫폼 파일 데이터 개체 작성.	102

제어 파일에서 플랫폼 파일 데이터 개체 작성.	103
장 7: 데이터의 논리적 보기.	105
데이터의 논리적 보기 개요.	105
논리적 데이터 개체 모델 예제.	106
데이터의 논리적 보기 개발.	106
논리적 데이터 개체 모델.	107
논리적 데이터 개체 모델 작성.	107
모델링 도구에서 논리적 데이터 개체 모델 가져오기.	108
논리적 데이터 개체 모델 속성.	108
CA ERwin Data Modeler 가져오기 속성.	109
IBM Cognos Business Intelligence Reporting - Framework Manager 가져오기 속성.	110
SAP BusinessObjects Designer 가져오기 속성.	111
SAP PowerDesigner CDM 가져오기 속성.	112
SAP PowerDesigner PDM 가져오기 속성.	113
XSD 가져오기 속성.	113
논리적 데이터 개체.	114
논리적 데이터 개체 속성.	114
특성 관계.	115
논리적 데이터 개체 작성.	116
논리적 데이터 개체 매핑.	118
논리적 데이터 개체 읽기 매핑.	119
논리적 데이터 개체 쓰기 매핑.	119
논리적 데이터 개체 매핑 작성.	119
장 8: 데이터 보기.	121
데이터 보기 개요.	121
구성.	122
구성 속성.	122
데이터 뷰어 구성.	125
매핑 구성.	126
웹 서비스 구성.	127
기본 구성 속성 업데이트.	128
구성 문제 해결.	128
데이터 내보내기.	128
개체 종속성.	129
개체 종속성 보기.	129
개체 종속성 보기.	129
개체 종속성 필터링.	130
로그.	130
로그 파일 형식.	131
유효성 검사 기본 설정.	131

오류 메시지 그룹화.	131
오류 메시지 제한.	131
Developer tool에서 작업 모니터링.	132

장 9: 응용 프로그램 배포..... 133

응용 프로그램 배포 개요.	133
응용 프로그램 작성.	134
전체 응용 프로그램.	134
증분 응용 프로그램.	134
응용 프로그램 속성.	135
응용 프로그램에 배포.	137
보관 파일에 배포.	138
리소스 매개 변수를 사용한 배포.	139
런타임 개체 보기.	140
응용 프로그램 재배포.	140
응용 프로그램 상태 정보.	141
응용 프로그램 작성, 배포 및 업데이트 방법.	141
응용 프로그램 작성.	142
응용 프로그램 배포.	143
개체 배포.	144
응용 프로그램 보관 가져오기.	145
런타임 응용 프로그램에 연결.	146
응용 프로그램 편집.	146
데이터 통합 서비스에 응용 프로그램 재배포.	146

장 10: 응용 프로그램 패치 배포..... 148

응용 프로그램 패치 배포 개요.	148
응용 프로그램 개체 인스턴스.	149
Application Object Dependencies.	149
증분 배포 마법사.	151
응용 프로그램 선택.	152
소스 개체 선택.	152
패치 배포 전략.	153
응용 프로그램 패치 유형 선택.	153
직접 종속성만 상속.	154
직접 및 간접 종속성 상속.	155
직접, 간접 및 원격 종속성 상속.	155
응용 프로그램 개체 영향 결정.	156
개체 영향 미리 보기.	157
영향을 받는 개체 식별.	157
개체 영향의 범위 식별.	159
패치 배포 전략 지정.	160

패치 배포 전략 속성.	160
응용 프로그램 패치 배포.	161
응용 프로그램 패치 배포의 규칙 및 지침.	162

장 11: 응용 프로그램 패치 배포 예..... 163

응용 프로그램 패치 배포 예 개요.	163
패치 유형: 직접.	164
패치 유형: 직접 및 간접.	165
패치 유형: 직접, 간접 및 원격.	166
결론.	167

장 12: CI/CD(지속적인 통합 및 지속적인 전달)..... 169

CI/CD 개요.	169
REST API 또는 infacmd 사용.	170
지속적인 통합.	170
개체 배포.	170
개체 할당.	171
개체 충돌 해결.	171
지속적인 전달.	171
자동화된 통합 프로세스 샘플.	172
1단계. 배포를 위해 매핑 준비.	172
2단계. 데이터 통합 서비스에 매핑 배포.	172
3단계. 온전성 테스트 실행.	173
4단계. 편집을 위해 매핑 준비.	173
자동화된 전달 프로세스 샘플.	174
일반적인 통합 및 전달 작업 참조.	174
REST API를 사용하여 응용 프로그램 배포 및 재배포.	177

장 13: 개체 가져오기 및 내보내기..... 178

개체 가져오기 및 내보내기 개요.	178
개체 가져오기 및 내보내기.	179
개체 내보내기.	180
개체 내보내기.	180
개체 가져오기.	181
프로젝트 가져오기.	181
개체 가져오기.	183
이전 Informatica 릴리스에서 개체 가져오기.	184

부록 A: 데이터 유형 참조..... 185

데이터 유형 참조 개요.	185
변환 데이터 유형.	186
Integer 데이터 유형.	187

Binary 데이터 유형.	189
Date/Time 데이터 유형.	189
Decimal 및 Double 데이터 유형.	191
String 데이터 유형.	193
복합 데이터 유형.	193
복합 파일 및 변환 데이터 유형.	197
Avro 데이터 유형 및 변환 데이터 유형.	197
JSON 데이터 유형 및 변환 데이터 유형.	199
ORC 데이터 유형 및 변환 데이터 유형.	200
Parquet 데이터 유형 및 변환 데이터 유형.	201
Rules and Guidelines for Data Types.	202
플랫 파일 및 변환 데이터 유형.	204
DB2 for LUW 및 변환 데이터 유형.	205
DB2 for i5/OS, DB2 for z/OS 및 변환 데이터 유형.	206
지원되지 않는 DB2 for i5/OS 및 DB2 for z/OS 데이터 유형.	207
JDBC 및 변환 데이터 유형.	207
Microsoft SQL Server 및 변환 데이터 유형.	209
Uniqueidentifier 데이터 유형.	211
비관계형 및 변환 데이터 유형.	211
ODBC 및 변환 데이터 유형.	213
Oracle 및 변환 데이터 유형.	215
숫자(P,S) 데이터 유형.	216
Char, Varchar, Clob 데이터 유형.	217
지원되지 않는 Oracle 데이터 유형.	217
SAP HANA 및 변환 데이터 유형.	217
XML 및 변환 데이터 유형.	219
데이터 변환.	221
포트 간 데이터 변환.	221
부록 B: 키보드 바로 가기.	223
고급 매핑 옵션에 대한 키보드 바로 가기.	223
개체 키보드 바로 가기.	224
포트 키보드 바로 가기.	225
변환 색상표 키보드 바로 가기.	226
작업 영역 키보드 바로 가기.	226
부록 C: 연결 속성.	228
연결 속성 개요.	229
Adabas 연결 속성.	230
Amazon Redshift 연결 속성.	232
Amazon S3 연결 속성.	234
블록체인 연결 속성.	236

Cassandra 연결 속성.....	237
DataSift 연결 속성.....	238
Databricks 연결 속성.....	239
Facebook 연결 속성.....	241
Greenplum 연결 속성.....	242
Google Analytics 연결 속성.....	243
Google BigQuery 연결 속성.....	244
연결 모드.....	245
Google Cloud Spanner 연결 속성.....	245
Google Cloud Storage 연결 속성.....	246
Hadoop 연결 속성.....	247
Hadoop 클러스터 속성.....	247
공통 속성.....	249
거부 디렉터리 속성.....	250
Blaze 구성.....	251
Spark 구성.....	252
HBase 연결 속성.....	253
HDFS 연결 속성.....	253
MapR-DB용 HBase 연결 속성.....	255
Hive 연결 속성.....	255
HTTP 연결 속성.....	259
IBM DB2 연결 속성.....	260
IBM DB2 for i5/OS 연결 속성.....	263
IBM DB2 for z/OS 연결 속성.....	266
IMS 연결 속성.....	269
JDBC 연결 속성.....	271
JDBC V2 연결 속성.....	273
JD Edwards EnterpriseOne 연결 속성.....	275
LDAP 연결 속성.....	276
LinkedIn 연결 속성.....	277
Microsoft Azure Blob Storage 연결 속성.....	278
Microsoft Azure Cosmos DB SQL API 연결 속성.....	279
Microsoft Azure Data Lake Storage Gen1 연결 속성.....	280
Microsoft Azure Data Lake Storage Gen2 연결 속성.....	281
Microsoft Azure SQL 데이터 웨어하우스 연결 속성.....	282
MS SQL Server 연결 속성.....	283
Netezza 연결 속성.....	287
OData 연결 속성.....	287
ODBC 연결 속성.....	288
Oracle 연결 속성.....	290
Salesforce 연결 속성.....	292
Salesforce Marketing Cloud 연결 속성.....	293

SAP 연결 속성.....	294
순차적 연결 속성.....	297
Snowflake 연결 속성.....	299
Teradata Parallel Transporter 연결 속성.....	300
Tableau 연결 속성.....	302
Tableau V3 연결 속성.....	303
Twitter 연결 속성.....	304
Twitter 스트리밍 연결 속성.....	305
VSAM 연결 속성.....	306
웹 콘텐츠-Kapow Katalyst 연결 속성.....	308
웹 서비스 연결 속성.....	309
데이터베이스 연결의 식별자 속성.....	311
정규 식별자.....	311
구분자로 분리된 식별자.....	311
식별자 속성.....	312
인덱스.....	314

서문

Informatica® Developer tool 가이드를 사용하여 Developer tool의 사용자 인터페이스에 대해 알아보십시오.
모델 리포지토리에 연결하고, 연결 및 데이터 개체를 생성하고, 응용 프로그램을 배포하는 방법을 알아보십시오.

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

Informatica Developer

이 장에 포함된 항목:

- [Informatica Developer 개요, 15](#)
- [Informatica Developer 시작, 18](#)
- [Informatica Developer 인터페이스, 19](#)
- [Informatica Developer 설정, 22](#)
- [도메인, 24](#)
- [프로젝트, 24](#)
- [프로젝트 사용 권한, 25](#)
- [폴더, 28](#)
- [개체 작업 복사, 29](#)
- [태그, 30](#)

Informatica Developer 개요

Developer tool은 데이터 통합, 데이터 품질, 데이터 프로파일링 및 데이터 서비스 솔루션을 디자인하고 구현하는 데 사용되는 응용 프로그램입니다.

Developer tool을 사용하여 메타데이터를 가져오고 연결을 작성하고 데이터 개체를 작성할 수 있습니다. 또한 Developer tool을 사용하여 프로필, 매핑 및 워크플로우를 작성하고 실행할 수 있습니다.

Informatica Data Quality 및 프로파일링

Developer tool에서 데이터 품질 기능을 사용하여 데이터의 콘텐츠와 구조를 분석하고 비즈니스 요구에 맞춰 데이터를 개선할 수 있습니다.

Developer tool을 통해 프로세스를 디자인하고 실행하여 다음과 같은 태스크를 수행할 수 있습니다.

- 프로필 데이터. 프로파일링으로 데이터의 콘텐츠와 구조를 표시합니다. 프로파일링은 데이터의 강점과 단점을 파악하여 프로젝트 계획을 정립하는 데 도움이 될 수 있으므로 어떤 데이터 프로젝트에서도 핵심 단계로 꼽힙니다.
- 데이터 품질 검토를 위해 성과 기록표를 생성하십시오. 성과 기록표는 프로필의 품질 측정을 그래픽으로 표현합니다.
- 데이터 값 표준화. 프로필을 실행할 때 찾는 오류와 불일치를 제거하기 위해 데이터를 표준화하십시오. 구두점, 형식 및 맞춤법에서의 변형을 표준화할 수 있습니다. 예를 들어 도시, 도 및 우편 번호 값이 일관하도록 보장할 수 있습니다.

- 데이터 구문 분석. 구문 분석은 여러 값으로 구성된 필드를 읽고 포함된 정보의 유형에 따라 각 값에 대한 필드를 생성합니다. 또한 구문 분석은 정보를 레코드에 추가할 수 있습니다. 예를 들어 제품 데이터에 측정 단위를 추가하는 구문 분석 작업을 정의할 수 있습니다.
- 우편 주소 유효성 검사. 주소 유효성 검사는 우편 주소 데이터의 정확성 및 배달 가능성을 평가하고 향상시킵니다. 주소 유효성 검사는 주소 레코드를 전국 우편 회사의 주소 참조 데이터와 비교하여 주소의 오류를 수정하고 부분적 주소를 완성합니다. 주소 유효성 검사를 통해 메일 전송을 가속화하고 메일 비용을 감소시키는 우편 정보를 추가할 수도 있습니다.
- 중복 레코드 찾기. 중복 분석은 각 레코드에 있는 하나 이상의 필드에서 데이터를 비교하여 레코드 간의 유사점을 계산합니다. 분석할 필드를 선택하고 데이터에 적용할 비교 전략을 선택합니다. **Developer tool**에서는 두 가지 유형의 중복 분석을 사용할 수 있습니다. 하나는 유사 레코드 또는 중복 레코드를 식별하는 필드 일치이고 다른 하나는 레코드 데이터에서 유사하거나 중복된 ID를 식별하는 ID 일치입니다.
- 예외 관리. 예외란 수동으로 수정하는 데이터 품질 문제가 포함된 레코드를 말합니다. 다른 데이터 품질 프로세스를 실행한 후 데이터 집합에 남아 있는 예외 레코드를 캡처하는 매핑을 실행할 수 있습니다. **Analyst** 도구에서 예외 레코드를 검토하고 편집합니다.
- 참조 데이터 테이블 생성. **Informatica**는 표준화 및 구문 분석 등 여러 가지 유형의 데이터 품질 프로세스를 향상시킬 수 있는 참조 데이터를 제공합니다. 프로필 결과의 데이터를 사용하여 참조 테이블을 생성할 수 있습니다.
- 데이터 품질 규칙 생성 및 실행. **Informatica**는 프로젝트 목표에 맞춰 실행하거나 편집할 수 있는 규칙을 제공합니다. **Developer tool**에서 맵셋을 생성하고 규칙으로 맵셋의 유효성을 검사할 수 있습니다.
- **Informatica** 사용자와 공동 작업. 모델 리포지토리는 참조 데이터 및 규칙을 저장하며 **Developer tool** 및 **Analyst** 도구의 사용자가 이러한 리포지토리를 사용할 수 있습니다. 사용자는 프로젝트에 대한 공동 작업을 할 수 있고 다른 사용자는 다른 단계의 프로젝트에서 개체 소유권을 보유할 수 있습니다.
- 메타데이터를 재사용하려면 **Developer tool** 또는 **infacmd**를 사용하여 **PowerCenter**에서 매핑을 가져오거나 **PowerCenter**로 매핑을 내보냅니다.

PowerCenter에서 모델 리포지토리로 데이터를 가져오려면 다음 태스크를 완료합니다.

1. **PowerCenter** 클라이언트 또는 다음 명령을 사용하여 **PowerCenter** 개체를 파일로 내보냅니다.
`pmrep ExportObject`
2. 다음 명령을 사용하여 내보내기 파일을 모델 리포지토리 파일로 변환합니다.
`infacmd ipc importFromPC`
3. **Developer tool** 또는 다음 명령을 사용하여 개체를 가져옵니다.
`infacmd tools importObjects`

모델 리포지토리에서 **PowerCenter** 리포지토리로 데이터를 내보내려면 다음 태스크를 완료합니다.

1. **Developer tool** 또는 다음 명령을 사용하여 모델 리포지토리 개체를 파일로 내보냅니다.
`infacmd tools ExportObjects`
 또는 바로 `infacmd ipc ExportToPC`를 실행하여 내보낼 수도 있습니다.
2. 다음 명령을 사용하여 내보내기 파일을 **PowerCenter** 파일로 변환합니다.
`infacmd ipc ExporttoPC`
3. **PowerCenter** 또는 다음 명령을 사용하여 개체를 가져옵니다.
`pmrep importObjects`

Informatica Data Services

데이터 서비스는 데이터에 액세스하고 데이터를 변환하기 위해 실행할 수 있는 재사용 가능한 작업의 모음입니다.

Developer tool에서 데이터 서비스 기능을 사용하여 다음과 같은 작업을 수행할 수 있습니다.

- 데이터의 논리적 보기를 정의합니다. 데이터의 논리적 보기는 기업 내 데이터의 구조와 사용을 설명합니다. 기업에서 사용하는 데이터의 유형과 해당 데이터가 구조화되는 방식을 보여 주는 논리적 데이터 개체 모델을 생성할 수 있습니다.
- 논리적 모델을 데이터 소스 또는 대상에 매핑. 논리적 모델의 개체를 데이터 소스 또는 대상에 연결하는 매핑을 생성합니다. 서로 다른 여러 소스의 데이터를 연결하여 데이터의 단일 보기를 생성할 수 있습니다. 또한 모델을 따르는 데이터를 서로 다른 여러 대상에 로드할 수 있습니다.
- 데이터의 가상 보기 생성. 가상 페더레이션 데이터베이스를 데이터 통합 서비스에 배포할 수 있습니다. 최종 사용자는 실제 소스 데이터에 영향을 주지 않으면서 가상 데이터에 대해 SQL 쿼리를 실행할 수 있습니다.
- 웹 서비스 인터페이스를 통해 데이터 통합 기능에 대한 액세스 제공. 웹 서비스를 데이터 통합 서비스에 배포할 수 있습니다. 최종 사용자는 SOAP 메시지를 통해 요청을 웹 서비스에 보내고 응답을 받습니다.
- 메타데이터를 재사용하려면 Developer tool 또는 infacmd를 사용하여 PowerCenter에서 매핑을 가져오거나 PowerCenter로 매핑을 내보냅니다.

PowerCenter에서 모델 리포지토리로 데이터를 가져오려면 다음 작업을 완료합니다.

1. PowerCenter 클라이언트 또는 다음 명령을 사용하여 PowerCenter 개체를 파일로 내보냅니다.
`pmrep ExportObject`
2. 다음 명령을 사용하여 내보내기 파일을 모델 리포지토리 파일로 변환합니다.
`infacmd ipc importFromPC`
3. Developer tool 또는 다음 명령을 사용하여 개체를 가져옵니다.
`infacmd tools importObjects`

모델 리포지토리에서 PowerCenter 리포지토리로 데이터를 내보내려면 다음 작업을 완료합니다.

1. Developer tool 또는 다음 명령을 사용하여 모델 리포지토리 개체를 파일로 내보냅니다.
`infacmd tools exportObjects`
또는 바로 `infacmd ipc ExportToPC`를 실행하여 내보낼 수도 있습니다.
 2. 다음 명령을 사용하여 내보내기 파일을 PowerCenter 파일로 변환합니다.
`infacmd ipc ExporttoPC`
 3. PowerCenter 또는 다음 명령을 사용하여 개체를 가져옵니다.
`pmrep importObjects`
- 도메인 사용자가 명령줄에서 실행할 수 있는 매핑 생성 및 배포.
 - 프로파일 데이터. 프로파일링 옵션을 사용하면 데이터를 프로파일링하여 데이터의 콘텐츠와 구조를 파악할 수 있습니다. 프로파일링은 데이터의 강점과 단점을 파악하여 프로젝트 계획을 정립하는 데 도움이 될 수 있으므로 어떤 데이터 프로젝트에서도 핵심 단계로 꼽힙니다.

Informatica Developer 시작

Developer 도구가 로컬 시스템에 설치되어 있을 경우 **Windows** 시작 메뉴를 사용하여 도구를 시작합니다.
Developer 도구가 원격 시스템에 설치되어 있을 경우에는 명령줄을 사용하여 도구를 시작합니다.

로컬 시스템에서 Developer 도구 시작

Windows 시작 메뉴를 사용하여 로컬 시스템에 설치된 Developer 도구를 시작합니다.

1. Windows 시작 메뉴에서 **모든 프로그램 > Informatica [버전] > 클라이언트 > Developer 클라이언트 > Informatica Developer 실행**을 클릭합니다.

Developer 도구를 처음으로 실행하면 시작 페이지에 여러 가지 아이콘이 표시됩니다. Developer 도구를 다시 실행하면 시작 페이지가 나타나지 않습니다.

2. **작업 영역**을 클릭합니다.

Developer tool을 처음으로 시작하는 경우 도메인 및 모델 리포지토리를 추가해야 합니다. 도메인에 둘 이상의 데이터 통합 서비스가 포함되어 있는 경우 기본 서비스를 선택해야 합니다.

원격 시스템에서 Developer 도구 시작

명령줄을 사용하여 원격 시스템에 설치된 Developer 도구를 시작합니다.

Developer 도구가 원격 시스템에 설치된 경우 설치 디렉터리에 대한 쓰기 권한이 없을 수 있습니다. Developer 도구가 임시 파일을 쓸 수 있는 로컬 시스템의 작업 공간 디렉터리를 지정해야 합니다. 관리자는 모든 사용자의 기본 로컬 작업 공간 디렉터리를 구성할 수 있습니다. 사용자는 Developer 도구를 시작할 때 기본 디렉터리를 재정의할 수 있습니다.

구성된 로컬 작업 공간 디렉터리가 존재하지 않을 경우 Developer 도구가 임시 파일을 쓸 때 이 디렉터리를 작성합니다.

1. 명령 프롬프트를 엽니다.
2. 명령을 입력하여 Developer 도구를 시작합니다. 기본 로컬 작업 공간 디렉터리를 사용하거나 기본 디렉터리를 재정의할 수 있습니다.

- 기본 로컬 작업 공간 디렉터리를 사용하려면 다음 명령을 입력합니다.

```
\\<remote installation directory>\developer.exe
```

예:

```
\\MyRemoteMachine\Informatica\9.5.1\clients\DeveloperClient\developer.exe
```

- 기본 로컬 작업 공간 디렉터리를 재정의하려면 다음 명령을 입력합니다.

```
\\<remote installation directory>\developer.exe -data <local workspace directory>
```

예:

```
\\MyRemoteMachine\Informatica\9.5.1\clients\DeveloperClient\developer.exe -data C:\temp\MyWorkspace
```

로컬 작업 공간 디렉터리의 폴더 이름에 숫자 기호(#) 문자를 포함할 수 없습니다. 로컬 작업 공간 디렉터리의 폴더 이름에 공백이 있을 경우 전체 경로를 큰따옴표로 묶습니다.

Developer 도구를 처음으로 실행하면 시작 페이지에 여러 가지 아이콘이 표시됩니다. Developer 도구를 다시 실행하면 시작 페이지가 나타나지 않습니다.

3. **작업 영역**을 클릭합니다.

Developer tool을 처음으로 시작하는 경우 도메인 및 모델 리포지토리를 추가해야 합니다. 도메인에 둘 이상의 데이터 통합 서비스가 포함되어 있는 경우에는 기본 데이터 통합 서비스를 선택해야 합니다.

Informatica Developer 인터페이스

Developer tool을 사용하여 데이터 품질 및 데이터 서비스 솔루션을 설계하고 구현할 수 있습니다.

Developer tool에서 동시에 여러 가지 태스크를 수행할 수 있습니다. 또한 여러 폴더 및 프로젝트에서 동시에 작업할 수 있습니다. Developer tool에서 작업하려면 Developer tool 작업 영역에 액세스합니다.

다음 그림은 Developer tool 작업 영역을 보여 줍니다.



1. Object Explorer 보기
2. 연결 탐색기 보기
3. 아웃라인 보기
4. 진행률 보기
5. 속성 보기
6. 데이터 뷰어 보기
7. 편집기

Windows에서 Developer tool의 권장되는 화면 해상도는 1920 * 1080입니다.

Informatica Developer의 보기

Developer tool 작업 영역은 편집기와 보기로 구성됩니다. 편집기에서는 매핑 같은 개체를 편집합니다.

Developer tool은 편집기에서 선택된 개체에 기반하여 보기를 표시합니다.

추가적인 보기를 선택하고 보기를 숨길 수 있으며 Developer tool 작업 영역의 다른 위치로 보기를 이동할 수 있습니다.

표시할 보기를 선택하려면 창 > 보기 표시를 클릭하십시오.

기본 보기

기본적으로 Developer tool은 다음과 같은 보기를 표시합니다.

연결 탐색기 보기

관계형 데이터베이스에 대한 연결을 표시합니다.

데이터 뷰어 보기

소스 데이터, 프로필 결과를 표시하고 변환의 출력을 미리 봅니다.

Object Explorer 보기

도메인과 도메인의 디자인 타임 개체 및 런타임 개체를 표시합니다. 디자인 타임 개체는 모델 리포지토리의 프로젝트와 폴더에 저장됩니다. 런타임 개체는 런타임 응용 프로그램의 일부로 데이터 통합 서비스에 저장됩니다.

아웃라인 보기

Object Explorer 보기에서 선택된 개체에 종속된 개체를 표시합니다.

진행률 보기

Developer tool에서 작업(예: 매핑 실행)의 진행률을 표시합니다.

속성 보기

편집기에서 선택된 개체의 속성을 표시합니다.

추가 보기

Developer tool 작업 영역에는 다음과 같은 보기도 표시됩니다.

알림 보기

연결 상태 알림을 표시합니다.

치트 시트 보기

사용자가 여는 치트 시트를 표시합니다. 치트 시트를 열려면 **도움말 > 치트 시트**를 클릭하고 치트 시트를 선택합니다.

체크 아웃된 개체 보기

체크 아웃한 모든 개체를 표시합니다.

[데이터 프로세서 이벤트] 보기

Developer tool에서 데이터 프로세서 변환을 실행할 때 발생하는 초기화, 실행 및 요약 이벤트에 대한 정보를 표시합니다.

[데이터 프로세서 16진수 소스] 보기

입력 문서를 16진수 형식으로 표시합니다.

도움말 보기

상황에 맞는 온라인 도움말을 표시합니다.

개체 종속성 보기

개체를 보거나 수정하거나 삭제할 때 개체 종속성을 표시합니다.

검색 보기

검색 결과를 표시합니다. 또한 검색 옵션 대화 상자를 실행할 수 있습니다.

태그 보기

비즈니스 용도에 따라 모델 리포지토리의 개체를 정의하는 태그를 표시합니다.

[알림] 보기

휴먼 태스크의 모든 작업이 완료되면 사용자나 그룹에 알리는 옵션을 표시합니다.

유효성 검사 로그 보기

개체 유효성 검사 오류를 표시합니다.

버전 기록 보기

선택한 개체의 버전 기록을 표시합니다. 체크 인 설명을 읽고 개체 체크 인에 대한 사용자 정보를 볼 수 있습니다.

Informatica Developer 시작 페이지

Developer tool을 처음으로 열면 시작 페이지가 나타납니다. 시작 페이지에서 Developer tool을 설정하고 시작하는 방법을 익힐 수 있습니다.

시작 페이지에는 다음 옵션이 표시됩니다.

- 개요. 개요 단추를 클릭하면 데이터 품질 및 데이터 서비스 솔루션에 대한 개요가 표시됩니다.
- 첫 번째 단계. 첫 번째 단계 단추를 클릭하면 Developer tool을 설정하고 Informatica Data Quality 및 Informatica Data Services 단원에 액세스하는 방법을 확인할 수 있습니다.
- 자습서. 자습서 단추를 클릭하면 Developer tool과 데이터 품질 및 데이터 서비스 솔루션에 대한 치트 시트를 볼 수 있습니다.
- 웹 리소스. 웹 리소스 단추를 클릭하면 Informatica 기술 자료에 연결됩니다. Informatica 방법 라이브러리에 액세스할 수 있습니다. Informatica 방법 라이브러리에는 Informatica Data Quality, Informatica Data Services 및 기타 Informatica 제품에 대한 문서가 포함되어 있습니다.
- 작업 영역. 작업 영역 단추를 클릭하면 Developer tool에서 작업을 시작할 수 있습니다.

시작 페이지를 닫은 후 시작 페이지에 액세스하려면 **도움말 > 시작**을 클릭합니다.

치트 시트

Developer 도구에는 온라인 도움말의 일부로 치트 시트가 포함됩니다. 치트 시트는 Developer 도구에서 하나 이상의 태스크를 완료하는 데 도움이 되는 단계별 가이드입니다.

치트 시트를 따라 태스크를 수행하고 결과를 살펴봅니다. 예를 들어 차트 시트를 수행하여 실제 데이터 개체를 가져오고 미리 볼 수 있습니다.

치트 시트에 액세스하려면 **도움말 > 치트 시트**를 클릭합니다.

Informatica Developer 온라인 도움말

Informatica Developer 온라인 도움말 시스템은 Developer tool을 최대한 활용하는 데 필요한 정보를 포함하고 있습니다. 목차, 검색 및 색인 옵션을 사용하여 Developer tool 기능에 대해 알아봅니다. 유용하다고 생각되는 페이지에 북마크를 추가합니다. 관련 항목 옵션을 사용하여 유사한 정보가 포함된 페이지를 찾습니다.

Informatica 기본 설정

기본 설정 대화 상자에는 Developer 도구 및 Eclipse 플랫폼의 설정이 포함되어 있습니다.

Informatica 기본 설정을 통해 Developer 도구의 설정을 관리합니다. 예를 들어 Informatica 기본 설정을 사용하여 구성, 연결, 변환 설정, 태그 또는 사용 가능한 데이터 통합 서비스를 관리할 수 있습니다.

Developer 도구는 Eclipse 플랫폼을 기반으로 합니다. **기본 설정** 대화 상자에는 Eclipse 플랫폼의 설정을 관리하는 기본 설정도 포함되어 있습니다. Informatica는 Informatica 기본 설정만 지원합니다.

Informatica 기본 설정에 액세스하려면 **창 > 기본 설정**을 클릭합니다. **기본 설정** 대화 상자에서 **Informatica**를 선택합니다.

Informatica Marketplace

Informatica Marketplace에서는 데이터 통합 구현을 확충하거나 확장하거나 향상시키는 사전 구성 솔루션을 제공합니다.

Informatica Marketplace에 액세스하려면 도구 모음에서 **Marketplace**를 클릭합니다. Marketplace 보기가 Developer tool에서 나타납니다.

처음으로 Marketplace에 로그인하려면 먼저 사용자로 등록해야 합니다.

로그인한 후 편집기에서 미리 작성된 솔루션에 대한 링크를 볼 수 있습니다. Marketplace 검색 상자에서 솔루션을 검색하고 검색 결과를 확인하여 솔루션을 찾을 수 있습니다. Developer tool에서 사용하기 위해 모델 리포지토리로 가져올 수 있는 매핑, 매핑 개체, 프로필 또는 워크플로우가 솔루션에 포함될 수 있습니다.

Marketplace 솔루션을 가져오려면 Marketplace 솔루션 옆의 **가져오기** 단추를 클릭하고 단계에 따라 솔루션을 모델 리포지토리로 가져옵니다. 솔루션을 가져오려면 모델 리포지토리에 연결되어 있어야 합니다. 솔루션의 관련 소스 파일 및 설명서를 복사하기 위해 가져오기 프로세스 중에 폴더를 선택해야 합니다.

솔루션을 모델 리포지토리로 가져온 후 매핑을 실행하거나, 매핑을 실행하기 전에 편집할 수 있습니다.

또한 Marketplace 커뮤니티의 다른 사용자에게 도움이 되도록 솔루션을 게시할 수도 있습니다.

Informatica Developer 설정

Informatica Developer를 설정하여 모델 리포지토리 개체에 액세스합니다. 데이터 통합 서비스를 선택하여 데이터를 미리 보고, 프로필, 매핑 및 워크플로우를 실행합니다.

Developer tool을 설정하려면 다음 태스크를 수행합니다.

1. 도메인을 추가합니다.
2. 모델 리포지토리를 추가합니다.
3. 도메인에 둘 이상의 데이터 통합 서비스가 포함되어 있는 경우 기본 서비스를 선택합니다.
4. 도메인에 둘 이상의 메타데이터 액세스 서비스가 포함되어 있는 경우 기본 서비스를 선택합니다.

Developer tool을 설정한 후 작업을 저장할 프로젝트 및 폴더를 작성할 수 있습니다.

1단계. 도메인 추가

도메인에서 실행되는 서비스에 액세스하기 위해 Developer 도구에서 도메인을 추가합니다.

도메인을 추가하기 전에 도메인 이름, 호스트 이름 및 도메인에 연결할 포트 번호가 있는지 확인합니다. 이 정보는 관리자로부터 얻을 수 있습니다.

1. **창 > 기본 설정**을 클릭합니다.
기본 설정 대화 상자가 표시됩니다.
2. **Informatica > 도메인**을 선택합니다.
3. **추가**를 클릭합니다.
새 도메인 대화 상자가 표시됩니다.
4. 도메인 이름, 호스트 이름 및 포트 번호를 입력합니다.
5. **마침**을 클릭합니다.
6. **확인**을 클릭합니다.

2단계. 모델 리포지토리 추가

프로젝트 및 폴더에 액세스하기 위해 모델 리포지토리를 추가합니다.

모델 리포지토리를 추가하기 전에 다음 선행 조건을 확인합니다.

- 관리자가 **Administrator** 도구에서 모델 리포지토리 서비스를 구성했습니다.
- 모델 리포지토리 서비스에 액세스하기 위한 사용자 이름과 암호가 있습니다. 이 정보는 관리자로부터 얻을 수 있습니다.

1. **파일 > 리포지토리에 연결**을 클릭합니다.

리포지토리에 연결 대화 상자가 표시됩니다.

2. **찾아보기**를 클릭하여 모델 리포지토리 서비스를 선택합니다.

3. **확인**을 클릭합니다.

4. **다음**을 클릭합니다.

5. 사용자 이름과 암호를 입력합니다.

6. **다음**을 클릭합니다.

프로젝트 열기 대화 상자가 나타납니다.

7. **개체 탐색기** 보기에 나타나는 프로젝트 목록을 필터링하려면 열지 않을 프로젝트의 선택을 취소합니다.

8. **마침**을 클릭합니다.

모델 리포지토리 서비스가 **Object Explorer** 보기에 나타나고 열기 위해 선택한 프로젝트가 표시됩니다.

Object Explorer 보기는 서비스가 포함되어 있는 도메인도 표시합니다.

3단계. 기본 데이터 통합 서비스 선택

데이터 통합 서비스는 **Developer** 도구의 데이터 통합 태스크를 수행합니다. 도메인에 둘 이상의 데이터 통합 서비스가 포함되어 있는 경우 기본 서비스를 선택합니다. 매핑을 실행하거나 데이터를 미리 볼 때 기본 데이터 통합 서비스를 재정의할 수 있습니다.

참고: 도메인에 하나의 데이터 통합 서비스만 포함되는 경우 이 단계는 선택 사항입니다.

데이터 통합 서비스를 선택하기 전에 도메인을 추가합니다.

1. **창 > 기본 설정**을 클릭합니다.

기본 설정 대화 상자가 표시됩니다.

2. **Informatica > 데이터 통합 서비스**를 선택합니다.

3. 도메인을 확장합니다.

4. 데이터 통합 서비스를 선택합니다.

5. **기본값으로 설정**을 클릭합니다.

6. **확인**을 클릭합니다.

4단계. 기본 메타데이터 액세스 서비스 선택

메타데이터 액세스 서비스는 **Developer tool**이 **Hadoop** 연결 정보에 액세스하여 메타데이터를 가져오고 미리 볼 수 있게 해줍니다. 도메인에 둘 이상의 메타데이터 액세스 서비스가 포함되어 있는 경우 기본 서비스를 선택합니다. 데이터 개체를 가져올 때 기본 메타데이터 액세스 서비스를 재정의할 수 있습니다.

참고: 도메인에 하나의 메타데이터 액세스 서비스만 포함되는 경우 이 단계는 선택 사항입니다.

메타데이터 액세스 서비스를 선택하기 전에 도메인을 추가합니다.

1. **창 > 기본 설정**을 클릭합니다.
기본 설정 대화 상자가 표시됩니다.
2. **Informatica > 메타데이터 액세스 서비스**를 선택합니다.
3. 도메인을 확장합니다.
4. 메타데이터 액세스 서비스를 선택합니다.
5. **기본값으로 설정**을 클릭합니다.
6. **확인**을 클릭합니다.

도메인

Informatica 도메인은 Informatica 환경을 정의하는 노드 및 서비스의 컬렉션입니다.

Developer 도구에서 도메인을 추가합니다. 또한 도메인 정보를 편집하거나 도메인을 제거할 수 있습니다.

Developer 도구 기본 설정에서 도메인 정보를 관리합니다.

프로젝트

프로젝트는 Developer 도구에서 폴더 및 개체를 저장하는 데 사용하는 최상위 컨테이너입니다.

프로젝트를 사용하여 데이터 서비스 및 데이터 품질 솔루션에 사용할 개체를 체계화하고 관리할 수 있습니다.

개체 탐색기 보기에서 프로젝트를 관리하고 봅니다. 프로젝트를 작성하면 Developer 도구가 프로젝트를 모델 리포지토리에 저장합니다.

작성하는 각 프로젝트는 Analyst 도구에도 나타납니다.

다음 표에는 프로젝트에 대해 수행할 수 있는 태스크가 설명되어 있습니다.

태스크	설명
프로젝트 관리	프로젝트 콘텐츠를 관리합니다. 프로젝트를 생성 및 복제하고 이름을 바꾸고 삭제할 수 있습니다. 프로젝트 콘텐츠를 볼 수 있습니다.
프로젝트 필터링	Object Explorer 보기에 나타나는 프로젝트 목록을 필터링합니다.
폴더 관리	폴더에서 프로젝트 콘텐츠를 체계화합니다. 프로젝트 내에서 폴더를 생성 및 복제하고 이름을 바꾸고 이동할 수 있습니다.
개체 관리	프로젝트에서 또는 프로젝트 내 폴더에서 개체 콘텐츠를 보고, 개체를 복제하고, 이름을 바꾸고, 개체를 이동 및 삭제합니다.
프로젝트 검색	프로젝트에서 폴더 또는 개체를 검색합니다. 검색 결과를 보고 결과에서 개체를 선택하여 해당 내용을 확인할 수 있습니다.
사용 권한 할당	프로젝트의 개체를 보고 편집할 수 있는 사용자 및 그룹을 선택합니다. 다른 사용자 및 그룹에 사용 권한을 할당할 수 있는 사용자 및 그룹을 지정합니다.

프로젝트 작성

개체 및 폴더를 저장할 프로젝트를 작성합니다.

1. **개체 탐색기** 보기에서 모델 리포지토리 서비스를 선택합니다.
2. **파일 > 새로 만들기 > 프로젝트**를 클릭합니다.
새 프로젝트 대화 상자가 나타납니다.
3. 프로젝트의 이름을 입력합니다.
4. **다음**을 클릭합니다.
새 프로젝트 대화 상자의 **프로젝트 사용 권한** 페이지가 나타납니다.
5. 필요에 따라 사용자 또는 그룹을 선택하고 사용 권한을 할당합니다.
6. **마침**을 클릭합니다.
개체 탐색기 보기의 모델 리포지토리 서비스 아래에 프로젝트가 표시됩니다.

프로젝트 필터링

개체 탐색기 보기에 나타나는 프로젝트 목록을 필터링할 수 있습니다. 많은 수의 프로젝트에 액세스할 수 있지만 그 중 일부만 관리해야 하는 경우 프로젝트를 필터링할 수 있습니다.

Developer 도구는 사용자가 필터링한 프로젝트 목록을 다음 번 리포지토리에 연결할 때까지 유지합니다.

다음과 같은 시점에 프로젝트를 필터링할 수 있습니다.

리포지토리에 연결하기 전

리포지토리에 연결하기 전에 프로젝트를 필터링하면 Developer 도구가 리포지토리에 연결하는 데 걸리는 시간을 줄일 수 있습니다.

파일 > 리포지토리에 연결을 선택합니다. 리포지토리를 선택하고 사용자 이름과 암호를 입력한 후 **다음**을 클릭합니다. **프로젝트 열기** 대화 상자에 액세스 권한이 있는 모든 프로젝트가 표시됩니다. 리포지토리에서 열려는 프로젝트를 선택한 후 **마침**을 클릭합니다.

리포지토리에 연결한 후

리포지토리에 연결된 경우 **파일 > 프로젝트 닫기**를 클릭하여 개체 탐색기 보기에서 프로젝트를 필터링하고 제외합니다. **프로젝트 닫기** 대화 상자에 현재 개체 탐색기 보기에서 열려 있는 모든 프로젝트가 표시됩니다. 필터링하여 제외할 프로젝트를 선택한 후 **마침**을 클릭합니다.

필터링한 프로젝트를 열려면 **파일 > 프로젝트 열기**를 클릭합니다.

프로젝트 사용 권한

사용자 또는 그룹에 프로젝트 사용 권한을 할당합니다. 프로젝트 사용 권한은 사용자 또는 그룹이 개체를 보거나, 개체를 편집하거나, 다른 사용자에게 권한을 할당할 수 있는지 여부를 결정합니다.

다음과 같은 사용 권한을 할당할 수 있습니다.

읽기

사용자 또는 그룹이 프로젝트의 모든 개체를 열고, 미리 보고, 내보내고, 유효성을 검사하고, 배포할 수 있습니다. 사용자 또는 그룹이 프로젝트 세부 정보를 볼 수도 있습니다.

쓰기

사용자 또는 그룹이 프로젝트의 모든 개체에 대한 읽기 권한을 가집니다. 또한 사용자 또는 그룹이 프로젝트의 모든 개체를 편집하고, 프로젝트 세부 정보를 편집하고, 프로젝트의 모든 개체를 삭제하고, 프로젝트를 삭제할 수 있습니다.

권한 부여

사용자 또는 그룹이 프로젝트의 모든 개체에 대한 읽기 권한을 가집니다. 그리고 사용자 또는 그룹이 다른 사용자 또는 그룹에 사용 권한을 할당할 수 있습니다.

모델 리포지토리 서비스에 대한 관리자 역할이 할당된 사용자는 모델 리포지토리 서비스의 모든 프로젝트에 대한 모든 사용 권한을 상속합니다. 그룹에 할당된 사용자는 그룹 사용 권한을 상속합니다.

외부 개체에 대한 사용 권한

사용 권한은 프로젝트 내의 개체에 적용됩니다. **Developer tool**은 종속 개체가 다른 프로젝트에 있는 경우 종속 개체에 대한 사용 권한을 확장하지 않습니다.

종속 개체는 다른 개체가 사용하는 개체입니다. 예를 들어, 재사용 불가능한 식 변환이 포함된 맵렛을 작성한다고 가정하면 맵렛은 상위 개체이고 식 변환은 맵렛의 종속 개체입니다.

상위 프로젝트 내의 재사용 가능한 개체를 사용할 때 **Developer tool**은 개체의 인스턴스를 작성합니다. 예를 들어 재사용 가능한 조희 변환이 포함된 매핑을 작성한다고 가정하면 매핑은 상위 개체이고 여기에는 조희 변환의 인스턴스가 포함됩니다.

개체는 다른 프로젝트에 있는 종속 개체의 인스턴스를 포함할 수 있습니다. 다른 프로젝트의 종속 개체 인스턴스를 보려면 다른 프로젝트에 대한 읽기 권한이 필요합니다. 다른 프로젝트의 종속 개체 인스턴스를 편집하려면 상위 개체 프로젝트에 대한 쓰기 권한과 다른 프로젝트에 대한 읽기 권한이 필요합니다.

종속 개체 인스턴스에 대한 사용 권한

다른 프로젝트의 종속 개체 인스턴스가 포함된 개체에 액세스해야 할 수 있습니다. 다른 개체에 대한 읽기 권한이 없을 경우 **Developer tool**은 사용자가 상위 개체에 액세스하는 방법에 따라 여러 가지 옵션을 제공합니다.

볼 수 없는 종속 개체 인스턴스가 포함된 상위 개체에 액세스하려 하는 경우 **Developer tool**은 경고 메시지를 표시합니다. 작업을 계속하면 **Developer tool**이 작업 유형별로 다른 결과를 생성합니다.

다음 표에는 상위 개체에 대해 수행할 수 있는 작업의 결과가 나와 있습니다.

작업	결과
상위 개체를 엽니다.	Developer tool 이 상위 개체를 여는 방법을 지정하라는 메시지를 사용자에게 표시합니다. <ul style="list-style-type: none">- 사본 열기. Developer tool이 상위 개체의 사본을 작성합니다. 사본에는 볼 수 없는 종속 개체 인스턴스가 포함되지 않습니다.- 열기. Developer tool이 개체를 열지만 볼 수 없는 종속 개체 인스턴스를 제거합니다. 상위 개체를 저장하면 Developer tool이 상위 개체에서 종속 개체 인스턴스를 제거합니다. Developer tool은 리포지토리에서 종속 개체를 제거하지 않습니다.- 취소. Developer tool이 상위 개체를 열지 않습니다.
Developer tool 에서 사용하기 위해 상위 개체를 XML 파일로 내보냅니다.	Developer tool 이 종속 개체 인스턴스 없이 내보내기 파일을 생성합니다.
상위 개체를 PowerCenter로 내보냅니다.	상위 개체를 내보낼 수 없습니다.

작업	결과
상위 개체의 유효성을 검사합니다.	종속 개체가 상위 개체의 일부가 아닌 것처럼 Developer tool이 상위 개체의 유효성을 검사합니다.
상위 개체를 배포합니다.	상위 개체를 배포할 수 없습니다.
상위 개체를 복사하고 붙여 넣습니다.	Developer tool이 종속 개체 인스턴스 없이 새 개체를 생성합니다.

보안 세부 정보

볼 수 없는 종속 개체 인스턴스가 포함된 상위 개체에 액세스하면 Developer 도구가 경고 메시지를 표시합니다. 경고 메시지를 통해 종속 개체에 대한 세부 정보를 확인할 수 있습니다.

종속 개체에 대한 세부 정보를 보려면 경고 메시지에서 **세부 정보** 단추를 클릭합니다. 보안 세부 정보 표시 모델 리포지토리 서비스 권한이 있을 경우 Developer 도구가 사용자가 볼 수 없는 개체가 포함된 프로젝트를 나열합니다. 보안 세부 정보 표시 권한이 없을 경우 Developer 도구는 프로젝트 이름을 볼 수 있는 충분한 권한이 없음을 사용자에게 알립니다.

상위 개체 액세스

다른 프로젝트의 종속 개체 인스턴스를 사용하는 상위 개체를 작성하는 경우 사용자가 상위 개체를 편집하지 못할 수 있습니다. 사용자가 상위 개체를 편집할 수 있도록 하고 상위 개체 기능을 유지하려면 맵셋에 종속 개체의 인스턴스를 작성합니다.

예를 들어 다른 프로젝트의 재사용 가능한 조회 변환을 포함하는 맵핑을 작성할 수 있습니다. 프로젝트의 사용자가 조회 변환이 아닌 맵핑을 편집할 수 있게 설정할 수 있습니다.

조회 변환을 맵핑에 배치하면 다른 프로젝트에 대한 읽기 권한이 없는 사용자가 맵핑을 열려고 하면 경고 메시지가 표시됩니다. 이러한 사용자는 맵핑의 사본을 열거나 맵핑을 열 수 있지만 Developer 도구가 조회 변환 인스턴스를 제거합니다.

사용자가 맵핑을 편집할 수 있도록 하려면 다음 태스크를 수행합니다.

1. 프로젝트에 맵셋을 작성합니다. 입력 변환, 재사용 가능한 조회 변환 및 출력 변환을 맵셋에 추가합니다.
2. 맵핑을 편집하고 조회 변환을 맵셋으로 대체합니다.
3. 맵핑을 저장합니다.

프로젝트의 사용자가 맵핑을 열면 조회 변환 대신 맵셋이 표시됩니다. 사용자는 맵셋을 제외하고 맵핑의 모든 부분을 편집할 수 있습니다.

사용자가 맵핑을 내보내면 Developer 도구가 내보내기 파일에 조회 변환을 포함하지 않습니다.

사용 권한 할당

사용자 및 그룹을 프로젝트에 추가하고 이러한 사용자 및 그룹에 대해 사용 권한을 할당할 수 있습니다. 사용 권한을 할당하여 사용자가 프로젝트의 개체에 대해 수행할 수 있는 태스크를 결정합니다.

1. **개체 탐색기** 보기에서 프로젝트를 선택합니다.
2. **파일 > 속성**을 클릭합니다.
속성 창이 표시됩니다.
3. **사용 권한**을 선택합니다.

4. **추가**를 클릭하여 사용자를 추가하고 사용자에 대해 사용 권한을 할당합니다.
도메인 사용자 및 그룹 대화 상자가 나타납니다.
5. 사용자 및 그룹 목록을 필터링하려면 이름 또는 문자열을 입력합니다.
필요에 따라 필터에 와일드카드 문자를 사용합니다.
6. 보안 도메인별로 필터링하려면 **보안 도메인을 기준으로 필터링** 단추를 클릭합니다.
7. 기본 보안 도메인의 사용자 및 그룹을 표시하려면 **기본**을 선택합니다. 또는 모든 사용자 및 그룹을 표시하려면 **모두**를 선택합니다.
8. 사용자 또는 그룹을 선택하고 **확인**을 클릭합니다.
새 프로젝트 대화 상자의 **프로젝트 사용 권한** 페이지에 사용자 또는 그룹이 나타납니다.
9. 사용자 또는 그룹에 대해 읽기, 쓰기 또는 부여 권한을 선택합니다.
10. **확인**을 클릭합니다.

폴더

폴더를 사용하여 프로젝트의 개체를 체계화합니다. 폴더를 작성하여 비즈니스 요구에 따라 개체를 그룹화합니다. 예를 들어 폴더를 작성하여 프로젝트의 특정 태스크에 대한 개체를 그룹화할 수 있습니다. 프로젝트 또는 다른 폴더에서 폴더를 작성할 수 있습니다.

폴더는 **개체 탐색기** 보기에서 프로젝트 내에 나타납니다. 폴더는 다른 폴더, 데이터 개체 및 개체 유형을 포함할 수 있습니다.

폴더에 대해 다음과 같은 태스크를 수행할 수 있습니다.

- 폴더를 작성합니다.
- 폴더를 봅니다.
- 폴더 이름을 바꿉니다.
- 폴더를 복제합니다.
- 폴더 이동.
- 폴더를 삭제합니다.

폴더 작성

폴더를 작성하여 프로젝트의 관련 개체를 저장합니다. 폴더를 프로젝트 또는 다른 폴더 내에 작성해야 합니다.

1. **개체 탐색기** 보기에서 폴더를 작성할 프로젝트 또는 폴더를 선택합니다.
2. **파일 > 새로 만들기 > 폴더**를 클릭합니다.
새 폴더 대화 상자가 나타납니다.
3. 폴더 이름을 입력합니다.
4. **마침**을 클릭합니다.
폴더가 해당 프로젝트 또는 상위 폴더 아래 나타납니다.

개체 작업 복사

프로젝트 내에서 또는 다른 프로젝트로 개체를 복사할 수 있습니다. 또한 같은 프로젝트의 폴더나 다른 프로젝트의 폴더로 개체를 복사할 수 있습니다.

개체의 사본을 다른 이름으로 저장할 수 있습니다. 또한 개체를 링크로 복사하여 **Analyst** 도구에서 개체를 보거나 전자 메일 메시지 등의 다른 매체에서 개체에 대한 링크를 제공할 수 있습니다.

다음과 같은 개체를 다른 프로젝트 또는 폴더로 복사하거나, 개체의 사본을 다른 이름으로 저장하거나, 개체를 링크로 복사할 수 있습니다.

- 응용 프로그램
- 데이터 서비스
- 논리적 데이터 개체 모델
- 맵핑
- 맵렛
- 실제 데이터 개체
- 프로필
- 참조 테이블
- 재사용 가능 변환
- 규칙
- 성과 기록표
- 가상 저장 프로시저
- 워크플로우

개체를 복사할 때는 다음 지침을 따릅니다.

- 맵핑, 맵렛, 규칙 및 가상 저장 프로시저의 세그먼트를 복사할 수 있습니다.
- 폴더를 다른 프로젝트에 복사할 수 있습니다.
- 논리적 데이터 개체를 링크로 복사할 수 있습니다.
- 개체를 복사한 후 여러 번 붙여 넣을 수 있습니다.
- 프로젝트 또는 폴더에 동일한 이름의 개체가 있을 경우 개체의 이름을 바꾸거나 개체를 대체할 수 있습니다.

개체 복사

개체를 복사해서 다른 프로젝트 또는 폴더에서 사용 가능하게 합니다.

1. 프로젝트 또는 폴더에서 개체를 선택합니다.
2. **편집 > 복사**를 클릭합니다.
3. 개체를 복사할 프로젝트 또는 폴더를 선택합니다.
4. **편집 > 붙여넣기**를 클릭합니다.

개체 사본 저장

개체의 사본을 저장하여 개체를 다른 이름으로 저장합니다.

모델 리포지토리 외부에 있는 참조 데이터 값을 지정하는 참조 데이터 개체의 사본을 저장하면 모델 리포지토리에서 개체가 올바르게 작동하지 않다고 표시합니다. 사본을 작성하려면 **Object Explorer** 보기에서 개체를 선택하고 **편집 > 복사**를 클릭한 후 복사한 개체를 원하는 위치에 붙여 넣습니다.

1. 편집기에서 개체를 엽니다.
2. **파일 > 다른 이름으로 사본 저장**을 클릭합니다.
3. 개체 사본의 이름을 입력합니다.
4. **찾아보기**를 클릭하고 개체를 복사할 프로젝트 또는 폴더를 선택합니다.
5. **마침**을 클릭합니다.

태그

태그는 비즈니스 용도에 따라 도메인의 개체를 정의하는 메타데이터입니다. 비즈니스 용도에 따라 개체를 그룹화하기 위해 태그를 작성합니다.

태그를 작성한 후 태그를 하나 이상의 개체와 연결할 수 있습니다. 또한 태그와 개체 간의 연결을 제거할 수도 있습니다. 태그를 사용하여 도메인에서 태그와 연결된 개체를 검색할 수 있습니다. **Developer** 도구는 모든 태그의 용어집을 표시합니다.

예를 들어 XYZCorp_CustomerOrders라는 태그를 작성하고 XYZ Corporation의 고객 주문에 대한 정보가 포함된 테이블에 태그를 할당할 수 있습니다. 사용자는 XYZCorp_CustomerOrders 태그로 검색하여 이 태그와 연결된 테이블을 식별할 수 있습니다.

태그와 연결된 매핑을 배포할 때 태그는 데이터 통합 서비스에서 런타임 버전의 매핑으로 전달됩니다. 응용 프로그램 패치를 사용하여 배포된 매핑을 업데이트하는 경우 패치의 이름이 매핑의 런타임 버전과 태그로 연결됩니다.

참고: Developer tool에서 개체와 연결된 태그는 Analyst 도구에서 동일한 개체의 태그로 나타납니다.

태그 작성

태그를 작성하여 비즈니스 용도에 따라 개체를 정의하는 메타데이터를 추가합니다.

1. 다음 방법 중 하나를 사용하여 태그를 작성합니다.
 - **창 > 기본 설정**을 클릭합니다. **기본 설정** 대화 상자에서 **Informatica > 태그**를 선택합니다. 모델 리포지토리 서비스를 선택하고 **추가**를 클릭합니다.
 - 편집기에서 개체를 엽니다. **태그** 보기에서 **편집**을 클릭합니다. **개체에 대한 태그 할당** 대화 상자에서 **새로 만들기**를 클릭합니다.
2. 태그의 이름을 입력합니다.
3. 필요에 따라 설명을 입력합니다.
4. **확인**을 클릭합니다.

태그 할당

태그를 개체에 할당하여 개체를 메타데이터 정의와 연결합니다.

1. 편집기에서 개체를 엽니다.
2. **태그 보기**에서 **편집**을 클릭합니다.

개체에 대한 태그 할당 대화 상자가 나타납니다. **사용 가능한 태그** 영역에 리포지토리에 정의된 모든 태그가 표시됩니다. 이름 또는 설명으로 태그를 검색할 수 있습니다. **태그 할당** 영역에 열려 있는 개체 및 개체에 할당된 태그가 표시됩니다.

3. **사용 가능한 태그** 영역에서 태그를 선택합니다.
4. **태그 할당** 영역에서 개체를 선택합니다.
5. **할당**을 클릭합니다.
6. 개체에서 태그를 제거하려면 **사용 가능한 태그** 영역에서 태그를 선택하고 **태그 할당** 영역에서 개체를 선택한 후 **제거**를 클릭합니다.

태그 보기

개체에 할당된 모든 태그를 보거나 모델 리포지토리에 정의된 모든 태그를 볼 수 있습니다.

1. 개체에 할당된 태그를 보려면 편집기에서 개체를 엽니다.
2. **태그 보기**를 선택합니다.
태그 보기에 개체에 할당된 모든 태그가 표시됩니다.
3. 모델 리포지토리에 정의된 모든 태그를 보려면 **창 > 기본 설정**을 클릭합니다.
기본 설정 대화 상자가 표시됩니다.
4. **Informatica > 태그**를 선택합니다.

태그 영역에 모델 리포지토리에 정의된 모든 태그가 표시됩니다. 이름 또는 설명으로 태그를 검색할 수 있습니다.

제 2 장

모델 리포지토리

이 장에 포함된 항목:

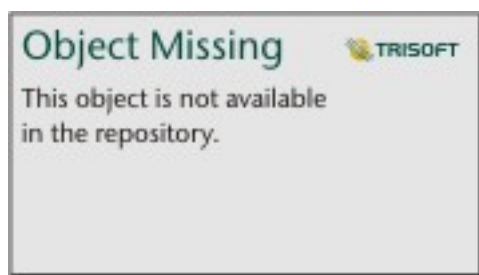
- [모델 리포지토리 개요, 32](#)
- [Informatica Developer의 개체, 33](#)
- [리포지토리 개체 잠금, 35](#)
- [버전 지정 개체를 사용하는 팀 기반 개발, 36](#)
- [모델 리포지토리에 연결, 40](#)
- [모델 리포지토리 서비스 새로 고침, 40](#)

모델 리포지토리 개요

모델 리포지토리는 프로젝트 및 폴더에 대한 메타데이터를 저장하는 관계형 데이터베이스입니다.

모델 리포지토리에 연결하여 실제 데이터 개체, 매핑, 프로필 및 기타 개체를 작성하고 편집합니다. 응용 프로그램에 개체를 포함한 다음 최종 사용자와 타사 도구가 개체에 액세스할 수 있도록 응용 프로그램을 배포합니다.

다음 이미지는 **Object Explorer** 보기에 열려 있는 mrs1이라는 이름의 모델 리포지토리를 보여줍니다.



모델 리포지토리 서비스에서 모델 리포지토리를 관리합니다. 모델 리포지토리에 액세스하는 모든 클라이언트 응용 프로그램과 응용 프로그램 서비스는 모델 리포지토리 서비스를 통해 연결합니다. 클라이언트 응용 프로그램에는 **Developer tool**과 **Analyst** 도구가 포함됩니다. 모델 리포지토리에 액세스하는 **Informatica** 서비스에는 모델 리포지토리 서비스, 분석 서비스, 데이터 통합 서비스가 포함됩니다.

Developer tool을 설정할 때 모델 리포지토리를 추가해야 합니다. **Developer tool**을 열 때마다 프로젝트 및 폴더에 액세스하기 위해 모델 리포지토리에 연결합니다.

개체를 편집할 때 모델 리포지토리는 배타적 편집이 가능하도록 개체를 잠급니다. 모델 리포지토리를 타사 버전 제어 시스템과 통합할 수도 있습니다. 버전 제어 시스템 통합을 사용하면 개체를 체크 아웃 및 체크인 하고, 개체의 체크 아웃을 실행 취소하고, 개체의 기록 버전을 보거나 검색할 수 있습니다.

Informatica Developer의 개체

Developer tool의 **Object Explorer** 보기에서 특정 개체를 작성하거나 관리하거나 볼 수 있습니다.

Developer tool에서 다음과 같은 모델 리포지토리 개체를 작성할 수 있습니다.

응용 프로그램

데이터 개체, 매핑, SQL 데이터 서비스, 웹 서비스 및 워크플로우를 포함할 수 있는 배포 가능한 개체입니다. 응용 프로그램을 작성, 편집 및 삭제할 수 있습니다.

데이터 서비스

데이터에 액세스하고 데이터를 변환하기 위해 실행할 수 있는 재사용 가능한 작업의 모음입니다. 데이터 서비스는 웹 서비스를 통해 액세스하거나 SQL 쿼리를 실행할 수 있는 데이터의 일관된 모델을 제공합니다. 데이터 서비스를 작성, 편집 및 삭제할 수 있습니다.

데이터 개체 작업

소스 또는 대상에서 특정 런타임 작업을 수행하는 데 필요한 속성이 포함되는 리포지토리 개체입니다. 일부 PowerExchange 어댑터 데이터 소스에 필요합니다.

폴더

모델 리포지토리 내 개체의 컨테이너입니다. 폴더를 사용하여 프로젝트의 개체를 체계화하고 폴더를 작성하여 비즈니스 요구에 따라 개체를 그룹화할 수 있습니다. 폴더를 작성, 편집 및 삭제할 수 있습니다.

논리적 데이터 개체

엔터프라이즈에서 논리적 항목을 설명하는 논리적 데이터 개체 모델의 개체입니다. 특성과 키로 구성되고 특성 간의 관계를 설명합니다. 논리적 데이터 개체 모델에서 논리적 데이터 개체를 작성, 편집 및 삭제할 수 있습니다.

논리적 데이터 개체 매핑

논리적 데이터 개체를 하나 이상의 실제 데이터 개체에 연결하는 매핑입니다. 변환 논리를 포함할 수 있습니다. 논리적 데이터 개체에 대해 논리적 데이터 개체 매핑을 작성, 편집 및 삭제할 수 있습니다.

논리적 데이터 개체 모델

논리적 데이터 개체를 포함하고 이러한 개체 간에 관계를 정의하는 데이터 모델입니다. 논리적 데이터 개체 모델을 작성, 편집 및 삭제할 수 있습니다.

매핑

데이터 변환 규칙을 정의하는 변환 개체로 연결되는 입력 및 출력의 집합입니다. 매핑을 작성, 편집 및 삭제할 수 있습니다.

맵렛

여러 매핑에서 사용하거나 규칙으로 유효성을 검사할 수 있는 변환 집합이 포함된 재사용 가능 개체입니다. 맵렛을 작성, 편집 및 삭제할 수 있습니다.

작업 매핑

웹 서비스 클라이언트에 대한 웹 서비스 작업을 수행하는 매핑입니다. 작업 매핑은 입력 변환, 출력 변환 및 여러 개의 결합 변환을 포함할 수 있습니다. 웹 서비스에서 작업 매핑을 작성, 편집 및 삭제할 수 있습니다.

실제 데이터 개체

리소스에서 읽거나 리소스를 조회하거나 리소스에 쓰는 데 사용되는 데이터의 실제 표현입니다. 실제 데이터 개체를 작성, 편집 및 삭제할 수 있습니다.

프로필

소스 데이터에서 패턴을 검색하는 규칙을 포함하는 개체입니다. 프로필을 실행하여 데이터 구조를 평가하고 데이터 열에 필요한 정보 유형이 있는지 확인합니다. 프로필을 작성, 편집 및 삭제할 수 있습니다.

참조 테이블

표준 버전의 데이터 값 집합과 찾으려고 할 수 있는 대체 버전의 값을 포함합니다. 참조 테이블을 보고 삭제할 수 있습니다.

규칙

프로필을 실행할 때 소스 데이터에 적용되는 조건을 정의하는 비즈니스 논리입니다. 규칙은 프로필에서 사용하는 미드스트림 맵셋입니다. 규칙을 작성, 편집 및 삭제할 수 있습니다.

규칙 사양

하나 이상의 비즈니스 규칙 논리를 포함하는 재사용 가능 개체입니다. **Analyst** 도구 사용자는 규칙 사양을 생성하고 규칙 사양을 모델 리포지토리에 저장합니다. **Object Explorer** 보기에서 규칙 사양을 선택하고 규칙 사양을 매핑으로 끌어다 놓을 수 있습니다.

Developer tool에서 규칙 사양은 읽기 전용 개체입니다. 규칙 사양 논리를 보거나 편집하려면 규칙 사양을 마우스 오른쪽 단추로 클릭하고 **Analyst** 도구에서 여는 옵션을 선택합니다.

참고: **Analyst** 도구 사용자는 규칙 사양에서 맵셋을 생성할 수도 있습니다. 규칙 사양과 해당하는 맵셋은 모델 리포지토리에서 종속된 개체입니다. 맵셋은 **Developer tool**에서 편집할 수 있습니다.

성과 기록표

프로필 결과에서 규칙의 출력 또는 소스의 열에 대한 유효한 값을 그래픽으로 표현한 것입니다. 성과 기록표를 작성, 편집 및 삭제할 수 있습니다.

변환

데이터를 생성, 수정 또는 전달하는 매핑의 리포지토리 개체입니다. 각 변환은 서로 다른 기능을 수행합니다. 변환은 재사용 가능하거나 재사용 불가능할 수 있습니다. 변환을 작성, 편집 및 삭제할 수 있습니다.

유형 정의 라이브러리

Spark 엔진에서 실행되는 매핑에 대한 복합 데이터 유형 정의를 저장하는 리포지토리 개체입니다. 복합 데이터 유형 정의는 구조체 데이터의 스키마입니다. 매핑 또는 맵셋에 대한 유형 정의 라이브러리 및 복합 데이터 유형 정의는 매핑 편집기의 아웃라인 보기에서 볼 수 있습니다.

가상 스키마

데이터베이스 구조를 정의하는 가상 데이터베이스의 스키마입니다. **SQL** 데이터 서비스에서 가상 스키마를 작성, 편집 및 삭제할 수 있습니다.

가상 저장 프로시저

SQL 데이터 서비스의 프로시저 또는 데이터 흐름 지침 집합입니다. 가상 스키마에서 가상 저장 프로시저를 작성, 편집 및 삭제할 수 있습니다.

가상 테이블

가상 데이터베이스의 테이블입니다. 가상 스키마에서 가상 테이블을 작성, 편집 및 삭제할 수 있습니다.

가상 테이블 매핑

가상 테이블을 대상으로 포함하는 매핑입니다. 가상 테이블의 가상 테이블 매핑을 작성, 편집 및 삭제할 수 있습니다.

워크플로우

비즈니스 프로세스를 정의하는 이벤트, 태스크 및 결정 사항 집합의 그래픽 표현입니다. 워크플로우를 작성, 편집 및 삭제할 수 있습니다.

개체 속성

프로젝트 또는 폴더의 속성이나 모델 리포지토리에 있는 다른 개체의 속성을 볼 수 있습니다.

속성 보기의 **일반** 탭에 개체 속성이 표시됩니다. 개체 속성으로는 리포지토리에서 개체의 이름, 속성 및 위치가 있습니다. 또한 개체의 속성에는 개체를 작성한 사용자와 마지막으로 업데이트한 사용자 및 이벤트가 발생한 시간도 있습니다.

개체 속성에 액세스하려면 **개체 탐색기** 보기에서 개체를 선택하고 **파일 > 속성**을 클릭합니다.

리포지토리 개체 잠금

개체 편집을 시작하면 모델 리포지토리에서 다른 사용자가 개체에 변경 내용을 저장할 수 없도록 개체를 잠급니다. 개체를 저장하면 잠금이 유지됩니다. 개체를 닫으면 리포지토리가 잠금을 해제합니다.

개체 잠금은 CI/CD 파이프라인에서 지속적인 통합의 일부입니다. CI/CD에 대한 자세한 내용은 [“CI/CD 개요” 페이지 169](#)를 참조하십시오.

잠금 관리

Developer tool이 예기치 않게 중지되면 모델 리포지토리가 개체 잠금을 유지합니다. 모델 리포지토리에 다시 연결할 때 사용자는 자신이 잠근 개체를 볼 수 있습니다. 개체 편집을 계속하거나 개체의 잠금을 해제할 수 있습니다.

개체를 잠갔던 개발자가 없고 다른 개발자가 개체를 개발하도록 할당된 경우 개체의 잠금을 해제하려고 할 수 있습니다. 잠긴 개체는 **잠긴 개체** 대화 상자를 통해 보고 잠금을 해제합니다.

잠긴 개체 대화 상자를 보려면 **파일 > 잠긴 개체**를 클릭합니다.

다음 이미지는 **잠긴 개체** 대화 상자를 보여 줍니다.



잠긴 개체 대화 상자에서 하나 이상의 개체를 잠금 해제할 수 있습니다. 다음 작업 중 하나를 수행하여 잠금 해제할 개체를 선택하십시오.

- 확인란을 사용하여 하나 이상의 개체를 선택합니다.
- 하나 이상의 행을 선택하고 **선택 사항 선택**을 클릭하여 선택한 각 개체의 확인란을 활성화합니다.

그러면 **잠금 해제**를 클릭하여 선택된 모든 개체를 잠금 해제할 수 있습니다.

팁: 작업이 중단된 경우, 잠갔던 개체를 보고 작업 중이던 개체를 식별할 수 있습니다.

잠금 관리 규칙 및 지침

개체 잠금을 관리할 때 다음 규칙과 지침을 고려하십시오.

- 모델 리포지토리는 개체를 열 때 개체를 잠그지 않습니다. 모델 리포지토리는 개체 편집을 시작한 이후에만 개체를 잠급니다. 예를 들어, 모델 리포지토리는 사용자가 편집 가능한 필드에 커서를 삽입하거나 매핑 개체를 연결할 때 매핑을 잠급니다.
- 둘 이상의 클라이언트 도구를 사용하여 개체를 개발할 수 있습니다. 예를 들어, 하나의 시스템에서 개체를 편집한 후 다른 시스템에서 개체를 열어 편집을 계속할 수 있습니다. 첫 번째 시스템으로 돌아갈 경우 개체 편집기를 닫고 다시 열어서 잠금을 회복해야 합니다. 사용자가 연 개체를 관리 권한이 있는 사용자가 잠금 해제하는 경우에도 동일한 원리가 적용됩니다.
- 편집을 위해 개체를 열어 놓았을 때 모델 리포지토리 서비스가 재시작되면 해당 개체에 대한 잠금이 손실됩니다. 개체에 대한 잠금이 회복될 때까지 다른 사용자가 해당 개체를 열고 편집할 수 있습니다. 개체에 대한 잠금을 회복하려면 개체에 변경 내용을 저장하고 닫은 다음 편집을 위해 개체를 다시 여십시오.
- 개체가 포함된 폴더를 삭제하면 어떤 개체도 삭제할 수 없으며 해당 폴더도 삭제할 수 없습니다. 예를 들어 개체의 잠금을 소유하지 않아서 개체를 삭제할 수 없는 경우 개체 및 폴더가 그대로 남아 있습니다.
- 두 명 이상의 개발자가 동시에 SQL 데이터 서비스 개체의 콘텐츠를 열고 편집할 수 있습니다. 예를 들어 사용자 A가 SQL 데이터 서비스를 열어 편집을 시작한 상태에서 사용자 B가 같은 개체를 열어 편집을 시작할 수 있습니다. 사용자 B가 사용자 A보다 먼저 개체를 저장한 후 닫으면 모델 리포지토리에서는 사용자 A가 개체를 저장할 때까지 사용자 A에게 잠재적인 충돌에 대해 알리지 않습니다. 이 경우 사용자 A는 SQL 데이터 서비스를 다른 이름으로 저장하여 변경 내용을 저장할 수 있습니다.
- 관리자는 잠근 개체에 대한 쓰기 권한을 취소하거나 다른 사용자에게 잠금을 재할당할 수 있습니다. 이 경우 개체를 편집하거나 저장할 수 없습니다. 다른 이름으로 개체를 저장할 수 있습니다.

버전 지정 개체를 사용하는 팀 기반 개발

팀 기반 개발은 모델 리포지토리와 타사 버전 제어 시스템의 통합입니다. 버전 제어 시스템은 개체의 여러 버전을 저장하고 각 버전에 버전 번호를 할당합니다.

개체 버전은 **Developer tool**을 통해 관리합니다. 개체 체크 아웃 및 체크 인, 개체의 기록 버전 보기 및 검색과 체크 아웃 실행 취소 같은 작업을 수행할 수 있습니다.

모델 리포지토리는 개발 팀의 다른 멤버가 덮어쓰지 못하도록 개체를 보호합니다. 다른 사용자가 체크 아웃한 개체를 열면 해당 개체를 체크 아웃한 사용자를 식별하는 알림을 받게 됩니다. 체크 아웃된 개체를 읽기 전용 모드로 열거나 다른 이름으로 저장할 수 있습니다.

버전 제어 시스템에 대한 연결이 활성화된 경우 모델 리포지토리는 각 개체의 최신 버전을 가집니다.

모델 리포지토리가 버전 제어 시스템에 대한 연결을 잃으면 체크 아웃된 개체의 상태가 유지됩니다. 버전 제어 시스템 연결이 다운되면 개체를 계속해서 열고, 편집하고, 저장하고, 닫을 수 있습니다. 모델 리포지토리는 개체 상태를 추적하고 유지 관리합니다.

연결이 복원되면 개체 체크 인, 개체 체크 아웃 실행 취소와 같은 버전 제어 시스템 관련 작업을 다시 시작할 수 있습니다. 연결이 다운된 동안 개체를 열어 편집한 경우 모델 리포지토리는 해당 개체를 사용자에게 체크 아웃합니다.

참고: SQL 데이터 서비스 개체는 버전 제어에 포함되지 않습니다.

개체 버전 관리는 **CI/CD** 파이프라인에서 지속적인 통합의 일부입니다. **CI/CD**에 대한 자세한 내용은 [“CI/CD 개요” 페이지 169](#)를 참조하십시오.

버전 지정 개체 관리

모델 리포지토리가 버전 제어 시스템과 통합되어 있는 경우 개체의 버전을 관리할 수 있습니다. 예를 들어 개체를 체크 아웃 및 체크 인하고, 체크 아웃을 실행 취소하고, 체크 아웃한 개체를 볼 수 있습니다.

다음 작업을 수행할 수 있습니다.

개체를 체크 아웃합니다.

개체를 체크 아웃하는 경우 개체를 체크 인하거나 체크 아웃을 실행 취소할 때까지 체크 아웃 상태를 유지합니다. **체크 아웃된 개체** 보기에서 체크 아웃한 개체를 볼 수 있습니다. 개체를 체크 아웃하려는 경우 개체에 대한 편집을 시작하거나 Object Explorer에서 개체를 마우스 오른쪽 단추로 클릭하고 체크 아웃을 선택할 수 있습니다.

개체의 체크아웃을 실행 취소합니다.

체크 아웃을 실행 취소할 때 개체를 변경하지 않으며 버전 번호나 버전 기록을 충분하지 않고 체크 인합니다. 체크 아웃한 후 개체에 적용한 모든 변경 내용은 손실됩니다. 체크아웃 실행 취소를 위해 **체크 아웃된 개체** 보기를 사용하거나 개체에서 오른쪽 클릭 메뉴를 사용할 수 있습니다. 예를 들어, 개체에 대한 변경 내용을 삭제하기 위해 체크 아웃을 실행 취소할 수 있습니다.

참고: 체크아웃된 개체를 이동한 후 체크아웃을 실행 취소하면 개체가 현재 위치에 남아 있고 버전 기록이 다시 시작됩니다. 체크아웃을 실행 취소해도 개체가 체크아웃 이전 위치로 복원되지 않습니다.

개체를 체크 인합니다.

개체를 체크 인할 때, 버전 제어 시스템이 버전 기록을 업데이트하고 버전 번호를 증분합니다. 최대 4KB 제한까지 체크 인 설명을 추가할 수 있습니다. 개체 체크 인을 위해 **체크 아웃된 개체** 보기를 사용하거나 개체에서 오른쪽 클릭 메뉴를 사용할 수 있습니다.

참고: Windows에서 경로 길이를 260바이트로 제한하기 때문에 Perforce 및 SVN 버전 제어 시스템에서도 파일 경로 길이를 제한합니다. 제한 길이를 초과하는 경로를 가진 파일을 체크 인하려는 경우 Perforce에서 오류가 반환됩니다. 260바이트 경로 길이는 시스템의 문자 설정에 따라 다양한 경로 길이를 생성할 수 있습니다.

개체를 삭제합니다.

먼저 버전 지정 개체를 체크 아웃해야 해당 개체를 삭제할 수 있습니다. 삭제 작업을 수행할 때 개체가 체크 아웃되어 있지 않은 경우 모델 리포지토리가 개체를 사용자에게 체크 아웃하고 삭제하도록 표시합니다. 삭제 작업을 완료하려면 개체를 체크 인해야 합니다.

버전 지정 개체를 삭제할 때 버전 제어 시스템은 모든 버전을 삭제합니다.

개체 삭제를 위해 **체크 아웃된 개체** 보기를 사용하거나 개체에서 오른쪽 클릭 메뉴를 사용할 수 있습니다.

체크 인 대화 상자

체크 인 대화 상자에 체크 아웃한 모든 개체가 표시됩니다.

체크 인 대화 상자를 사용하여 개체를 체크 인합니다. 또한 이 옵션을 사용하여 삭제하려고 표시한 개체를 삭제할 수 있습니다.

다음 이미지는 **체크 인** 대화 상자를 보여줍니다.



작업 유형 열에 모델 리포지토리 서비스가 체크 인한 각 항목에서 실행한 작업이 표시됩니다. 모델 리포지토리 서비스는 다음 작업 유형을 실행합니다.

작업 유형	설명
작성	개체가 새로 작성되었고 한 번도 체크 인되지 않았습니다. 해당 개체를 선택한 후 체크 인 을 클릭하면 모델 리포지토리 서비스에서 버전 제어 시스템에 개체를 체크 인하고 버전 번호 1을 할당합니다.
삭제	개체가 삭제용으로 표시되어 있습니다. 해당 개체를 선택한 후 체크 인 을 클릭하면 모델 리포지토리 서비스에서 개체를 삭제합니다.
편집	개체가 편집용으로 표시되어 있습니다. 해당 개체를 선택하고 체크 인 을 클릭하면 모델 리포지토리 서비스에서 버전 제어 시스템의 버전을 새 버전으로 바꿉니다.
이동	개체를 체크 아웃한 후 하나의 모델 리포지토리 프로젝트 또는 폴더에서 다른 프로젝트나 폴더로 개체를 이동했습니다. 해당 개체를 선택하고 체크 인 을 클릭하면 모델 리포지토리 서비스에서 버전 제어 시스템의 새 위치로 개체를 체크 인합니다.

위치 열에 개체의 현재 위치가 표시됩니다. 개체를 이동한 경우 **작업 유형**이 이동입니다.

팁: 버전 기록 보기에서 이동된 개체의 이전 위치를 볼 수 있습니다.

새 버전 열에는 체크 인을 완료한 후의 개체 버전 번호가 표시됩니다.

체크 인 대화 상자에서 하나 이상의 개체를 체크 인할 수 있습니다. 다음 작업 중 하나를 수행하여 체크 인할 개체를 선택하십시오.

- 확인란을 사용하여 하나 이상의 개체를 선택합니다.
- 하나 이상의 행을 선택하고 **선택 사항 선택**을 클릭하여 선택한 각 개체의 확인란을 활성화합니다.

원하는 경우 **체크 인 설명** 텍스트 창에 설명을 추가할 수 있습니다. 그런 다음 **체크 인**을 클릭하여 선택한 모든 개체를 체크 인할 수 있습니다.

버전 지정 개체 삭제

버전 지정 개체를 삭제하려는 경우 해당 개체를 삭제하도록 표시한 다음 체크 인합니다.

1. **개체 탐색기** 또는 **체크 아웃된 개체** 보기에서 개체를 마우스 오른쪽 단추로 클릭하고 **삭제**를 선택합니다.
작업 특성이 **편집**에서 **삭제**로 변경됩니다.
2. 개체를 선택하고 **체크 인**을 선택합니다.
모델 리포지토리가 개체를 삭제합니다.

개체의 기록 버전

버전 제어 시스템은 개체를 체크 인할 때마다 해당 개체의 사본을 저장합니다.

버전 제어 시스템의 개체의 버전을 저장하고 여기에 버전 번호를 할당합니다. 가장 큰 번호를 가진 버전이 가장 나중 버전입니다.

버전 기록 보기를 사용하여 개체의 버전 기록을 보고, 읽기 전용 보기에서 기록 버전을 열고, 개체의 기록 버전을 복원할 수 있습니다. 개체의 기록 버전을 복원하여 이전 기능을 복원할 수 있습니다.

개체의 기록 버전을 복원할 때 해당 버전은 최신 버전이 됩니다. 모델 리포지토리는 개체의 현재 버전을 기록 버전으로 바꾸고 개체를 사용자에게 체크 아웃합니다. 개체가 편집기에서 열리면 **Developer tool**이 개체를 복원된 버전으로 새로 고칩니다.

개체의 기록 버전 복원

버전 제어 시스템에서 개체의 기록 버전을 보거나 가져올 수 있습니다.

1. **개체 탐색기**에서 개체를 마우스 오른쪽 단추로 클릭하고 **버전 기록 보기**를 선택합니다.

개정 기록 보기가 열립니다.

2. 기록 버전을 보려면 **개정 기록** 보기에 나열된 버전 중 하나를 마우스 오른쪽 단추로 클릭하고 **버전 보기**를 선택합니다.

Developer tool이 개체의 읽기 전용 버전을 엽니다. **Developer tool**이 편집기 창의 제목 표시줄에 버전 번호를 나열합니다.

3. 기록 버전을 복원하려면 **개정 기록** 보기에 나열된 버전 중 하나를 마우스 오른쪽 단추로 클릭하고 **버전 복원**을 선택합니다.

편집기에서 개체가 열립니다. 개체를 편집하거나 개체를 편집하지 않고 체크 인할 수 있습니다.

4. 개체를 체크 인합니다.

개체의 복원된 버전은 모델 리포지토리와 버전 제어 시스템에서 최신 버전이 됩니다.

체크 아웃된 개체 보기

체크 아웃된 개체 보기는 체크 아웃한 모든 개체를 나열합니다.

체크 아웃된 개체 보기에서 다음 작업을 수행할 수 있습니다.

- 개체의 체크 아웃을 실행 취소합니다.
- 개체를 체크 인합니다.

작업 중 하나를 수행하려면 개체를 마우스 오른쪽 단추로 클릭한 후 작업을 선택합니다.

삭제용으로 표시된 개체를 **작업 유형**에서 삭제하려면 해당 개체를 마우스 오른쪽 단추로 클릭한 후 **체크 인**을 선택합니다.

버전 기록 보기

버전 기록 보기는 선택된 개체의 버전 기록을 표시합니다. 체크 인 설명을 읽고 개체 체크 인에 대한 사용자 정보를 볼 수 있습니다.

이 보기는 다음과 같은 정보를 표시합니다.

- 버전. 버전이 저장된 순서를 보여주는 정수입니다. 버전 제어 시스템은 가장 최근에 체크 인된 버전에 가장 큰 번호를 할당합니다.
- 이름. 개체의 이름입니다.

- 위치. 모델 리포지토리의 개체 경로입니다.
- 작업 유형. 개체를 체크 인할 때 사용자가 수행한 작업입니다. 개체를 체크 인할 때 개체를 추가, 편집 또는 삭제할 수 있습니다.
- 사용자. 개체를 체크 인한 팀원의 사용자 ID입니다.
- 버전 날짜. 사용자가 개체를 체크 인한 날짜와 시간입니다.
- 버전 설명. 선택된 버전에 대한 체크 인 설명입니다.

버전 기록을 보려면 **개체 탐색기**에서 개체를 마우스 오른쪽 단추로 클릭하고 **버전 기록 보기**를 선택합니다.

팀 기반 개발 문제 해결

팀 기반 개발과 관련된 기능을 사용할 경우 다음과 같은 문제 해결 팁을 고려하십시오.

너무 긴 개체 경로 이름에 대한 오류가 발생하여 Perforce 버전 제어 시스템에서 일부 개체를 체크 인하지 못합니다.

Windows OS에서는 파일 경로에 포함할 수 있는 문자 수가 제한되기 때문에 모델 리포지토리 개체의 경로 및 파일 이름이 길면 체크 인할 때 작업이 실패할 수 있습니다. 이 경우 "Submit aborted"라는 Perforce 오류 메시지가 나타나고 파일 경로가 내부 길이 제한을 초과한다고 표시됩니다.

이 문제를 해결하려면 모델 리포지토리에서 프로젝트, 폴더 및 개체 이름의 길이를 제한하십시오. 모든 인스턴스에서 짧은 이름을 사용하면 개체 경로 이름의 총 문자 수를 제한하는 데 도움이 됩니다.

모델 리포지토리에 연결

Developer tool을 열 때마다 프로젝트 및 폴더에 액세스하기 위해 모델 리포지토리에 연결합니다. 모델 리포지토리에 연결하면 모델 리포지토리를 관리하는 모델 리포지토리 서비스가 포함된 도메인에 액세스하기 위해 연결 정보를 입력합니다.

1. **개체 탐색기** 보기에서 모델 리포지토리를 마우스 오른쪽 단추로 클릭하고 **연결**을 클릭합니다.
리포지토리에 연결 대화 상자가 표시됩니다.
2. 도메인 사용자 이름과 암호를 입력합니다.
3. 네임스페이스를 선택합니다.
4. **확인**을 클릭합니다.
Developer tool이 모델 리포지토리에 연결합니다. Developer tool이 리포지토리의 프로젝트를 표시합니다.

모델 리포지토리 서비스 새로 고침

모델 리포지토리 서비스를 새로 고쳐 모델 리포지토리의 새 개체와 업데이트된 개체를 표시할 수 있습니다.

모델 리포지토리 서비스를 새로 고치려면 **개체 탐색기** 보기에서 모델 리포지토리 서비스를 마우스 오른쪽 단추로 클릭하고 **새로 고침**을 선택합니다.

제 3 장

Informatica Developer에서 검색

이 장에 포함된 항목:

- [Informatica Developer에서 검색 개요, 41](#)
- [도메인 검색, 41](#)
- [Business Glossary 검색, 43](#)
- [편집기 검색, 44](#)

Informatica Developer에서 검색 개요

Informatica Developer에서 검색을 수행하여 개체를 검색하고 비즈니스 용어를 조회할 수 있습니다.

Object Explorer 보기에 나타나는 도메인을 검색하여 개체 및 개체 속성의 최신 버전을 찾을 수 있습니다.

Developer tool에서 Business Glossary Desktop에 액세스하여 Developer tool 개체 이름을 비즈니스 용어로 조회할 수 있습니다. 또한 편집기에서 개체, 포트, 그룹, 식 및 특성을 찾을 수도 있습니다.

도메인 검색

도메인에서 개체 및 개체 속성을 검색할 수 있습니다.

검색 쿼리를 작성한 후 검색 결과를 필터링할 수 있습니다. 검색 결과를 보고 결과에서 개체를 선택하여 해당 내용을 확인할 수 있습니다. 검색 결과는 **검색** 보기에 나타납니다. 검색에서 2048개가 넘는 개체를 찾을 경우 결과를 표시할 수 없습니다. 결과에 2048개가 넘는 개체가 포함되어 검색이 실패하면 더 적은 수의 개체가 검색 기준과 일치하도록 검색 옵션을 변경합니다.

다음 표에는 개체를 검색하는 데 사용할 수 있는 검색 옵션이 나와 있습니다.

검색 옵션	설명
포함하는 텍스트	검색할 개체 또는 속성입니다. 정확한 문자열을 입력하거나 와일드카드를 사용합니다. 대/소문자를 구분하지 않습니다.
이름	해당 이름을 포함하는 하나 이상의 개체입니다. 정확한 문자열을 입력하거나 와일드카드를 사용합니다. 대/소문자를 구분하지 않습니다.

검색 옵션	설명
태그	태그를 사용하는 하나 이상의 개체입니다. 정확한 문자열을 입력하거나 와일드카드를 사용합니다. 대/소문자를 구분하지 않습니다.
검색 범위	검색할 하나 이상의 개체 유형입니다.
범위	선택한 작업 공간 또는 개체를 검색합니다.

검색 엔진은 도메인의 메타데이터를 인덱싱합니다. 메타데이터를 올바르게 인덱싱하기 위해 검색 엔진은 사용자가 인덱싱하려는 메타데이터의 언어에 적절한 검색 분석기를 사용합니다. **Developer tool**은 검색 엔진을 사용하여 디자인 타임 및 런타임 개체에 대해 검색을 수행합니다.

여러 가지 언어로 검색할 수 있습니다. 다른 언어로 검색하려면 관리자가 검색 분석기를 변경한 후 검색 분석기를 사용하도록 다음과 같은 모델 리포지토리를 구성해야 합니다.

- 디자인 타임 개체를 저장하는 모델 리포지토리.
- 런타임 개체가 배포된 데이터 통합 서비스와 연결된 모델 리포지토리.

참고: 개체 또는 개체 속성을 검색하려면 **Object Explorer** 보기에서 개체에 연결되어 있어야 합니다. 추가 개체로 검색을 확장하려면 개체가 포함되어 있는 런타임 응용 프로그램 또는 모델 리포지토리에 연결합니다.

검색 지침

도메인 검색 시 다음 지침을 참조하십시오.

- 디자인 타임 개체에 대해 검색을 수행하려면 먼저 개체를 저장해야 합니다.
- 검색은 포트와 동적 포트를 반환하지만 생성된 포트는 반환하지 않습니다.
- 매핑을 검색하면 검색 패턴과 유사한 이름을 가진 맵렛이 검색 결과에 나타납니다.
- 모델 리포지토리 서비스 또는 런타임 응용 프로그램을 새로 고치면 검색 보기에 검색 결과가 더 이상 나타나지 않습니다. 새로 고친 모델 리포지토리 서비스나 새로 고친 런타임 응용 프로그램에 기반한 결과를 얻으려면 도메인을 다시 검색해야 합니다.

개체 및 속성 검색

도메인에서 개체 및 속성을 검색합니다.

1. **검색 > 개체 검색**을 클릭합니다.
검색 대화 상자가 표시됩니다.
2. 검색할 개체 또는 속성을 입력합니다. 필요에 따라 와일드카드 문자를 포함합니다.
3. 개체의 속성을 검색하려는 경우 필요에 따라 쉽표로 구분하여 하나 이상의 이름 또는 태그를 입력합니다.
4. 원하는 경우 검색할 개체 유형을 선택합니다.
5. 선택한 작업 공간 또는 개체를 검색하도록 선택합니다.
6. **검색**을 클릭합니다.
검색 결과가 **검색** 보기에 나타납니다.
7. **검색** 보기에서 개체를 두 번 클릭하여 **편집기**에서 엽니다.
개체가 런타임 응용 프로그램의 일부인 경우 이 개체는 읽기 전용입니다.

Business Glossary 검색

Developer tool의 개체 이름이 갖는 의미를 Business Glossary Desktop에서 비즈니스 용어로 조회하면 해당하는 개체 이름의 비즈니스 요구 사항 및 현재 구현을 확인할 수 있습니다.

비즈니스 용어집은 비즈니스 언어를 사용하여 비즈니스 사용자에게 대한 개념을 정의하는 용어의 집합입니다. 비즈니스 용어는 개념에 대한 비즈니스 정의 및 사용법을 제공합니다. Business Glossary Desktop은 비즈니스 용어집을 호스팅하는 Metadata Manager 서비스에 연결하는 클라이언트입니다. Business Glossary Desktop을 사용하여 비즈니스 용어집에서 비즈니스 용어를 조회합니다.

Business Glossary Desktop이 시스템에 설치된 경우 Developer tool에서 개체를 선택하고 바로 가기 키 또는 검색 메뉴를 사용하여 비즈니스 용어집에서 개체의 이름을 조회할 수 있습니다. Object Explorer 보기 같은 Developer tool 보기에서 개체 이름을 조회하거나 편집기에서 열, 프로필 및 변환 포트의 이름을 조회할 수 있습니다.

예를 들어 개발자가 비즈니스 용어집에서 Developer tool의 Sales_Audit 데이터 개체에 해당하는 비즈니스 용어를 찾기를 원할 수 있습니다. 개발자는 비즈니스 용어 정보를 보고 Developer tool에서 Sales_Audit 개체의 비즈니스 요구 사항과 현재 구현을 이해할 수 있습니다. 따라서 개발자는 데이터 개체의 의미와 개체에 필요한 변경 사항을 파악할 수 있습니다.

Business Glossary Desktop 조회

Business Glossary Desktop은 비즈니스 용어집에서 개체 이름을 조회하고 해당 개체 이름과 일치하는 비즈니스 용어를 반환할 수 있습니다.

Business Glossary Desktop은 개체 이름이 하이픈, 밑줄 또는 대문자로 구분되어 있을 경우 개체를 두 개로 분할합니다.

예를 들어 개발자가 Sales_Audit라는 데이터 개체를 조회할 경우 Business Glossary Desktop은 Sales_Audit를 검색 창자에 표시하지만 이름을 Sales 및 Audit로 분할하고 두 개의 비즈니스 용어를 조회합니다.

비즈니스 용어 조회

Developer 도구의 개체 이름을 Business Glossary Desktop에서 비즈니스 용어로 조회하면 해당하는 개체 이름의 비즈니스 요구 사항 및 현재 구현을 확인할 수 있습니다.

Business Glossary Desktop을 시스템에 설치해야 합니다.

1. 개체를 선택합니다.
2. Business Glossary Desktop을 열기 위해 바로 가기 키를 사용할지 아니면 검색 메뉴를 사용할지 선택합니다.
 - 바로 가기 키를 사용하려면 다음과 같은 바로 가기 조합을 사용합니다.
CTRL+Shift+F
 - 검색 메뉴를 사용하려면 **검색 > Business Glossary**를 클릭합니다.

Business Glossary Desktop이 나타나고 개체 이름과 일치하는 비즈니스 용어가 표시됩니다.

비즈니스 용어 조회를 위한 바로 가기 키 사용자 지정

바로 가기 키를 사용자 지정하여 Business Glossary Desktop을 여는 키 조합을 변경할 수 있습니다.

1. Developer 도구 메뉴에서 **창 > 기본 설정 > 일반 > 키**를 클릭합니다.
2. 명령 목록에서 **Business Glossary 검색**을 찾거나 검색하려면 다음 옵션 중 하나를 선택합니다.

- 키를 검색하려면 검색 상자에 **Business Glossary** 검색을 입력합니다.
 - 키를 스크롤하려면 스크롤하여 **명령** 열 아래에서 **Business Glossary 검색** 명령을 찾습니다.
3. **Business Glossary 검색** 명령을 클릭합니다.
 4. **명령 바인딩 해제**를 클릭합니다.
 5. **바인딩** 필드에 키 조합을 입력합니다.
 6. **적용**을 클릭하고 **확인**을 클릭합니다.

편집기 검색

편집기에서 열려 있는 개체, 포트, 그룹, 식 및 특성을 찾을 수 있습니다. **Developer tool**은 열려 있는 편집기 내에서 해당 개체를 강조 표시합니다. 개체가 모델 리포지토리에 있을 필요가 없습니다.

편집기 아래에 찾기 필드를 표시하려면 **편집 > 찾기/교체**를 선택합니다. 개체를 찾으려면 찾을 개체에 대한 검색 문자열과 유형을 입력합니다. 찾을 수 있는 개체 유형은 편집기별로 다릅니다. 개체 유형을 지정하지 않으면 **Developer tool**이 변환에서 검색 문자열을 찾습니다.

포트, 열 또는 특성을 검색하는 경우에는 데이터 유형도 선택할 수 있습니다. 예를 들어 "_ID"라는 문자열이 포함된 이름이 있는 정수 또는 **bigint** 포트를 찾을 수 있습니다.

다음 표에는 각 편집기에서 찾을 수 있는 개체의 유형이 나와 있습니다.

편집기	개체 유형
매핑	매핑 개체, 식, 그룹, 포트 및 복합 데이터 유형 정의
맵렛	맵렛 개체, 식, 그룹, 포트 및 복합 데이터 유형 정의
논리적 데이터 개체 모델	논리적 데이터 개체 및 특성
실제 데이터 개체 읽기 또는 쓰기 매핑	매핑 개체 및 열
SQL 데이터 서비스	가상 테이블 및 특성
가상 저장 프로시저	변환, 식, 그룹 및 포트
가상 테이블 매핑	가상 테이블 매핑 개체, 식, 그룹 및 포트
웹 서비스 작업 매핑	웹 서비스 작업 매핑 개체, 식, 그룹 및 포트
워크플로우	워크플로우 개체
<i>참고: 모델 리포지토리는 생성된 포트의 이름을 저장하지 않으므로 편집기에서 해당 포트를 검색할 수 없습니다.</i>	

Developer tool은 검색 문자열을 찾을 경우 개체 위치를 표시합니다. 또한 검색 문자열이 일치하는 개체를 강조 표시합니다. 검색 문자열이 매핑 편집기의 아이콘화된 변형에서 일치하는 경우 **Developer tool**은 아이콘화된 변형을 강조 표시합니다.

다음과 같은 옵션을 선택하여 찾기 결과를 탐색할 수 있습니다.

- 다음 일치. 검색 문자열의 다음 일치 항목을 찾습니다.

- 이전 일치. 검색 문자열의 이전 일치 항목을 찾습니다.
- 모두 강조 표시. 검색 문자열의 모든 일치를 강조 표시합니다.
- 아이콘화된 변형 확장. 검색 문자열이 일치하는 모든 아이콘화된 변형을 확장합니다.

제 4 장

연결

이 장에 포함된 항목:

- [연결 개요, 46](#)
- [연결 탐색기 보기, 48](#)
- [연결 관리, 48](#)
- [연결 전환, 51](#)
- [타사 JDBC 드라이버, 54](#)
- [환경 SQL, 54](#)

연결 개요

연결은 도메인 구성 리포지토리에 연결을 정의하는 리포지토리 개체입니다.

데이터 개체를 가져오고, 데이터를 미리 보고, 데이터를 프로파일링하고, 매핑을 실행하기 위한 연결을 작성합니다. **Developer tool**은 사용자가 데이터 개체를 가져올 때 이 연결을 사용합니다. 사용자가 데이터를 미리 보거나, 매핑을 실행하거나, 웹 서비스를 사용할 때 데이터 통합 서비스는 연결을 사용합니다.

참고: **Developer tool**은 플랫폼 파일 데이터 개체를 가져오거나 플랫폼 파일 데이터를 미리 보거나, 읽거나, 쓰는 데 연결을 사용하지 않습니다.

Developer tool은 도메인 구성 리포지토리에서 연결을 저장합니다. **Developer tool**에서 작성하는 연결은 **Analyst** 도구 및 **Administrator** 도구에서 사용할 수 있습니다.

기본 설정 대화 상자 또는 **연결 탐색기** 보기에서 연결을 작성하고 관리합니다.



1. 연결 탐색기 보기
2. 연결 작성
3. 연결 표시

연결을 작성한 후 연결에 대해 다음과 같은 작업을 수행할 수 있습니다.

연결을 편집합니다.

연결 이름 및 설명을 변경할 수 있습니다. 또한 사용자 이름, 암호 및 연결 문자열 같은 연결 세부 정보도 편집할 수 있습니다.

데이터 통합 서비스는 연결 ID로 연결을 식별합니다. 따라서 연결 이름을 변경할 수 있습니다. 연결 이름을 바꾸면 Developer tool이 해당 연결을 사용하는 개체를 업데이트합니다.

배포된 응용 프로그램 및 매개 변수 파일은 연결 ID가 아닌 이름으로 연결을 식별합니다. 따라서 연결 이름을 바꿀 경우 해당 연결을 사용하는 모든 응용 프로그램을 다시 배포해야 합니다. 또한 해당 연결 매개 변수를 사용하는 모든 매개 변수 파일도 업데이트해야 합니다.

연결을 복사합니다.

연결을 복사하여 다른 연결과 유사한 연결을 작성할 수 있습니다. 예를 들어 사용자 이름 및 암호만 다른 Oracle 연결 두 개를 작성할 수 있습니다.

연결 삭제

연결을 삭제하면 해당 연결을 사용하는 개체가 더 이상 유효하지 않습니다. 실수로 연결을 삭제할 경우 삭제한 연결과 동일한 연결 ID를 사용하여 다른 연결을 작성하면 해당 연결을 다시 작성할 수 있습니다.

연결 목록을 새로 고칩니다.

연결 목록을 새로 고쳐 도메인의 최신 연결 목록을 표시할 수 있습니다. Administrator 도구 또는 Analyst 도구에서 사용자가 연결을 추가 또는 삭제하거나 연결 이름을 바꾼 후 연결 목록을 새로 고칩니다.

연결 유형

연결을 통해 데이터 소스에서 데이터를 읽고 데이터 소스에 데이터를 쓸 수 있습니다.

비윈시 환경, Hadoop 및 Databricks에 액세스하도록 연결을 생성합니다. HBase, HDFS, Hive 소스 또는 Hadoop 환경의 대상에 액세스하는 경우에도 그러한 연결을 생성해야 합니다. Developer tool, Administrator 도구 및 infacmd를 사용하여 연결을 생성할 수 있습니다.

다음과 같은 종류의 연결을 생성할 수 있습니다.

Hadoop 연결

Hadoop 환경에서 매핑을 실행하려면 Hadoop 연결을 생성합니다.

HBase 연결

HBase에 액세스하려면 HBase 연결을 생성합니다. HBase 연결은 NoSQL 연결입니다.

HDFS 연결

Hadoop 클러스터의 Hadoop 파일 시스템에서 데이터를 읽고 쓰려면 HDFS 연결을 생성합니다.

Hive 연결

소스 또는 대상으로 Hive에 액세스하려면 Hive 연결을 생성합니다. 매핑이 윈시 또는 Hadoop 환경에 대해 활성화된 경우에는 소스로 Hive에 액세스할 수 있습니다. 매핑이 Blaze 엔진에서 실행되는 경우에는 대상으로 Hive에 액세스할 수 있습니다.

JDBC 연결

Sqoop을 통해 관계형 데이터를 가져오고 내보내려면 JDBC 연결을 생성하고 해당 연결에서 Sqoop 속성을 구성합니다.

Databricks 연결

Databricks 환경에서 매핑을 실행하려면 Databricks 연결을 생성합니다.

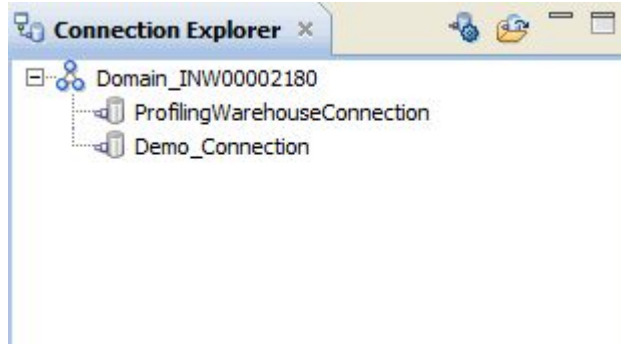
참고: 소셜 미디어 웹 사이트나 Teradata와 같은 기타 소스 또는 대상에 대한 연결 생성과 관련한 자세한 내용은 해당하는 PowerExchange 어댑터 사용자 가이드를 참조하십시오.

모든 데이터 소스의 지원 버전에 대한 자세한 내용은 [Informatica Product Availability Matrix](#)를 참조하십시오.

연결 탐색기 보기

연결 탐색기 보기를 사용하여 관계형 데이터베이스 연결을 확인하고 관계형 데이터 개체를 생성할 수 있습니다.

다음 그림은 Developer 도구의 연결 탐색기를 보여 줍니다.



연결 탐색기 보기에서 다음과 같은 작업을 수행할 수 있습니다.

- 연결을 보기에 추가합니다. **연결 선택** 단추를 클릭하여 연결 탐색기 보기에 추가할 하나 이상의 연결을 선택합니다.
- 관계형 데이터베이스에 연결합니다. 데이터베이스를 마우스 오른쪽 단추로 클릭하고 **연결**을 클릭합니다.
- 관계형 데이터베이스에서 연결을 끊습니다. 데이터베이스를 마우스 오른쪽 단추로 클릭하고 **연결 끊기**를 클릭합니다.
- 관계형 데이터 개체를 작성합니다. 관계형 데이터베이스에 연결한 후 데이터베이스를 확장하여 테이블을 표시합니다. 테이블을 마우스 오른쪽 단추로 클릭하고 **프로젝트에 추가**를 클릭하여 새 관계형 데이터 개체 대화 상자를 엽니다.
- 연결을 새로 고칩니다. 연결을 마우스 오른쪽 단추로 클릭하고 **새로 고침**을 클릭합니다.
- 기본 스키마만 표시합니다. 연결을 마우스 오른쪽 단추로 클릭하고 **기본 스키마만 표시**를 클릭합니다. 기본값은 활성화됩니다.
- 연결 탐색기 보기에서 연결을 삭제합니다. 연결은 모델 리포지토리에서 그대로 유지됩니다. 연결을 마우스 오른쪽 단추로 클릭하고 **삭제**를 클릭합니다.

연결 관리

기본 설정 대화 상자 또는 연결 탐색기 보기에서 연결을 작성하고 관리합니다.

연결 작성

계산 클러스터, 데이터베이스, 엔터프라이즈 응용 프로그램, 파일 시스템, 비관계형 데이터베이스, NoSQL 데이터베이스, 소셜 미디어 응용 프로그램 또는 웹 서비스 액세스를 위한 연결을 생성합니다. 실제 데이터 개체를 가져오거나, 데이터를 미리 보거나, 데이터를 프로파일링하거나, 매핑을 실행하기 전에 연결을 작성하십시오.

1. **창 > 기본 설정**을 클릭합니다.
2. 작성할 연결 유형을 선택합니다.
 - 웹 서비스 연결이 아닌 연결을 선택하려면 **Informatica > 연결**을 선택합니다.
 - 웹 서비스 연결을 선택하려면 **Informatica > 웹 서비스 > 연결**을 선택합니다.
3. **사용 가능한 연결** 목록에서 도메인을 확장합니다.
4. **사용 가능한 연결** 목록에서 연결 유형을 선택하고 **추가**를 클릭합니다.
새 **<연결 유형> 연결** 대화 상자가 나타납니다.
5. 다음 정보를 입력합니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 128자를 초과하거나 공백을 포함하거나 다음과 같은 특수 문자를 포함할 수 없습니다. ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결에 대한 설명입니다(선택 사항).
위치	연결이 존재하는 도메인입니다.
유형	Oracle, Twitter 또는 웹 서비스 등의 특정한 연결 유형입니다.

6. **다음**을 클릭합니다.
7. 연결 속성을 구성합니다.
8. **연결 테스트**를 클릭하여 연결 속성을 제대로 입력했고 데이터베이스, 응용 프로그램, 파일 시스템 또는 URI에 연결할 수 있는지 확인합니다.
참고: Databricks에 대한 **연결 테스트**는 지원되지 않습니다.
9. **마침**을 클릭합니다.
연결을 작성한 후 **연결 탐색기** 보기에 추가할 수 있습니다.

연결 편집

연결 이름, 설명 및 연결 속성을 편집할 수 있습니다.

1. **창 > 기본 설정**을 클릭합니다.
2. 편집할 연결 유형을 선택합니다.
 - 웹 서비스 연결이 아닌 연결을 선택하려면 **Informatica > 연결**을 선택합니다.
 - 웹 서비스 연결을 선택하려면 **Informatica > 웹 서비스 > 연결**을 선택합니다.
3. **사용 가능한 연결** 목록에서 도메인을 확장합니다.

4. **사용 가능한 연결**에서 연결을 선택하고 **편집**을 클릭합니다.
연결 편집 대화 상자가 표시됩니다.
5. 필요한 경우 연결 이름과 설명을 편집합니다.
참고: 연결 이름을 변경할 경우 해당 연결을 사용하는 모든 응용 프로그램을 다시 배포해야 합니다. 또한 해당 연결 매개 변수를 사용하는 모든 매개 변수 파일도 업데이트해야 합니다.
6. **다음**을 클릭합니다.
7. 필요한 경우 연결 속성을 편집합니다.
8. **연결 테스트**를 클릭하여 연결 속성을 제대로 입력했고 데이터베이스에 연결할 수 있는지 확인합니다.
9. **확인**을 클릭하여 **연결 편집** 대화 상자를 닫습니다.
10. **확인**을 클릭하여 **기본 설정** 대화 상자를 닫습니다.

연결 복사

연결을 도메인 내에서 복사하거나 다른 도메인으로 복사할 수 있습니다.

1. **창 > 기본 설정**을 클릭합니다.
2. 복사할 연결 유형을 선택합니다.
 - 웹 서비스 연결이 아닌 연결을 선택하려면 **Informatica > 연결**을 선택합니다.
 - 웹 서비스 연결을 선택하려면 **Informatica > 웹 서비스 > 연결**을 선택합니다.
3. **사용 가능한 연결** 목록에서 도메인을 확장합니다.
4. **사용 가능한 연결**에서 연결을 선택하고 **복사**를 클릭합니다.
연결 복사 대화 상자가 표시됩니다.
5. 연결 이름 및 ID를 입력하고 도메인을 선택합니다.
이름 및 ID는 도메인에서 고유해야 합니다.
참고: Hadoop, HDFS, HBase, Hive 또는 Databricks 연결을 다른 도메인에 복사하기 전에 도메인에서 연결과 연관된 클러스터 구성과 일치하는 이름의 클러스터 구성을 생성합니다. 예를 들어 연결이 XYZ라는 이름의 클러스터 구성과 연관된 경우 대상 도메인에서 XYZ라는 이름의 클러스터 구성을 생성합니다.
6. **확인**을 클릭하여 **연결 복사** 대화 상자를 닫습니다.
7. **확인**을 클릭하여 **기본 설정** 대화 상자를 닫습니다.

연결 삭제

기본 설정 대화 상자를 통해 연결을 삭제하면 Developer 도구가 모델 리포지토리에서 연결을 제거합니다.

1. **창 > 기본 설정**을 클릭합니다.
2. 삭제할 연결 유형을 선택합니다.
 - 웹 서비스 연결이 아닌 연결을 선택하려면 **Informatica > 연결**을 선택합니다.
 - 웹 서비스 연결을 선택하려면 **Informatica > 웹 서비스 > 연결**을 선택합니다.
3. **사용 가능한 연결** 목록에서 도메인을 확장합니다.
4. **사용 가능한 연결**에서 연결을 선택하고 **제거**를 클릭합니다.
5. **확인**을 클릭하여 **기본 설정** 대화 상자를 닫습니다.

연결 목록 새로 고침

연결 목록을 새로 고쳐 도메인의 최신 연결 목록을 표시합니다.

1. 창 > 기본 설정을 클릭합니다.
2. 새로 고칠 연결 유형을 선택합니다.
 - 웹 서비스 연결이 아닌 연결을 선택하려면 **Informatica > 연결**을 선택합니다.
 - 웹 서비스 연결을 선택하려면 **Informatica > 웹 서비스 > 연결**을 선택합니다.
3. 사용 가능한 연결 목록에서 도메인을 선택합니다.
4. 새로 고침을 클릭합니다.
5. 사용 가능한 연결 목록에서 도메인을 확장하여 최신 연결 목록을 표시합니다.
6. 확인을 클릭하여 기본 설정 대화 상자를 닫습니다.

연결 전환

다른 관계형 데이터베이스 연결을 사용하도록 관계형 데이터 개체 또는 사용자 지정된 데이터 개체의 연결을 전환할 수 있습니다. 여러 데이터 개체에 대한 연결을 동시에 전환할 수도 있습니다. 연결을 전환하면 새 연결을 사용하도록 각 매핑을 업데이트할 필요가 없기 때문에 시간과 노력을 줄일 수 있습니다.

연결을 전환하면 Developer tool이 해당 데이터 개체에 기반한 모든 읽기, 쓰기 및 조회 변환에서 데이터 개체에 대한 연결 세부 정보를 업데이트합니다. Developer tool은 새 연결이 가리키는 데이터베이스에 기반하여 데이터 개체의 데이터베이스 유형도 업데이트합니다.

연결을 다음 연결 유형 중 하나로 전환할 수 있습니다.

- IBM DB2
- Microsoft SQL Server
- ODBC
- Oracle

Developer tool은 연결을 전환할 때 메타데이터 호환성을 검사하지 않습니다. 따라서 연결을 전환하기 전에 새 연결이 가리키는 데이터베이스에 연결을 전환하는 데이터 개체와 동일한 열 및 메타데이터가 포함된 테이블이 있는지 확인해야 합니다. 그렇지 않을 경우 데이터 손실 또는 불일치가 발생할 수 있습니다.

예

Developer tool에서 Oracle 관계형 데이터 개체를 작성한 후 여러 매핑에 쓰기 변환으로 추가했습니다.

Oracle 데이터베이스를 IBM DB2 데이터베이스로 마이그레이션합니다. Developer tool에서 기존 매핑을 업데이트하여 데이터를 IBM DB2 데이터베이스에 쓰려고 합니다.

각 매핑에서 쓰기 변환의 기반이 되는 데이터 개체를 바꾸는 대신 Oracle 데이터 개체의 연결을 IBM DB2 연결로 전환할 수 있습니다. 그러면 데이터를 쓰려는 IBM DB2 데이터베이스를 새 연결이 가리킵니다. Oracle 관계형 데이터 개체가 포함된 매핑을 실행하면 데이터 통합 서비스가 새 IBM DB2 연결을 사용하여 매핑을 실행합니다.

연결을 전환하기 전에

연결을 전환하기 전에 다음과 같은 요구 사항이 충족되는지 확인하십시오.

- 연결을 전환하려는 데이터 개체에 대해 쓰기 권한을 가지고 있어야 합니다.

- 새 연결이 가리키는 데이터베이스에 연결을 전환하려는 데이터 개체와 동일한 열 및 메타데이터가 포함된 테이블이 있어야 합니다. 테이블에 포함된 열의 전체 자릿수와 소수 자릿수가 데이터 개체의 전체 자릿수와 소수 자릿수보다 작으면 데이터가 손실되거나 일치하지 않을 수 있습니다.
- 연결을 전환하려는 데이터 개체에 저장되지 않은 변경 내용이 없어야 합니다. 저장되지 않은 변경 내용이 있으면 **Developer tool**이 연결을 전환하지 않습니다.
- 연결을 전환하려는 사용자 지정된 데이터 개체에 서로 다른 데이터베이스 유형의 관계형 데이터 개체가 포함되어 있으면 안 됩니다. 데이터베이스 유형이 서로 다른 관계형 데이터 개체가 포함되어 있으면 연결을 전환할 때 **Developer tool**에서 오류를 표시하지 않습니다. 그러나 해당 사용자 지정된 데이터 개체를 사용하여 매핑을 실행하면 매핑이 실패합니다.
- 원래 연결과 새 연결이 가리키는 테이블의 이름이 대/소문자를 포함하여 정확하게 일치해야 합니다. 연결에서 따옴표로 묶은 식별자 및 대/소문자가 혼합된 테이블 이름을 지원하는 경우, **Developer tool**은 테이블 이름이 대/소문자를 구분하는 것으로 처리합니다.
- 버전 지정 리포지토리에 개체가 저장된 경우에는 연결을 전환하려는 데이터 개체가 체크 아웃되어 있어야 합니다.

연결 전환

다른 관계형 데이터베이스 연결을 사용하도록 관계형 데이터 개체 또는 사용자 지정된 데이터 개체의 연결을 전환하고, 새 연결을 사용하도록 기존 매핑을 동시에 업데이트할 수 있습니다.

1. **Object Explorer** 보기에서 연결을 전환하려는 데이터 개체를 마우스 오른쪽 단추로 클릭합니다.

특정 프로젝트 또는 여러 프로젝트의 서로 다른 폴더에서 데이터 개체를 여러 개 선택할 수 있습니다.

연결을 선택한 후 해당 연결을 사용하는 관계형 데이터 개체 및 사용자 지정된 데이터 개체 전체에 대해 연결을 동시에 전환할 수도 있습니다.

2. **연결 전환**을 클릭합니다.

연결 전환 대화 상자가 나타나고, 선택한 데이터 개체가 표시됩니다.



3. **추가**를 클릭하여, 연결을 전환하려는 데이터 개체의 목록을 업데이트합니다.

개체를 제거하려면 원하는 개체를 선택한 후 **제거**를 클릭합니다.

4. **새 연결** 필드 옆의 **찾아보기**를 클릭합니다.

Developer tool이 데이터 개체에 대해 사용할 수 있는 연결을 표시합니다.

5. 데이터 개체에 대해 사용하려는 새 연결을 선택하고 **확인**을 클릭합니다.

6. **연결 전환**을 클릭합니다.

새 연결이 가리키는 데이터베이스에 연결을 전환하는 데이터 개체와 동일한 열 및 메타데이터가 있는 테이블이 포함되었는지 묻는 메시지가 표시됩니다.

7. **확인**을 클릭하여 데이터 개체의 연결을 전환합니다.

연결이 전환되었음을 알리는 메시지가 나타납니다. **Developer tool**이 해당 데이터 개체에 연결된 모든 **Developer tool** 개체에서 데이터 개체의 데이터베이스 유형 및 연결 세부 정보를 업데이트합니다. 데이터 개체가 포함된 매핑을 실행하면 데이터 통합 서비스가 새 연결을 사용합니다.

연결을 전환한 후에

연결을 전환한 후에는 필요한 경우 데이터 개체 속성을 확인하고 속성을 수동으로 편집할 수 있습니다.

연결을 전환한 후 다음과 같은 태스크를 수행하십시오.

- 데이터 유형 매핑을 확인합니다.
- 테이블 소유자 이름을 확인합니다.
- 조회 변환을 확인합니다.
- 힌트를 다시 구성합니다.
- 데이터 개체를 동기화합니다.

데이터 유형 매핑 확인

연결을 전환하면 **Developer tool**은 원래 연결과 새 연결이 가리키는 데이터베이스 사이에 가장 잘 일치하는 데이터 유형을 식별합니다. **Developer tool**은 데이터 개체의 데이터 유형을 설정하고, 해당 데이터 개체에 기반한 매핑 내의 변환을 적절하게 설정합니다. 매핑을 실행하기 전에 데이터 유형을 확인하고, 필요한 경우 수동으로 업데이트하십시오.

예를 들어 **Oracle** 연결을 **Microsoft SQL Server** 연결로 전환한다고 가정해 보겠습니다. 고정 전체 자릿수를 가진 데이터 유형의 경우 기본적으로 **Developer tool**은 **Microsoft SQL Server** 데이터베이스의 전체 자릿수에 기반하여 전체 자릿수를 설정합니다. 그러나 가변 전체 자릿수를 가진 데이터 유형의 경우 **Developer tool**은 **Oracle** 데이터베이스에 기반하여 전체 자릿수와 소수 자릿수를 설정합니다. 시간대가 포함된 타임스탬프 같은 데이터 유형의 경우 **Developer tool**은 데이터 유형을 **Varchar (0,0)** 또는 **Microsoft SQL Server** 데이터베이스가 **Varchar**에 대해 지원하는 동등한 데이터 유형으로 설정합니다.

테이블 소유자 이름 확인

연결을 전환하면 **Developer tool**은 데이터 개체의 테이블 소유자 이름을 유지합니다. 필요한 경우 데이터 개체 속성에서 테이블 소유자 이름을 수동으로 편집할 수 있습니다.

테이블 소유자 이름을 빈 값으로 설정한 후 연결을 전환하면 새 연결이 가리키는 데이터베이스의 기본 스키마 또는 **Public** 스키마에 테이블이 있는 경우에만 데이터를 미리 볼 수 있습니다. 테이블이 그 이외의 스키마에 있으면 데이터 미리보기가 실패합니다. 데이터를 미리 보려면 데이터 개체의 런타임 속성에서 테이블 소유자 이름을 수동으로 업데이트해야 합니다.

조회 변환 확인

연결을 전환한 후에는 필요한 경우, 조회 캐시를 다시 빌드하고 데이터 개체에 기반한 조회 변환에 대해 조회 조건을 확인할 수 있습니다.

다음 태스크를 수행합니다.

조회 캐시 다시 빌드

조회 캐시를 유지하면서 연결된 데이터 개체에 대한 연결을 전환하도록 데이터 통합 서비스를 구성한 경우에는 조회 변환을 업데이트하여 새 연결을 기반으로 조회 캐시를 다시 빌드해야 합니다. 이렇게 하지 않으면 매핑을 실행했을 때 다른 데이터베이스 연결을 사용하여 캐시 파일이 작성되었음을 알리는 오류가 발생합니다.

조회 조건 확인

연결을 전환하면 **Developer tool**은 원래 연결과 새 연결이 가리키는 데이터베이스 사이에 가장 잘 일치하는 데이터 유형을 식별하고 데이터 유형을 적절하게 설정합니다. 연결을 전환한 후에는 데이터 유형의 변경 때문에 조회 조건이 유효하지 않을 수 있습니다. 이 경우에는 조회 조건을 확인하고 수동으로 업데이트해야 합니다.

예를 들어 IBM DB2 소스 테이블, 조회 테이블 및 대상 테이블이 포함된 매핑을 작성한다고 가정해 보겠습니다. 조회 테이블 및 소스 테이블의 정수 열에 대해 조회 조건을 구성합니다. 조회 테이블의 연결을 IBM DB2에서 Oracle로 전환하면 조회 테이블에서 정수 열의 변환 데이터 유형이 10진수로 변경됩니다. 이 경우 정수 열을 10진수 열과 비교할 수 없기 때문에 조회 조건이 유효하지 않습니다.

힌트 다시 구성

사용자 지정된 데이터 개체의 연결을 전환하면 Developer tool에서는 해당 사용자 지정된 데이터 개체에 대해 구성된 힌트를 유지하지 않습니다. 따라서 힌트를 수동으로 다시 구성해야 합니다.

데이터 개체 동기화

연결을 전환하면 Developer tool은 활성 참조 키 제약 조건만 유지합니다.

여러 데이터 개체의 연결을 동시에 전환할 경우에는 데이터 개체를 동기화하여 키 관계를 정확하게 유지해야 합니다.

서로 순환 참조 키 제약 조건을 가진 여러 테이블이 데이터 개체에 포함되어 있는 경우에 테이블 중 일부의 연결을 다른 데이터베이스 유형으로 전환하면 Developer 테이블은 아무런 오류를 표시하지 않고 연결을 전환합니다. 하지만 데이터 개체의 키 관계를 확인하면 Developer tool이 원래 데이터베이스를 참조하는 키 관계를 표시합니다. 키 관계를 업데이트하여 새 데이터베이스를 가리키려면 데이터 개체를 동기화해야 합니다.

타사 JDBC 드라이버

JDBC를 통해 소스, 대상 및 조회에 연결하려면 타사 공급업체의 JDBC Type 4 드라이버를 설치하고 구성합니다.

Developer tool에서 메타데이터를 가져오려면 타사 JDBC 드라이버 jar 파일을 다음 위치에 복사합니다.

<Informatica 설치 디렉터리>\clients\externaljdbcjars

데이터 미리보기, 프로필 또는 매핑을 실행하려면 JDBC 드라이버 jar 파일을 다음 위치에 복사합니다.

<Informatica 설치 디렉터리>\externaljdbcjars

환경 SQL

통합 서비스는 자동 커밋 모드에서 환경 SQL을 실행하고 SQL을 실행한 후 트랜잭션을 닫습니다. 전체 읽기 또는 쓰기 프로세스 중에 열려 있는 트랜잭션에 종속되지 않는 SQL 명령을 사용하십시오. 예를 들어, 소스 데이터베이스가 읽기 전용 모드로 설정되어 있고 트랜잭션을 읽기 전용으로 설정하는 환경 SQL 문을 소스 연결에서 작성한 경우, 통합 서비스는 SQL을 실행한 후 커밋을 실행하고 소스를 읽기 전용 모드에서 읽을 수 없습니다.

연결 환경 SQL 또는 트랜잭션 환경 SQL을 구성할 수 있습니다.

소스, 대상, 조회 및 저장 프로시저 연결에 대해 환경 SQL을 사용하십시오. SQL 구문이 올바르지 않은 경우 통합 서비스는 데이터베이스에 연결하지 않고 세션이 실패합니다.

참고: 연결 개체에 "환경 SQL"이 있는 경우 연결에서 "연결 환경 SQL"을 사용합니다.

연결 환경 SQL

이 사용자 지정 SQL 문자열은 후속 트랜잭션에 대한 환경을 설정합니다. 통합 서비스에서 데이터베이스에 연결할 때마다 연결 환경 SQL을 실행합니다. 대상 연결에서 연결 환경 SQL을 구성하고 파이프라인에 대해 파티션 3개를 구성한 경우, 통합 서비스는 대상 데이터베이스에 대한 연결마다 한 번씩 SQL을 세 번 실행합니다. 전체 읽기 또는 쓰기 프로세스 중에 열려 있는 트랜잭션에 종속되지 않는 SQL 명령을 사용하십시오.

예를 들어, 연결 기간에 대한 따옴표 붙은 식별자 매개 변수를 설정하려면 다음 SQL 문을 사용하십시오.

```
SET QUOTED_IDENTIFIER ON
```

다음 상황에서 SQL 문을 사용하십시오.

- 큰따옴표가 개체 식별자가 되도록 연결 환경을 설정하고자 합니다.
- 대상 로드 유형을 일반적으로 구성하고 Microsoft SQL Server 대상 이름에 공백이 포함됩니다.

트랜잭션 환경 SQL

이 사용자 지정 SQL 문자열에서 환경을 설정하지만, 통합 서비스는 각 트랜잭션의 시작에서 트랜잭션 환경 SQL을 실행합니다.

전체 읽기 또는 쓰기 프로세스 중에 열려 있는 트랜잭션에 종속되는 SQL 명령을 사용하십시오. 예를 들어, 다음 문을 트랜잭션 환경 SQL로 사용하여 세션의 문자 처리 방법을 수정할 수 있습니다.

```
ALTER SESSION SET NLS_LENGTH_SEMANTICS=CHAR
```

각 트랜잭션 전에 이 명령을 실행해야 합니다. 각 연결마다 매개 변수를 한 번 설정하는 것이 충분하지 않기 때문에 명령이 연결 환경 SQL에 적절하지 않습니다.

환경 SQL 구성 지침

SQL 문을 작성할 때 다음 지침을 고려하십시오.

- 연결 개체와 연결된 데이터베이스에 유효한 SQL 명령을 입력할 수 있습니다. 데이터베이스에서 중첩된 설명을 허용하더라도 통합 서비스는 중첩된 설명을 허용하지 않습니다.
- SQL 편집기에서 SQL을 입력할 경우 SQL 문을 입력하십시오.
- 세미콜론을 사용하여 여러 개의 문을 구분합니다.
- 통합 서비스는 /* ... */ 내에 포함된 세미콜론은 무시합니다.
- 주석 외부에서 세미콜론을 사용해야 하는 경우 역슬래시(\)를 사용하여 이스케이프할 수 있습니다.
- 환경 SQL에서 매개 변수 및 변수를 사용할 수 있습니다. 매개 변수 파일에 정의할 수 있는 모든 매개 변수 또는 변수 유형을 사용할 수 있습니다. SQL 문 내에서 매개 변수 또는 변수를 입력하거나, 매개 변수 또는 변수를 환경 SQL로 사용할 수 있습니다. 예를 들어, 세션 매개 변수 \$ParamMyEnvSQL을 연결 또는 트랜잭션 환경 SQL로 사용하고, \$ParamMyEnvSQL을 매개 변수 파일의 SQL 문으로 설정할 수 있습니다.
- DB2 연결에 대한 연결 환경 SQL에서 sqlid를 사용하여 테이블 소유자 이름을 구성할 수 있습니다. 그러나 대상 인스턴스의 테이블 소유자 이름이 환경 SQL의 SET sqlid 문을 재정의합니다. SET sqlid 문에 지정된 테이블 소유자 이름을 사용하려면 대상 이름 접두사의 이름을 입력하지 마십시오.

제 5 장

실제 데이터 개체

이 장에 포함된 항목:

- [실제 데이터 개체 개요, 56](#)
- [실제 데이터 개체 유형, 57](#)
- [관계형 데이터 개체, 57](#)
- [사용자 지정된 데이터 개체, 60](#)
- [대상 테이블 작성 또는 대체, 65](#)
- [사용자 지정 쿼리, 68](#)
- [비관계형 데이터 개체, 79](#)
- [WSDL 데이터 개체, 81](#)
- [동기화, 84](#)
- [실제 데이터 개체 문제 해결, 86](#)

실제 데이터 개체 개요

실제 데이터 개체는 데이터 소스에서 읽고 데이터 소스를 조회하고 데이터 소스에 쓰는 데 사용되는 데이터의 실제 표현입니다. 데이터 개체 소스가 변경될 경우 실제 데이터 개체를 동기화할 수 있습니다. 실제 데이터 개체를 동기화하면 **Developer tool**이 개체 메타데이터를 다시 가져옵니다.

프로젝트 또는 폴더에서 실제 데이터 개체를 작성할 수 있습니다. 프로젝트 및 폴더의 실제 데이터 개체는 재사용 가능한 개체입니다. 실제 데이터 개체를 모든 유형의 매핑, 맵셋 또는 프로파일에서 사용할 수 있지만 매핑, 맵셋 또는 프로파일에서 이러한 데이터 개체를 변경할 수는 없습니다. 실제 데이터 개체를 업데이트하려면 프로젝트 또는 폴더 내에서 개체를 편집해야 합니다.

매핑, 맵셋 또는 프로파일에 실제 데이터 개체를 포함할 수 있습니다. 실제 데이터 개체를 읽기, 쓰기 또는 조회 변환으로 매핑 또는 맵셋에 추가할 수 있습니다. 실제 데이터 개체를 논리적 데이터 개체 매핑에 추가하여 논리적 데이터 개체를 매핑할 수 있습니다.

또한 **SQL** 데이터 서비스를 정의하는 경우 실제 데이터 개체를 가상 테이블 매핑에 포함할 수 있습니다. 웹 서비스를 정의하는 경우 실제 데이터 개체를 작업 매핑에 포함할 수 있습니다.

실제 데이터 개체 유형

데이터를 읽거나 데이터를 쓰려는 데이터 소스의 유형을 기반으로 다른 유형의 실제 데이터 개체를 작성할 수 있습니다.

실제 데이터 개체에는 다음과 같은 유형이 포함됩니다.

관계형 데이터 개체

관계형 테이블, 보기 또는 동의어를 소스로 사용하는 실제 데이터 개체입니다. 예를 들어 **Oracle** 보기에서 관계형 데이터 개체를 작성할 수 있습니다.

데이터 유형에 따라 관계형 데이터 개체를 소스, 대상 또는 조회 변환으로 매핑 또는 맵셋에 추가할 수 있습니다.

사용자 지정된 데이터 개체

한 개 또는 여러 개의 관계형 리소스 또는 관계형 데이터 개체를 소스로 사용하는 실제 데이터 개체입니다. 관계형 리소스에는 테이블, 보기 및 동의어가 있습니다. 예를 들어 기본 키-외래 키 관계가 있는 두 개의 **Microsoft SQL Server** 테이블에서 사용자 지정된 데이터 개체를 작성할 수 있습니다.

재사용 가능한 데이터 개체에서 데이터를 조인하거나, 행을 필터링하거나, 포트를 정렬하거나, 사용자 지정 쿼리를 실행하려는 경우 사용자 지정된 데이터 개체를 작성합니다.

비관계형 데이터 개체

비관계형 데이터베이스 리소스를 소스로 사용하는 실제 데이터 개체입니다. 예를 들어 **VSAM** 소스에서 비관계형 데이터 개체를 작성할 수 있습니다.

플랫 파일 데이터 개체

플랫 파일을 소스로 사용하는 실제 데이터 개체입니다. 구분자로 분리된 플랫 파일 또는 고정 너비 플랫 파일에서 플랫 파일 데이터 개체를 작성할 수 있습니다.

WSDL 데이터 개체

WSDL 파일을 소스로 사용하는 실제 데이터 개체입니다.

Informatica PowerExchange® 어댑터는 **SAP**, **Salesforce** 및 **Netezza**와 같은 실제 데이터 개체를 생성하는 데 사용할 수 있는 많은 데이터 소스에 대한 액세스도 제공합니다.

관계형 데이터 개체

관계형 데이터 개체는 관계형 테이블, 보기 또는 동의어를 소스로 가지고 있는 실제 데이터 개체입니다. 매핑, 맵셋 또는 프로필에 관계형 데이터 개체를 포함할 수 있습니다.

관계형 데이터 개체는 데이터베이스 리소스를 설명합니다. 관계형 데이터 개체를 리포지토리에 추가한 경우 소스 데이터베이스에서 가져옵니다. 리포지토리로 가져온 후 관계형 데이터 개체 정의를 변경할 수 있습니다. 포트를 추가 및 제거하고, 기본 키를 정의하고, 리포지토리의 여러 관계형 데이터 개체 간에 관계를 구성할 수 있습니다. 기본 연결, 데이터베이스 소유자 및 리소스 이름을 변경하거나 매개 변수화할 수 있습니다.

관계형 데이터 개체를 매핑, 맵셋 또는 프로필에 추가한 경우 읽기 또는 쓰기 개체를 작성할지 여부를 표시할 수 있습니다. 개체가 소스, 대상 또는 조회 테이블인지에 따라 다른 런타임 속성을 구성할 수 있습니다.

다음 그림은 편집기에 있는 샘플 관계형 데이터 개체를 보여줍니다.



소스 데이터베이스에 키 관계가 존재하는지 여부에 관계없이 관계형 데이터 개체 간에 기본 키-외래 키 관계를 작성할 수 있습니다.

여러 관계형 데이터 개체를 매핑 또는 맵셋에 소스로 추가할 수 있습니다. 여러 관계형 데이터 개체를 동시에 추가할 경우 **Developer tool**에서 다음 방법 중 하나로 개체를 추가하라는 메시지를 표시합니다.

- 관계형 데이터 개체로. **Developer tool**은 여러 관계형 리소스가 포함된 하나의 읽기 변환을 작성합니다. 읽기 변환에는 사용자 지정된 데이터 개체와 동일한 기능이 포함됩니다.
- 독립 데이터 개체로. **Developer tool**에서 각각의 관계형 데이터 개체에 대해 한 개의 읽기 변환을 작성합니다. 읽기 변환에는 관계형 데이터 개체와 동일한 기능이 포함됩니다.

다음과 같은 유형의 관계형 데이터 개체를 가져올 수 있습니다.

- DB2 for i5/OS
- DB2 for z/OS
- HAWQ
- IBM DB2
- JDBC
- Microsoft SQL Server
- Netezza
- ODBC
- Oracle
- SAP HANA

관계형 데이터 개체 가져오기

관계형 데이터 개체를 가져와서 매핑, 맵셋 또는 프로필에 추가할 수 있습니다.

관계형 데이터 개체를 가져오려면 먼저 데이터베이스에 대한 연결을 구성해야 합니다.

1. **Object Explorer** 보기에서 프로젝트나 폴더를 선택합니다.
2. **파일 > 새로 만들기 > 데이터 개체**를 클릭하십시오.
새로 만들기 대화 상자가 나타납니다.
3. **관계형 데이터 개체**를 선택하고 **다음**을 클릭합니다.
새 관계형 데이터 개체 대화 상자가 나타납니다.
4. 연결 옵션 옆에서 **찾아보기**를 클릭하고 데이터베이스에 대한 연결을 선택합니다.
5. **기존 리소스에서 데이터 개체 작성**을 클릭합니다.
6. **리소스 옵션** 옆에서 **찾아보기**를 클릭합니다.
리소스 선택 대화 상자가 나타납니다.

7. 가져오려는 테이블, 보기 또는 동의어를 선택합니다.
8. 데이터 개체를 필터링하려면 **필터** 섹션에서 이름을 입력하고 **검색**을 클릭합니다.
 검색 쿼리에서 물음표(?)를 단일 문자 와일드카드로 사용하거나 별표(*)를 여러 문자 와일드카드로 사용할 수 있습니다.
참고: 최대 2만 개의 검색 결과가 표시됩니다.
9. 가져오려는 리소스와 동일한 이름을 가진 실제 데이터 개체가 이미 있는 경우 **가져오기 충돌 해결** 대화 상자가 나타납니다. 가져오기 충돌의 해결 방법을 선택합니다. 다른 이름을 가진 실제 데이터 개체를 생성하거나, 기존 개체를 다시 사용하거나, 기존 개체를 바꿀 수 있습니다.
모든 충돌에 적용을 선택하여 동일한 솔루션을 모든 가져오기 충돌에 적용합니다.
10. 실제 데이터 개체의 이름을 입력합니다.
11. 위치 옵션 옆에서 **찾아보기**를 클릭하고 관계형 데이터 개체를 가져오려는 프로젝트를 선택합니다.
12. **마침**을 클릭합니다.
개체 탐색기 보기에서 프로젝트 또는 폴더의 **실제 데이터 개체** 아래에 데이터 개체가 나타납니다.

키 관계

관계형 데이터 개체 간에 키 관계를 작성할 수 있습니다. 관계형 데이터 개체를 사용자 지정된 데이터 개체에서 소스로 사용하거나 매핑 또는 맵셋에서 읽기 변환으로 사용하는 경우 키 관계를 통해 관계형 데이터 개체를 조인할 수 있습니다.

관계형 데이터 개체를 가져오는 경우 **Developer** 도구가 데이터베이스에 정의된 기본 키 정보를 유지합니다. 관련 관계형 데이터 개체를 동시에 가져오는 경우에도 **Developer** 도구가 외래 키 및 키 관계를 유지합니다. 하지만 관련 관계형 데이터 개체를 따로따로 가져오는 경우에는 개체를 가져온 후 키 관계를 다시 작성해야 합니다.

관계형 데이터 개체 간에 키 관계를 작성하려면 먼저 참조되는 개체에서 기본 키를 작성합니다. 그런 다음 관계형 데이터 개체에서 외래 키를 포함하는 관계를 작성합니다.

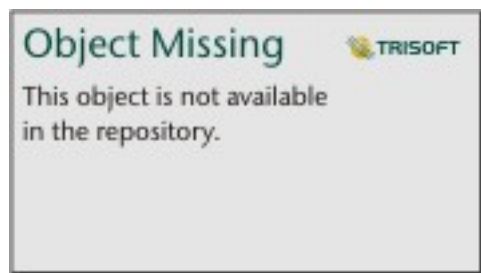
작성하는 키 관계는 관계형 데이터 개체 메타데이터에 존재합니다. 소스 관계형 리소스를 변경할 필요가 없습니다.

관계형 데이터 개체에 키 작성

관계형 데이터 개체에서 각 행을 식별하는 키 열을 작성합니다. 각 관계형 데이터 개체에서 한 개의 기본 키를 작성할 수 있습니다.

1. 관계형 데이터 개체를 엽니다.
2. **키** 보기를 선택합니다.

다음 그림은 편집기에서 열린 샘플 관계형 데이터 개체의 **키** 보기를 보여 줍니다.



3. **추가**를 클릭합니다.
새 키 대화 상자가 나타납니다.

4. 키 이름을 입력합니다.
5. 키가 기본 키이면 **기본 키**를 선택합니다.
6. 키 열을 선택합니다.
7. **확인**을 클릭합니다.
8. 관계형 데이터 개체를 저장합니다.

관계형 데이터 개체 간에 관계 작성

관계형 데이터 개체 간에 키 관계를 작성할 수 있습니다. 관계형 데이터 개체와 사용자 지정된 데이터 개체 간에 키 관계를 작성할 수는 없습니다.

참조하는 관계형 데이터 개체에 기본 키가 있어야 합니다.

1. 외래 키를 작성하려는 관계형 데이터 개체를 엽니다.
2. **관계** 보기를 선택합니다.
3. **추가**를 클릭합니다.
새 **관계** 대화 상자가 나타납니다.
4. 외래 키의 이름을 입력합니다.
5. 참조되는 관계형 데이터 개체에서 기본 키를 선택합니다.
6. **확인**을 클릭합니다.
7. **관계** 속성에서 외래 키 열을 선택합니다.
8. 관계형 데이터 개체를 저장합니다.

사용자 지정된 데이터 개체

사용자 지정된 데이터 개체는 하나 이상의 관계형 리소스를 사용하는 실제 데이터 개체입니다. 데이터 통합 서비스가 소스 데이터를 읽을 때 데이터를 조인하거나, 행을 필터링하거나, 포트를 정렬하거나, 사용자 지정 쿼리를 실행하려는 경우 사용자 지정된 데이터 개체를 작성합니다. 사용자 지정된 데이터 개체를 매핑, 맵렛 또는 프로필에서 재사용할 수 있습니다.

사용자 지정된 데이터 개체를 프로젝트 및 폴더에 작성할 수 있습니다. 사용자 지정된 데이터 개체를 매핑, 맵렛 또는 프로필에서 변경할 수는 없습니다. 프로젝트 또는 폴더에서 사용자 지정된 데이터 개체를 변경하면 Developer 도구가 이 개체를 사용하는 모든 매핑, 맵렛 및 프로필에서 개체를 업데이트합니다.

다음 그림은 편집기에서 열린 샘플 사용자 지정된 데이터 개체를 보여 줍니다.



사용자 지정된 데이터 개체를 작성하여 다음과 같은 태스크를 수행할 수 있습니다.

- 데이터 통합 서비스가 소스 데이터를 읽기 위해 실행하는 기본 쿼리를 대체하는 사용자 지정 쿼리를 작성합니다. 기본 쿼리는 데이터 통합 서비스가 소스에서 읽는 각 열을 참조하는 **SELECT** 문입니다.
- 데이터 개체에 대한 매개 변수를 정의합니다. 사용자 지정된 데이터 개체에서 매개 변수를 정의하고 할당하여 연결을 표현할 수 있습니다. 연결 이름, 테이블 소유자 및 테이블 이름에 대한 매개 변수를 정의할 수 있습니다. 사용자 지정된 데이터 개체를 사용하는 매핑을 실행하는 경우 런타임에 연결 매개 변수에 대해 여러 가지 값을 정의할 수 있습니다.
- 동일한 소스 데이터베이스에서 시작되는 소스 데이터를 조인합니다. 데이터베이스에 기본 키-외래 키 관계가 존재하는지 여부에 관계없이 기본 키-외래 키 관계를 사용하여 여러 테이블을 조인할 수 있습니다.
- 개체를 소스와 동기화할 때 키 관계를 유지합니다. 여러 테이블을 포함하는 사용자 지정된 데이터 개체를 작성하고 데이터베이스에 없는 키 관계를 정의하는 경우 데이터 개체를 동기화할 때 키 관계를 유지할 수 있습니다.
- 소스에서 고유 값을 선택합니다. 고유 항목 선택을 선택하면 데이터 통합 서비스가 기본 SQL 쿼리에 **SELECT DISTINCT** 문을 추가합니다.
- 데이터 통합 서비스가 소스 데이터를 읽을 때 행을 필터링합니다. 필터 조건을 포함시키면 데이터 통합 서비스가 기본 쿼리에 **WHERE** 절을 추가합니다.
- 정렬된 포트를 지정합니다. 정렬된 포트에 대한 번호를 지정하면 데이터 통합 서비스가 **ORDER BY** 절을 기본 SQL 쿼리에 추가합니다.
- 기본 내부 조인 대신 외부 조인을 지정합니다. 사용자 정의 조인을 포함하는 경우 데이터 통합 서비스가 SQL 쿼리의 메타데이터가 지정하는 조인 정보를 대체합니다.
- 사전 및 사후 매핑 SQL 명령을 추가합니다. 데이터 통합 서비스는 소스를 읽기 전에 소스 데이터베이스에 대해 사전 매핑 SQL 명령을 실행합니다. 또한 데이터 통합 서비스는 대상에 쓰기 전에 소스 데이터베이스에 대해 사후 매핑 SQL 명령을 실행합니다.

다음과 같은 유형의 연결 및 개체에서 사용자 지정된 데이터 개체를 작성할 수 있습니다.

- DB2 i5/OS 연결
- DB2 z/OS 연결
- IBM DB2 연결
- JDBC 연결
- Microsoft SQL Server 연결
- ODBC 연결
- Oracle 연결
- 관계형 데이터 개체

또한 사용자 지정 SQL 쿼리를 통해 소스를 사용자 지정된 데이터 개체에 추가할 수 있습니다.

키 관계

소스가 관계형 리소스인 경우 사용자 지정된 데이터 개체 내 소스 간에 키 관계를 작성할 수 있습니다. 키 관계를 통해 사용자 지정된 데이터 개체 내에서 소스를 조인할 수 있습니다.

참고: 사용자 지정된 데이터 개체가 관계형 데이터 개체를 소스로 사용하는 경우에는 사용자 지정된 데이터 개체 내에 키 관계를 작성할 수 없습니다. 대신 관계형 데이터 개체 간에 키 관계를 작성해야 합니다.

관계형 리소스를 사용자 지정된 데이터 개체로 가져오는 경우 **Developer** 도구가 데이터베이스에 정의된 기본 키 정보를 유지합니다. 관련 관계형 리소스를 사용자 지정된 데이터 개체로 동시에 가져오는 경우에도 **Developer** 도구가 키 관계 정보를 유지합니다. 하지만 관련 관계형 리소스를 따로따로 가져오는 경우에는 개체를 사용자 지정된 데이터 개체로 가져온 후 키 관계를 다시 작성해야 합니다.

사용자 지정된 데이터 개체 내 소스 간에 키 관계가 존재할 경우 데이터 통합 서비스가 각 소스의 관련 키를 기반으로 소스를 조인합니다. 기본 조인은 **WHERE** 절에서 다음과 같은 구문을 사용하는 내부 동등 조인입니다.

```
Source1.column_name = Source2.column_name
```

사용자 정의 조인을 입력하거나 사용자 지정 쿼리를 작성하여 기본 조인을 재정의할 수 있습니다.

사용자 지정된 데이터 개체에서 키 관계를 작성하려면 먼저 참조되는 소스 변환에서 기본 키를 작성합니다. 그런 다음 소스 변환에서 외래 키를 포함하는 관계를 작성합니다.

작성하는 키 관계는 사용자 지정된 데이터 개체 메타데이터에 존재합니다. 소스 관계형 리소스를 변경할 필요가 없습니다.

사용자 지정된 데이터 개체 쓰기 속성

데이터 통합 서비스는 관계형 리소스에 데이터를 쓸 때 쓰기 속성을 사용합니다. 쓰기 속성을 편집하려면 **쓰기** 보기에서 입력 변환을 선택한 다음 **고급** 속성을 선택합니다.

다음 테이블에는 사용자 지정된 데이터 개체에 대해 구성하는 쓰기 속성이 설명되어 있습니다.

속성	설명
Hive 대상 파티션 잘라내기	Hive 대상에서 데이터가 삽입되는 파티션을 덮어씁니다. 이 옵션을 활성화하려면 대상 테이블을 잘라내는 옵션도 선택해야 합니다. 외부의 분할된 테이블 및 분할되지 않은 테이블을 잘라낼 수 있습니다. 기본값은 비활성화됩니다.
로드 유형	대상 로드 유형입니다. 일반 또는 대량을 선택합니다. 일반을 선택할 경우 데이터 통합 서비스가 대상을 일반적으로 로드합니다. DB2, Sybase, Oracle 또는 Microsoft SQL Server에 로드하는 경우 대량을 선택할 수 있습니다. 다른 데이터베이스 유형에 대해 대량을 선택하면 데이터 통합 서비스가 일반 로드로 되돌립니다. 대량 로드를 사용하면 매핑 성능을 향상시킬 수 있지만 데이터베이스 로깅이 발생하지 않으므로 복구 기능이 제한됩니다. 대량 로드를 사용하여 Oracle 대상에 쓰는 경우 Oracle 데이터베이스에서 제약 조건을 비활성화하여 성능을 최적화할 수 있습니다. 매핑에 업데이트 전략 변환이 포함되는 경우에는 일반 모드를 선택하십시오. 일반 모드를 선택하고 Microsoft SQL Server 대상 이름에 공백이 포함되는 경우 다음 환경 SQL을 연결 개체에 구성합니다. SET QUOTED_IDENTIFIER ON
업데이트 재정의	대상에 대한 기본 UPDATE 문을 재정의합니다.
삭제	삭제 플래그가 지정된 모든 행을 삭제합니다. 기본값은 활성화됩니다.
삽입	삽입 플래그가 지정된 모든 행을 삽입합니다. 기본값은 활성화됩니다.
대상 스키마 전략	관계형 또는 Hive 대상 테이블에 대한 대상 스키마 전략의 유형입니다. 다음과 같은 대상 스키마 전략 중 하나를 선택할 수 있습니다. - RETAIN - 기존 대상 스키마 유지. 데이터 통합 서비스는 기존 대상 스키마를 유지합니다. - CREATE - 런타임 시 테이블 생성 또는 바꾸기. 데이터 통합 서비스가 식별하는 대상 테이블에 따라 런타임 시 대상 테이블을 삭제하고 테이블로 바꿉니다. - 매개 변수 할당. 대상 스키마 전략의 값을 나타내는 매개 변수를 할당한 후 런타임 시 매개 변수를 변경할 수 있습니다.

속성	설명
생성 또는 바꾸기에 대한 DDL 쿼리	데이터 통합 서비스가 대상 테이블을 생성하거나 바꿀 때 기준으로 사용하는 DDL 쿼리입니다. 이 옵션은 CREATE - 런타임 시 테이블 생성 또는 바꾸기 대상 스키마 전략을 선택하는 경우 적용 가능합니다.
대상 테이블 잘라내기	데이터를 로드하기 전에 대상을 잘라냅니다. 기본값은 비활성화됩니다.
업데이트 전략	기존 행에 대한 업데이트 전략입니다. 다음 전략 중 하나를 선택할 수 있습니다. - 업데이트 시 업데이트. 데이터 통합 서비스가 업데이트 플래그가 지정된 모든 행을 업데이트합니다. - 삽입 시 업데이트. 데이터 통합 서비스가 업데이트 플래그가 지정된 모든 행을 삽입합니다. 삽입 대상 옵션도 선택해야 합니다. - 업데이트 기타 항목 삽입. 업데이트 플래그가 지정된 행이 대상에 존재하는 경우 데이터 통합 서비스가 해당 행을 업데이트한 다음 삽입이 표시된 나머지 행을 삽입합니다. 삽입 대상 옵션도 선택해야 합니다.
PreSQL	소스를 읽기 전에 대상 데이터베이스에 대해 데이터 통합 서비스가 실행하는 SQL 명령. Developer tool은 SQL의 유효성을 검사하지 않습니다.
PostSQL	대상에 쓴 후에 대상 데이터베이스에 대해 데이터 통합 서비스가 실행하는 SQL 명령. Developer tool은 SQL의 유효성을 검사하지 않습니다.

사용자 지정된 데이터 개체 작성

사용자 지정된 데이터 개체를 작성하여 매핑, 랩셋 또는 프로필에 추가할 수 있습니다. 사용자 지정된 데이터 개체를 작성한 후에는 소스를 개체에 추가합니다.

1. **개체 탐색기** 보기에서 프로젝트나 폴더를 선택합니다.
2. **파일 > 새로 만들기 > 데이터 개체**를 클릭하십시오.
새로 만들기 대화 상자가 나타납니다.
3. **관계형 데이터 개체**를 선택하고 **다음**을 클릭합니다.
새 관계형 데이터 개체 대화 상자가 나타납니다.
4. 연결 옵션 옆에서 **찾아보기**를 클릭하고 데이터베이스에 대한 연결을 선택합니다.
5. **사용자 지정된 데이터 개체 작성**을 클릭합니다.
6. 사용자 지정된 데이터 개체의 이름을 입력합니다.
7. 위치 옵션 옆에서 **찾아보기**를 클릭하고 사용자 지정된 데이터 개체를 작성하려는 프로젝트를 선택합니다.
8. **마침**을 클릭합니다.
개체 탐색기 보기에서 프로젝트 또는 폴더의 실제 데이터 개체 아래에 사용자 지정된 데이터 개체가 나타납니다.

사용자 지정된 데이터 개체에 소스 추가 관계형 리소스 또는 관계형 데이터 개체를 소스로 추가할 수 있습니다. 또한 사용자 지정 **SQL** 쿼리를 통해 소스를 추가할 수 있습니다.

사용자 지정된 데이터 개체에 관계형 리소스 추가

사용자 지정된 데이터 개체를 작성한 후에는 소스를 개체에 추가합니다. 관계형 리소스를 소스로 사용할 수 있습니다.

관계형 리소스를 사용자 지정된 개체에 추가하려면 먼저 데이터베이스에 대한 연결을 구성해야 합니다.

1. **연결 탐색기** 보기에서 동일한 관계형 연결에 있는 하나 이상의 관계형 리소스를 선택합니다.
2. **연결 탐색기** 보기를 마우스 오른쪽 단추로 클릭하고 **프로젝트에 추가**를 선택합니다.
프로젝트에 추가 대화 상자가 나타납니다.
3. **사용자 지정된 기존 데이터 개체에 연결된 리소스로 추가**를 선택하고 **확인**을 클릭합니다.
데이터 개체에 추가 대화 상자가 나타납니다.
4. 사용자 지정된 데이터 개체를 선택하고 **확인**을 클릭합니다.
5. 여러 리소스를 사용자 지정된 데이터 개체에 추가하는 경우 **Developer** 도구가 데이터를 쓸 리소스를 선택 하라는 메시지를 표시합니다. 리소스를 선택하고 **확인**을 클릭합니다.
사용자 지정된 데이터 개체를 매핑에서 쓰기 변환으로 사용하는 경우 **Developer** 도구가 데이터를 이 리소스에 씁니다.

Developer 도구가 리소스를 사용자 지정된 데이터 개체에 추가합니다.

사용자 지정된 데이터 개체에 관계형 데이터 개체 추가

사용자 지정된 데이터 개체를 작성한 후에는 소스를 개체에 추가합니다. 관계형 데이터 개체를 소스로 사용할 수 있습니다.

1. 사용자 지정된 데이터 개체를 엽니다.
2. **읽기** 보기를 선택합니다.
3. **연결 탐색기** 보기에서 동일한 관계형 연결에 있는 하나 이상의 관계형 데이터 개체를 선택합니다.
4. 개체를 **개체 탐색기** 보기에서 사용자 지정된 데이터 개체의 **읽기** 보기로 끌어서 놓습니다.
5. 여러 관계형 데이터 개체를 사용자 지정된 데이터 개체에 추가하는 경우 **Developer** 도구가 데이터를 쓸 개체를 선택하라는 메시지를 표시합니다. 개체를 선택하고 **확인**을 클릭합니다.
사용자 지정된 데이터 개체를 매핑에서 쓰기 변환으로 사용하는 경우 **Developer** 도구가 데이터를 이 관계형 데이터 개체에 씁니다.

Developer 도구가 관계형 데이터 개체를 사용자 지정된 데이터 개체에 추가합니다.

사용자 지정된 데이터 개체에 키 작성

소스 변환에서 각 행을 식별하는 키 열을 작성합니다. 각 소스 변환에서 한 개의 기본 키를 작성할 수 있습니다.

1. 사용자 지정된 데이터 개체를 엽니다.
2. **읽기** 보기를 선택합니다.
3. 키를 작성하려는 소스 변환을 선택합니다.
소스는 관계형 데이터 개체가 아닌 관계형 리소스여야 합니다. 소스가 관계형 데이터 개체일 경우 관계형 데이터 개체에서 키를 작성해야 합니다.
4. **키 속성**을 선택합니다.
5. **추가**를 클릭합니다.
새 키 대화 상자가 나타납니다.

- 키 이름을 입력합니다.
- 키가 기본 키이면 **기본 키**를 선택합니다.
- 키 열을 선택합니다.
- 확인**을 클릭합니다.
- 사용자 지정된 데이터 개체를 저장합니다.

사용자 지정된 데이터 개체 내 관계 작성

사용자 지정된 데이터 개체의 소스 간에 키 관계를 작성할 수 있습니다.

참조하는 소스 변환에 기본 키가 있어야 합니다.

- 사용자 지정된 데이터 개체를 엽니다.
- 읽기** 보기를 선택합니다.
- 외래 키를 작성하려는 소스 변환을 선택합니다.

소스는 관계형 데이터 개체가 아닌 관계형 리소스여야 합니다. 소스가 관계형 데이터 개체일 경우 관계형 데이터 개체에서 관계를 작성해야 합니다.

- 관계** 속성을 선택합니다.
- 추가**를 클릭합니다.
새 **관계** 대화 상자가 나타납니다.
- 외래 키의 이름을 입력합니다.
- 참조된 소스 변환에서 기본 키를 선택합니다.
- 확인**을 클릭합니다.
- 관계** 속성에서 외래 키 열을 선택합니다.
- 사용자 지정된 데이터 개체를 저장합니다.

대상 테이블 작성 또는 대체

Developer tool을 사용하면 모델 리포지토리에 있는 하나 이상의 관계형 데이터 개체에 대해 **DDL** 스크립트를 생성하고, **DDL** 스크립트를 실행하여 대상 데이터베이스에서 테이블을 작성하거나 바꿀 수 있습니다. 대상이 이미 해당 데이터베이스에 존재할 경우 해당 대상을 삭제하고 다시 작성할 수 있습니다.

디자인 타임 또는 런타임 시 대상 테이블을 생성하거나 바꿀 수 있습니다. 디자인 타임에 테이블을 작성하거나 바꾸려면 매핑을 실행하기 전에 **DDL** 스크립트를 작성하고 실행해야 합니다. 매핑의 쓰기 변환에서 대상 스키마 전략을 구성하여 런타임 시 관계형 대상을 생성하거나 바꿀 수 있습니다.

Developer tool은 지원되는 모든 연결 유형에 대해 **DDL** 스크립트의 데이터베이스별 버전을 생성합니다. **JDBC** 또는 **ODBC** 대상을 선택하면 **Developer tool**이 **ANSI SQL-92** 일반 데이터 유형 형식을 생성합니다. 다음과 같은 데이터베이스 유형에 대해 **DDL** 스크립트를 생성할 수 있습니다.

- IBM DB2
- Greenplum
- Hive
- JDBC

- Microsoft SQL Server
- Netezza
- ODBC
- Oracle
- Teradata

참고: OLE DB SQL Server 연결을 사용하는 경우에는 DDL 스크립트를 생성할 수 없습니다.

대상 테이블 작성 또는 바꾸기에 대한 규칙 및 지침

DDL 스크립트를 생성하고 실행할 때 다음과 같은 규칙 및 지침을 고려하십시오.

- 소스 테이블 이름이 동일한 개체를 여러 개 선택하지 마십시오. 소스 테이블 이름이 같은 개체를 여러 개 선택하면 DDL 코드가 실패합니다. 소스 테이블 이름이 같은 개체 세 개에 대해 테이블 삭제 및 테이블 작성 옵션을 선택하면 DDL 코드는 첫 번째 테이블 삭제 및 테이블 작성 명령에 대해서는 실행되고 나머지 명령에 대해서는 실패합니다.
- 매핑을 실행하기 전에 데이터 유형을 확인하고, 필요한 경우 수동으로 업데이트하십시오. DDL 스크립트를 생성할 때 Oracle의 char 및 byte 의미는 무시됩니다. Oracle 데이터베이스의 char 및 varchar2 열이 포함된 테이블을 작성할 때는 char 및 byte 의미를 위한 저장소를 정의할 수 있습니다. Oracle 테이블의 메타데이터를 Developer tool로 가져오면 char 및 byte 의미가 무시됩니다. Oracle 테이블에 대해 DDL 스크립트를 생성하면 Developer tool이 데이터 유형을 문자로 정의합니다.
- ODBC 연결에 대해 DDL을 생성할 경우 Developer tool은 ANSI SQL-92 일반 데이터 유형 형식으로 DDL 스크립트를 작성합니다. ANSI SQL-92 형식은 대상 데이터베이스에서 해당 데이터 유형 또는 데이터 길이를 지원하지 않을 수 있으므로 일부 데이터베이스에서 실행되지 않을 수 있습니다.
- Greenplum에서 Netezza로 데이터베이스 테이블을 마이그레이션하는 DDL 스크립트를 작성 경우 NVARCHAR 데이터 유형은 Netezza 데이터베이스에 문자를 16000자까지만 지원하기 때문에 NVARCHAR 열에 문자를 16000자까지만 통합할 수 있습니다.
- DDL 스크립트를 생성하면 Developer tool은 원래 연결 및 새 연결이 가리키는 데이터베이스 사이에 가장 잘 일치하는 데이터 유형을 식별합니다. 데이터베이스마다 데이터 유형의 전체 자릿수 및 소수 자릿수가 다릅니다. Oracle 데이터베이스에서는 타임스탬프 데이터 유형의 기본 전체 자릿수와 소수 자릿수가 (29, 9)입니다. Oracle에서 Microsoft SQL Server로의 DDL 스크립트를 생성하면 타임스탬프 데이터 유형의 전체 자릿수와 소수 자릿수가 (26, 6)으로 줄어듭니다. Oracle에서 DB2로의 DDL 스크립트를 생성하면 타임스탬프 데이터 유형의 전체 자릿수와 소수 자릿수가 (27, 7)로 줄어듭니다.

디자인 타임에 DDL 생성 및 실행

DDL 스크립트를 생성하고 실행하려면 먼저 사용자가 대상 데이터베이스에 액세스하기 위한 적절한 읽기 및 쓰기 권한을 가지고 있는지 확인해야 합니다.

1. Developer tool의 **Object Explorer** 보기에서 대상 데이터베이스에 테이블을 작성하려는 관계형 데이터 개체를 선택합니다. 테이블을 여러 개 작성하려면 **Shift** 키와 **Ctrl** 키를 누른 채로 데이터 개체를 여러 개 선택합니다.
2. 선택한 데이터 개체를 마우스 오른쪽 단추로 클릭하고 **DDL 생성 및 실행**을 선택합니다.
DDL 생성 및 실행 대화 상자가 나타납니다.



3. 개체 선택 영역에서 DDL을 생성하려는 **대상 데이터베이스 유형**을 선택합니다.
4. DDL 생성 영역에서는 다음과 같은 옵션을 선택할 수 있습니다.
 - **테이블 삭제.** 테이블을 작성하기 전에 데이터베이스에서 테이블을 삭제합니다. 테이블을 바꾸려면 이 옵션을 선택합니다.
 - **테이블 작성.** 대상 데이터베이스에 테이블을 작성합니다.
 - **기본 키.** 선택한 데이터 개체에 기반하여 개인 키를 작성합니다.
 - **외래 키.** 선택한 데이터 개체에 기반하여 외래 키를 작성합니다.
 - **클립보드로 복사.** DDL 스크립트를 클립보드로 복사합니다.
 - **다른 이름으로 저장.** DDL 스크립트를 파일에 저장합니다.
5. DDL 생성 영역에서 **DDL 생성**을 클릭합니다.
 선택한 데이터 개체에 대해 생성된 DDL이 DLL 생성 영역에 나타납니다.
경고: DDL 생성 영역에서 생성된 DDL 스크립트를 편집하면 DDL 스크립트 실행 시 오류가 발생할 수 있습니다.
6. DDL 실행 영역에서 **찾아보기**를 클릭하고 대상 데이터베이스 연결을 선택합니다.
 개체 선택 영역에서 선택한 대상 데이터베이스 유형에 기반하여 **연결 선택** 대화 상자에 대상 데이터베이스 연결이 나열됩니다. 예를 들어 Oracle을 대상 데이터베이스 유형으로 선택했다면 Developer tool에 Oracle 연결이 표시됩니다.
7. **DDL 실행**을 클릭합니다.
8. **닫기**를 클릭합니다.

런타임 시 DDL 생성 및 실행

매핑의 쓰기 변환에서 런타임 시 관계형 또는 Hive 대상을 생성하거나 바꾸는 대상 스키마 전략을 선택하는 경우 DDL 쿼리를 정의할 수 있습니다.

데이터 통합 서비스는 정의된 DDL 쿼리에 따라 관계형 및 Hive 대상 테이블을 생성하거나 바꿉니다.

데이터 통합 서비스를 통해 생성하거나 바꿔야 하는 대상 테이블에 따라 DDL 쿼리를 정의하여 테이블을 사용자 지정하거나 파티션 같은 추가 매개 변수를 지정할 수 있습니다. 테이블에는 DDL 쿼리에서 정의하는 열이 포함됩니다. DDL 쿼리에 자리 표시자를 입력할 수 있습니다. 데이터 통합 서비스는 런타임 시 자리 표시자를 실제 값으로 대체합니다.

자세한 내용은 *Informatica Developer 매핑 가이드*를 참조하십시오.

DDL 생성 오류

DDL 스크립트를 생성하고 실행할 때 오류가 발생할 수 있습니다. 오류는 다음 중 한 가지 원인 때문에 발생할 수 있습니다.

- 선택한 데이터 유형을 대상 데이터베이스에서 지원하지 않는 경우

- 순환 종속성이 포함된 실제 데이터 개체를 선택한 경우
- 열이 없는 실제 데이터 개체를 선택한 경우

사용자 지정 쿼리

사용자 지정 SQL 쿼리는 사용자 지정된 데이터 개체에서 기본 SQL 쿼리를 재정의하는 SELECT 문입니다.

사용자 지정 쿼리는 데이터 통합 서비스가 관계형 소스에서 데이터를 읽는 데 사용하는 기본 SQL 쿼리를 재정의합니다. 또한 사용자 쿼리는 사용자가 소스 필터를 입력하거나, 정렬된 포트를 사용하거나, 사용자 정의 조인을 입력하거나, 고유 포트를 선택할 때 정의하는 단순 쿼리 설정도 재정의합니다.

사용자 쿼리를 작성하여 데이터베이스 언어에서는 유효하지만 변환 언어에서는 사용할 수 없는 SQL 작업을 수행할 수 있습니다. 사용자 쿼리를 사용자 지정된 데이터 개체에서 정의하면 여러 매핑 또는 프로필에서 개체를 재사용할 수 있습니다.

사용자 지정된 데이터 개체에서 사용자 지정 쿼리를 작성할 때 다음 지침을 따르십시오.

- SELECT 문에서 열 이름을 나열할 때 소스 변환에 나타나는 순서대로 나열합니다.
- 모든 데이터베이스 예약어를 따옴표로 묶습니다.
- 달러 기호(\$) 앞에 이스케이프 문자를 추가합니다. \$ 앞에 슬래시(\) 문자가 있을 경우 이스케이프 문자(\)를 슬래시 및 달러 기호 문자 모두에 추가합니다. 예를 들어 \$를 \\$로, \ \$를 \\\$로 입력합니다.

사용자 지정된 데이터 개체를 사용하여 셀프 조인을 수행하는 경우 셀프 조인을 포함하는 사용자 지정 SQL 쿼리를 입력해야 합니다. 사용자 지정 쿼리가 포함된 사용자 지정된 데이터 개체를 매핑에서 읽기 변환으로 사용할 수 있습니다. 소스 데이터베이스는 데이터를 데이터 통합 서비스에 전달하기 전에 쿼리를 실행합니다. 사용자 쿼리를 작성하여 소스를 빈 사용자 지정된 데이터 개체에 추가할 수 있습니다. 또한 사용자 지정 쿼리를 사용하여 기본 SQL 쿼리를 재정의할 수도 있습니다.

사용자 지정 쿼리 최적화

데이터 통합 서비스는 사용자 쿼리를 관계형 데이터 개체에서 실행하도록 푸시하여 성능을 향상시킬 수 있습니다. 쿼리가 데이터베이스에 대해 유효한 하위 쿼리를 구성하는 경우 사용자 지정 쿼리를 푸시하도록 선택합니다.

사용자 지정 쿼리를 사용하여 관계형 데이터 개체에서 데이터를 읽는 경우 데이터 통합 서비스가 쿼리를 데이터베이스에서 실행하도록 푸시하여 쿼리를 최적화할 수 있습니다. 쿼리가 데이터베이스에 대해 유효한 하위 쿼리를 구성하는 경우 데이터 통합 서비스가 사용자 지정 쿼리를 푸시할 수 있습니다. 사용자 지정 쿼리의 SQL 구문이 데이터베이스의 하위 쿼리에서 유효하지 않으면 결과 쿼리를 실행하지 못합니다.

사용자 지정 쿼리를 IBM DB2 이외의 관계형 데이터베이스에 푸시하는 경우 선택 목록에서 열 참조가 아닌 각 식의 별칭을 지정해야 합니다. 별칭을 통해 데이터 통합 서비스가 선택 목록의 식을 참조할 수 있습니다.

별칭 및 하위 쿼리의 유효한 SQL 구문에 대한 정보는 데이터베이스 설명서를 참조하십시오.

기본 쿼리

데이터 통합 서비스는 관계형 소스에서 데이터를 읽는 데 사용하는 기본 SQL 쿼리를 재정의합니다. 사용자 지정된 데이터 개체 또는 관계형 데이터 개체의 인스턴스에서 기본 쿼리를 재정의할 수 있습니다.

단순 또는 고급 쿼리를 통해 기본 쿼리를 재정의할 수 있습니다. 단순 쿼리를 사용하면 고유 값을 선택하거나, 소스 필터를 입력하거나, 포트를 정렬하거나 사용자 정의 조인을 입력할 수 있습니다. 고급 쿼리를 사용하면 소스에서 데이터를 읽기 위한 사용자 정의 SQL 쿼리를 작성할 수 있습니다. 사용자 지정 쿼리는 기본 및 단순 쿼리를 재정의합니다.

테이블 이름 또는 열 이름에 데이터베이스 예약어가 포함되는 경우 예약어 파일 **reswords.txt**를 작성하고 유지 관리할 수 있습니다. 데이터 통합 서비스가 액세스할 수 있는 시스템에 **reswords.txt** 파일을 작성합니다.

데이터 통합 서비스는 매핑을 실행할 때 **reswords.txt** 파일을 검색합니다. 파일이 존재하면 데이터 통합 서비스는 데이터베이스에 대해 **SQL**을 실행할 때 일치하는 예약어를 따옴표로 묶습니다. 기본 쿼리를 재정의하는 경우에는 데이터베이스 예약어를 따옴표로 묶어야 합니다.

데이터 통합 서비스는 기본 쿼리를 생성할 때 다음과 같은 문자가 포함된 테이블 및 필드 이름을 큰따옴표로 구분합니다.

/ + - = ~ ` ! % ^ & * () [] { } ' ; ? , < > \ | <space>

예약어 파일 작성

사용자 지정된 데이터 개체의 테이블 이름 또는 열 이름에 데이터베이스 예약어가 포함되는 경우 예약어 파일을 작성합니다.

예약어 파일을 사용하도록 데이터 통합 서비스를 구성하려면 관리자 권한이 있어야 합니다.

1. "reswords.txt"라는 파일을 작성합니다.
2. 데이터베이스 이름을 [Oracle]과 같이 대괄호 내에 입력하여 각 데이터베이스의 섹션을 작성합니다.
3. 파일에서 데이터베이스 이름 아래에 예약어를 추가합니다.

예:

```
[Oracle]
OPTION
START
where
number
[SQL Server]
CURRENT
where
number
```

입력은 대/소문자를 구분하지 않습니다.

4. reswords.txt 파일을 저장합니다.
5. Informatica Administrator에서 데이터 통합 서비스를 선택합니다.
6. 사용자 지정 속성을 편집합니다.
7. 다음과 같은 사용자 지정 속성을 추가합니다.

이름	값
예약어 파일	<path>\reswords.txt

8. 데이터 통합 서비스를 다시 시작합니다.

힌트

소스 **SQL** 쿼리에 힌트를 추가하여 데이터베이스 최적화 프로그램에 명령을 전달할 수 있습니다. 최적화 프로그램이 힌트를 사용하여 소스에 액세스할 쿼리 실행 계획을 선택합니다.

힌트 필드는 관계형 데이터 개체 인스턴스 또는 사용자 지정된 데이터 개체의 **쿼리** 보기에 나타납니다. 소스 데이터베이스는 **Oracle**, **Sybase**, **IBM DB2** 또는 **Microsoft SQL Server**여야 합니다. 다른 데이터베이스 유형의 경우 힌트 필드가 나타나지 않습니다.

데이터 통합 서비스는 소스 쿼리를 생성할 때 사용자가 **Developer** 도구에서 입력한 것과 동일한 **SQL** 힌트를 쿼리에 추가합니다. 데이터 통합 서비스는 힌트를 구문 분석하지 않습니다. 소스가 포함된 매핑을 실행하는 경우 매핑 로그에서 쿼리 및 쿼리의 힌트를 보여 줍니다.

데이터 통합 서비스는 데이터베이스 유형에 따라 쿼리의 특정 위치에 **SQL** 힌트를 삽입합니다. 힌트 구문에 대한 자세한 내용은 데이터베이스 설명서를 참조하십시오.

Oracle

데이터 통합 서비스가 **SELECT/UPDATE/INSERT/DELETE** 키워드 바로 뒤에 힌트를 추가합니다.

```
SELECT /*+ <힌트> */ FROM ...
```

'+'는 힌트의 시작을 나타냅니다.

힌트는 주석(/* ... */ 또는 --... 줄의 끝까지) 내에 포함됩니다.

Sybase

데이터 통합 서비스가 쿼리 뒤에 힌트를 추가합니다. 힌트에 계획 이름을 구성합니다.

```
SELECT ... PLAN <plan>
```

제목 계획 "(scalar_agg (i_scan type_price_ix titles)"에서 평균(가격) 선택

IBM DB2

optimize-for 절을 힌트로 입력할 수 있습니다. 데이터 통합 서비스가 이 절을 쿼리의 끝에 추가합니다.

```
SELECT ... OPTIMIZE FOR <n> ROWS
```

optimize-for 절은 쿼리에서 처리할 수 있는 행의 수를 데이터베이스 최적화 프로그램에 알려 줍니다. 이 절은 행의 수를 제한하지 않습니다. 데이터베이스가 <n>개가 넘는 행을 처리하는 경우 성능이 저하될 수 있습니다.

Microsoft SQL Server

데이터 통합 서비스가 쿼리의 끝에 **OPTION** 절의 일부로 힌트를 추가합니다.

```
SELECT ... OPTION ( <query_hints> )
```

힌트 규칙 및 지침

SQL 쿼리의 힌트를 구성할 때 다음 규칙 및 지침을 따르십시오.

- 푸시다운 최적화를 활성화하거나 관계형 데이터 개체에서 반 조인을 사용하는 경우 원래 소스 쿼리가 변경됩니다. 데이터 통합 서비스는 수정된 쿼리에 힌트를 적용하지 않습니다.
- 힌트를 조인 및 필드 재정의와 결합할 수 있지만 **SQL** 재정의의 구성하는 경우 **SQL** 재정의가 우선하고 데이터 통합 서비스가 다른 재정의의 적용하지 않습니다.
- **쿼리** 보기에는 샘플 보기 또는 고급 보기가 표시됩니다. 샘플 보기에서 힌트를 필터, 정렬 또는 조인 재정의와 함께 입력하는 경우 **Developer** 도구가 고급 보기에서 전체 쿼리 재정의의 보여 줍니다.

힌트 작성

쿼리 계획을 결정하도록 데이터베이스 최적화 프로그램에 지침을 보내는 쿼리를 작성할 수 있습니다.

1. 사용자 지정된 데이터 개체 또는 관계형 데이터 개체 인스턴스를 엽니다.
2. **읽기** 보기를 선택합니다.
3. **출력 변환**을 선택합니다.
4. **쿼리 속성**을 선택합니다.

5. 단순 쿼리를 선택합니다.
6. **힌트** 필드 옆에서 **편집**을 클릭합니다.
힌트 대화 상자가 나타납니다.
7. **SQL 쿼리** 필드에 힌트를 입력합니다.
Developer 도구는 힌트의 유효성을 검사하지 않습니다.
8. **확인**을 클릭합니다.
9. 데이터 개체를 저장합니다.

고유 항목 선택

고유 항목 선택 옵션을 사용하여 사용자 지정된 데이터 개체 또는 관계형 데이터 개체 인스턴스의 소스에서 고유한 값을 선택할 수 있습니다. 고유 항목 선택을 활성화하면 데이터 통합 서비스가 **SELECT DISTINCT** 문을 기본 SQL 쿼리에 추가합니다.

고유 항목 선택 옵션을 사용하여 소스 데이터를 필터링할 수 있습니다. 예를 들어 고유 항목 선택 옵션을 사용하여 총 매출을 나열하는 테이블에서 고유한 고객 ID를 추출할 수 있습니다. 매핑에서 관계형 데이터 개체를 사용하는 경우 데이터 통합 서비스가 데이터 흐름의 초기에 데이터를 필터링하므로 성능이 향상될 수 있습니다.

고유 항목 선택 사용

고유 항목 선택 속성을 사용하여 관계형 소스에서 고유한 값을 선택합니다.

1. 사용자 지정된 데이터 개체 또는 관계형 데이터 개체 인스턴스를 엽니다.
2. **읽기** 보기를 선택합니다.
3. **출력 변환**을 선택합니다.
4. **쿼리** 속성을 선택합니다.
5. 단순 쿼리를 선택합니다.
6. **고유 항목 선택** 옵션을 활성화합니다.
7. 사용자 지정된 데이터 개체를 저장합니다.

필터

사용자 지정 쿼리에 필터 값을 입력할 수 있습니다. 필터는 쿼리 **SELECT** 문에서 **WHERE** 절이 됩니다. 필터를 사용하여 데이터 통합 서비스가 소스 테이블에서 읽는 행의 수를 줄입니다.

소스 필터 입력

소스 필터를 입력하여 데이터 통합 서비스가 관계형 소스에서 읽는 행의 수를 줄입니다.

1. 사용자 지정된 데이터 개체 또는 관계형 데이터 개체 인스턴스를 엽니다.
2. **읽기** 보기를 선택합니다.
3. **출력 변환**을 선택합니다.
4. **쿼리** 속성을 선택합니다.
5. 단순 쿼리를 선택합니다.
6. **필터** 필드 옆에서 **편집**을 클릭합니다.
SQL 쿼리 대화 상자가 나타납니다.

7. **SQL 쿼리** 필드에 필터 조건을 입력합니다.
열 목록에서 열 이름을 선택할 수 있습니다.
8. **확인**을 클릭합니다.
9. **유효성 검사**를 클릭하여 필터 조건의 유효성을 검사합니다.
10. 데이터 개체를 저장합니다.

정렬된 포트

사용자 지정된 데이터 개체 또는 관계형 데이터 개체 인스턴스에 대한 기본 쿼리에서 행을 정렬할 수 있습니다. 정렬 기준으로 사용할 포트를 선택합니다. 데이터 통합 서비스가 포트를 기본 쿼리의 **ORDER BY** 절에 추가합니다.

다음 변환을 매핑에 포함하는 경우 소스 행을 정렬하여 성능을 향상시킬 수 있습니다.

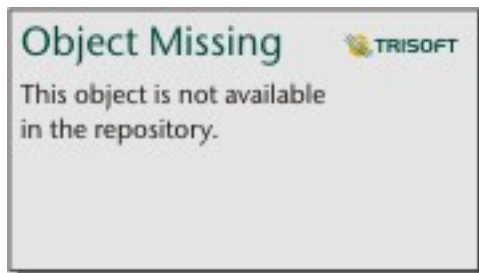
- 집계. 정렬된 입력에 대해 집계 변환을 구성하는 경우 정렬된 포트를 사용하여 정렬된 데이터를 보낼 수 있습니다. 집계 변환의 그룹 기준 포트가 사용자 지정된 데이터 개체에서 정렬된 포트의 순서와 일치해야 합니다.
- 조이너. 정렬된 입력에 대해 조이너 변환을 구성하는 경우 정렬된 포트를 사용하여 정렬된 데이터를 보낼 수 있습니다. 각각의 사용자 지정된 데이터 개체에서 정렬된 포트의 순서를 동일하게 구성합니다.

참고: 또한 분류기 변환을 사용하여 집계 및 조이너 변환 전에 관계형 및 플랫폼 파일 데이터를 정렬할 수 있습니다.

열 데이터 정렬

정렬된 포트를 사용하여 사용자 지정된 데이터 개체 또는 관계형 데이터 개체 인스턴스에서 열 데이터를 정렬합니다. 매핑 또는 맵셋에서 읽기 변환으로 데이터 개체를 사용하는 경우 읽기 변환의 변환 다운스트림으로 정렬된 데이터를 전달할 수 있습니다.

1. 사용자 지정된 데이터 개체 또는 관계형 데이터 개체 인스턴스를 엽니다.
2. **읽기** 보기를 선택합니다.
다음 그림은 편집기에서 열린 사용자 지정된 데이터 개체의 **읽기** 보기를 보여 줍니다.



3. 출력 변환을 선택합니다.
4. **쿼리** 속성을 선택합니다.
5. 단순 쿼리를 선택합니다.
6. **정렬** 필드 옆의 **편집**을 클릭합니다.
정렬 대화 상자가 표시됩니다.
7. 열을 정렬된 포트로 지정하려면 **새로 만들기** 단추를 클릭합니다.
8. 열 및 정렬 유형(오름차순 또는 내림차순)을 선택합니다.
9. **7** 및 **8** 단계를 반복하여 정렬할 다른 열을 선택합니다.

Developer 도구는 **정렬** 대화 상자에 나타나는 순서로 열을 정렬합니다.

10. **확인**을 클릭합니다.

쿼리 속성에서 Developer 도구가 **정렬** 필드에 정렬 열을 표시합니다.

11. **유효성 검사**를 클릭하여 정렬 구문의 유효성을 검사합니다.
12. 데이터 개체를 저장합니다.

사용자 정의 조인

사용자 지정된 데이터 개체 또는 관계형 데이터 개체 인스턴스에서 사용자 정의 조인을 구성할 수 있습니다. 사용자 정의 조인은 여러 소스의 데이터를 동일한 데이터 개체에 조인하는 조건을 정의합니다.

사용자 정의 조인을 사용자 지정된 데이터 개체 또는 관계형 데이터 개체 인스턴스에 추가하면 매핑에서 읽기 변환으로 데이터 개체를 사용할 수 있습니다. 소스 데이터베이스는 데이터를 데이터 통합 서비스에 전달하기 전에 조인을 수행합니다. 소스 테이블이 인덱싱되면 매핑 성능이 향상됩니다.

사용자 정의 조인을 작성하여 관련 소스의 데이터를 조인합니다. 사용자 정의 조인은 데이터 통합 서비스가 각 소스의 관련된 키에 따라 작성하는 기본 내부 조인을 재정의합니다. 사용자 정의 조인을 입력할 때 조인 조건을 지정하는 **WHERE** 절의 내용을 입력합니다. 사용자 정의 조인이 외부 조인을 수행하는 경우 데이터 통합 서비스가 데이터베이스 구문에 따라 조인 구문을 **WHERE** 절 또는 **FROM** 절에 삽입할 수 있습니다.

다음과 같은 경우 사용자 정의 조인을 입력해야 할 수 있습니다.

- 열에 기본 키-외래 키 관계가 없습니다.
- 조인에 사용되는 각 열의 데이터 유형이 일치하지 않습니다.
- 다른 유형의 조인(예: 외부 조인)을 지정하려고 합니다.

사용자 지정된 데이터 개체 또는 관계형 데이터 개체 인스턴스에서 사용자 정의 조인을 입력할 때 다음 지침을 따르십시오.

- 사용자 정의 조인에 **WHERE** 키워드를 포함하지 않습니다.
- 모든 데이터베이스 예약어를 따옴표로 묶습니다.
- Informatica 조인 구문을 사용하고 연결에 대해 **SQL에서 따옴표 활성화**가 설정되면 테이블 이름 및 열 이름을 수동으로 입력하는 경우 해당 이름을 따옴표로 묶어야 합니다. 사용자 정의 조인을 입력할 때 테이블 및 열을 선택하면 Developer 도구가 테이블 이름 및 열 이름을 따옴표로 묶습니다.

사용자 정의 조인은 데이터베이스에 있는 관련 리소스의 데이터를 조인합니다. 다른 유형의 소스를 조인하려면 소스로부터 데이터를 읽는 매핑에서 조이너 변환을 사용합니다. 셀프 조인을 수행하려면 셀프 조인이 포함된 사용자 지정 **SQL** 쿼리를 입력해야 합니다.

사용자 정의 조인 입력

사용자 지정된 데이터 개체 또는 관계형 데이터 개체에서 사용자 정의 조인을 구성하여 데이터 개체 소스에 대한 조인 조건을 정의합니다.

1. 사용자 지정된 데이터 개체 또는 관계형 데이터 개체 인스턴스를 엽니다.
2. **읽기** 보기를 선택합니다.
3. **출력 변환**을 선택합니다.
4. **쿼리 속성**을 선택합니다.
5. **단순 쿼리**를 선택합니다.
6. **조인 필드** 옆에서 **편집**을 클릭합니다.

SQL 쿼리 대화 상자가 나타납니다.

7. **SQL 쿼리** 필드에 사용자 정의 조인을 입력합니다.
열 목록에서 열 이름을 선택할 수 있습니다.
8. **확인**을 클릭합니다.
9. **유효성 검사**를 클릭하여 사용자 정의 조인의 유효성을 검사합니다.
10. 데이터 개체를 저장합니다.

외부 조인 지원

사용자 지정된 데이터 개체를 사용하여 동일한 데이터베이스에 있는 두 소스의 외부 조인을 수행할 수 있습니다. 데이터 통합 서비스는 외부 조인을 수행하는 경우 조인 조건과 일치하는 한 소스 리소스의 모든 행과 두 번째 소스 리소스의 행을 반환합니다.

두 개의 리소스를 조인하고 이러한 리소스 중 하나에서 모든 행을 반환하려는 경우 외부 조인을 사용합니다. 예를 들어 등록된 고객의 테이블을 월별 구매 테이블과 조인하여 등록된 고객의 활동을 확인하려는 경우 외부 조인을 수행할 수 있습니다. 등록된 고객의 테이블을 월별 구매 테이블과 조인하고 지난 달 구매를 하지 않은 고객을 포함하여 등록된 고객 테이블의 모든 행을 반환할 수 있습니다. 일반 조인을 수행하는 경우 데이터 통합 서비스가 해당 월에 구매를 한 등록된 고객과 등록된 고객의 구매만 반환합니다.

외부 조인을 사용하는 경우 조이너 변환에서 마스터 외부 조인 또는 상세 외부 조인과 동일한 결과를 생성할 수 있습니다. 하지만 외부 조인을 사용하면 데이터 흐름의 행 수를 줄여 성능을 향상시킬 수 있습니다.

두 가지 유형의 외부 조인을 입력할 수 있습니다.

- 왼쪽. 데이터 통합 서비스가 조인 구문의 왼쪽에 있는 리소스에 대한 모든 열과 조인 조건을 만족하는 두 리소스 모두의 행을 반환합니다.
- 오른쪽. 데이터 통합 서비스가 조인 구문의 오른쪽에 있는 리소스에 대한 모든 열과 조인 조건을 만족하는 두 리소스 모두의 행을 반환합니다.

참고: 기본 쿼리를 재정의하는 경우 중첩된 쿼리 문에서 외부 조인을 사용합니다.

사용자 정의 조인 또는 SQL 쿼리에서 외부 조인을 입력할 수 있습니다.

Informatica 조인 구문

조인 구문을 입력할 때 Informatica 또는 데이터베이스별 조인 구문을 사용합니다. Informatica 조인 구문을 사용하면 데이터 통합 서비스가 매핑 실행 시 구문을 변환하여 소스 데이터베이스에 전달합니다.

참고: 조인 조건에 대해 데이터베이스별 구문을 항상 사용합니다.

Informatica 조인 구문을 사용하는 경우 전체 조인 문을 괄호({Informatica syntax})로 둘러쌉니다. 데이터베이스 구문을 사용하는 경우 소스 데이터베이스에서 지원되는 구문을 괄호 없이 입력합니다.

Informatica 조인 구문을 사용하는 경우 테이블 이름을 접두사로 열 이름에 추가합니다. 예를 들어 REG_CUSTOMER 테이블에 FIRST_NAME이라는 이름의 열이 있을 경우 조인 구문에 "REG_CUSTOMER.FIRST_NAME"을 입력합니다. 또한 테이블 이름에 별칭을 사용하는 경우 Informatica 조인 구문 내에서 별칭을 사용하여 데이터 통합 서비스에서 별칭을 인식하도록 합니다.

단일 데이터 개체에서 왼쪽 외부 조인 또는 오른쪽 외부 조인을 일반 조인과 결합할 수 있습니다. 하지만 왼쪽 외부 조인과 오른쪽 외부 조인을 결합할 수는 없습니다. 여러 일반 조인과 여러 왼쪽 외부 조인을 사용합니다. 일부 데이터베이스는 한 개의 오른쪽 외부 조인을 사용하도록 제한합니다.

조인을 결합할 때는 일반 조인을 먼저 입력합니다.

일반 조인 구문

사용자 지정된 데이터 개체 또는 관계형 데이터 개체 인스턴스에서 조인 조건을 사용하여 일반 조인을 작성할 수 있습니다.

외부 조인을 작성하는 경우 기본 조인을 재정의해야 합니다. 따라서 일반 조인을 조인 재정의에 포함해야 합니다. 일반 조인을 조인 재정의에 포함하는 경우 외부 조인 앞에 일반 조인을 나열합니다. 조인 재정의에서 여러 일반 조인을 입력할 수 있습니다.

일반 조인을 작성하려면 다음과 같은 구문을 사용합니다.

```
{ source1 INNER JOIN source2 on join_condition }
```

다음 표에는 조인 재정의의 일반 조인 구문이 나와 있습니다.

구문	설명
<i>source1</i>	소스 리소스 이름입니다. 데이터 통합 서비스가 이 리소스에서 조인 조건과 일치하는 행을 반환합니다.
<i>source2</i>	소스 리소스 이름입니다. 데이터 통합 서비스가 이 리소스에서 조인 조건과 일치하는 행을 반환합니다.
<i>join_condition</i>	조인 조건입니다. 소스 데이터베이스에서 지원되는 구문을 사용합니다. AND 연산자를 사용하여 여러 조인 조건을 결합할 수 있습니다.

예를 들어 등록된 고객에 대한 데이터가 포함된 REG_CUSTOMER 테이블이 있다고 가정합니다.

CUST_ID	FIRST_NAME	LAST_NAME
00001	Marvin	Chi
00002	Dinah	Jones
00003	John	Bowden
00004	J.	Marks

PURCHASES 테이블은 매달 업데이트되고 다음 데이터를 포함합니다.

TRANSACTION_NO	CUST_ID	DATE	AMOUNT
06-2000-0001	00002	6/3/2000	55.79
06-2000-0002	00002	6/10/2000	104.45
06-2000-0003	00001	6/10/2000	255.56
06-2000-0004	00004	6/15/2000	534.95
06-2000-0005	00002	6/21/2000	98.65
06-2000-0006	NULL	6/23/2000	155.65
06-2000-0007	NULL	6/24/2000	325.45

6월에 발생한 각 트랜잭션의 고객 이름을 표시하는 행을 반환하려면 다음과 같은 구문을 사용합니다.

```
{ REG_CUSTOMER INNER JOIN PURCHASES on REG_CUSTOMER.CUST_ID = PURCHASES.CUST_ID }
```

데이터 통합 서비스가 다음과 같은 데이터를 반환합니다.

CUST_ID	DATE	AMOUNT	FIRST_NAME	LAST_NAME
00002	6/3/2000	55.79	Dinah	Jones
00002	6/10/2000	104.45	Dinah	Jones
00001	6/10/2000	255.56	Marvin	Chi
00004	6/15/2000	534.95	J.	Marks
00002	6/21/2000	98.65	Dinah	Jones

데이터 통합 서비스가 고객 ID가 일치하는 행을 반환합니다. 6월에 구매하지 않은 고객은 포함하지 않습니다. 또한 등록되지 않은 고객의 구매도 포함하지 않습니다.

왼쪽 외부 조인 구문

조인 재정의의 사용하여 왼쪽 외부 조인을 작성할 수 있습니다. 단일 조인 재정의에서 여러 개의 왼쪽 외부 조인을 입력할 수 있습니다. 왼쪽 외부 조인을 다른 조인과 함께 사용하는 경우 문에서 일반 조인 뒤에 모든 왼쪽 외부 조인을 함께 나열합니다.

왼쪽 외부 조인을 작성하려면 다음과 같은 구문을 사용합니다.

```
{ source1 LEFT OUTER JOIN source2 on join_condition }
```

다음 표에는 조인 재정의의 왼쪽 외부 조인 구문이 나와 있습니다.

구문	설명
<i>source1</i>	소스 리소스 이름입니다. 왼쪽 외부 조인을 사용하면 데이터 통합 서비스가 이 리소스의 모든 행을 반환합니다.
<i>source2</i>	소스 리소스 이름입니다. 데이터 통합 서비스가 이 리소스에서 조인 조건과 일치하는 행을 반환합니다.
<i>join_condition</i>	조인 조건입니다. 소스 데이터베이스에서 지원되는 구문을 사용합니다. AND 연산자를 사용하여 여러 조인 조건을 결합할 수 있습니다.

예를 들어 [“일반 조인 구문” 페이지 75](#)에 설명된 REG_CUSTOMER 및 PURCHASES 테이블을 사용하여 다음 조인 재정의로 6월에 무언가 구입한 고객의 수를 확인할 수 있습니다.

```
{ REG_CUSTOMER LEFT OUTER JOIN PURCHASES on REG_CUSTOMER.CUST_ID = PURCHASES.CUST_ID }
```

데이터 통합 서비스가 다음과 같은 데이터를 반환합니다.

CUST_ID	FIRST_NAME	LAST_NAME	DATE	AMOUNT
00001	Marvin	Chi	6/10/2000	255.56
00002	Dinah	Jones	6/3/2000	55.79
00003	John	Bowden	NULL	NULL
00004	J.	Marks	6/15/2000	534.95
00002	Dinah	Jones	6/10/2000	104.45

CUST_ID	FIRST_NAME	LAST_NAME	DATE	AMOUNT
00002	Dinah	Jones	6/21/2000	98.65

데이터 통합 서비스가 6월에 구매하지 않은 고객에 대해 null 값을 사용하여 REG_CUSTOMERS 테이블의 모든 등록된 고객을 반환합니다. 등록되지 않은 고객의 구입은 포함하지 않습니다.

여러 개의 조인 조건을 사용하여 6월의 단일 구매에서 100달러가 넘게 소비한 등록된 고객의 수를 확인할 수 있습니다.

```
{REG_CUSTOMER LEFT OUTER JOIN PURCHASES on (REG_CUSTOMER.CUST_ID = PURCHASES.CUST_ID AND PURCHASES.AMOUNT > 100.00) }
```

데이터 통합 서비스가 다음과 같은 데이터를 반환합니다.

CUST_ID	FIRST_NAME	LAST_NAME	DATE	AMOUNT
00001	Marvin	Chi	6/10/2000	255.56
00002	Dinah	Jones	6/10/2000	104.45
00003	John	Bowden	NULL	NULL
00004	J.	Marks	6/15/2000	534.95

동일한 기간 동안의 반품에 대한 정보를 통합하려면 여러 개의 왼쪽 외부 조인을 사용할 수 있습니다. 예를 들어 RETURNS 테이블에 다음과 같은 데이터가 있습니다.

CUST_ID	CUST_ID	RETURN
00002	6/10/2000	55.79
00002	6/21/2000	104.45

6월에 구매하고 반품한 고객의 수를 확인하려면 두 개의 왼쪽 외부 조인을 사용합니다.

```
{ REG_CUSTOMER LEFT OUTER JOIN PURCHASES on REG_CUSTOMER.CUST_ID = PURCHASES.CUST_ID LEFT OUTER JOIN RETURNS on REG_CUSTOMER.CUST_ID = PURCHASES.CUST_ID }
```

데이터 통합 서비스가 다음과 같은 데이터를 반환합니다.

CUST_ID	FIRST_NAME	LAST_NAME	DATE	AMOUNT	RET_DATE	RETURN
00001	Marvin	Chi	6/10/2000	255.56	NULL	NULL
00002	Dinah	Jones	6/3/2000	55.79	NULL	NULL
00003	John	Bowden	NULL	NULL	NULL	NULL
00004	J.	Marks	6/15/2000	534.95	NULL	NULL
00002	Dinah	Jones	6/10/2000	104.45	NULL	NULL
00002	Dinah	Jones	6/21/2000	98.65	NULL	NULL
00002	Dinah	Jones	NULL	NULL	6/10/2000	55.79
00002	Dinah	Jones	NULL	NULL	6/21/2000	104.45

데이터 통합 서비스는 누락된 값에 대해 NULL을 사용합니다.

오른쪽 외부 조인 구문

조인 재정의의 사용하여 오른쪽 외부 조인을 작성할 수 있습니다. 조인 구문에서 리소스의 순서를 반대로 지정하면 오른쪽 외부 조인이 왼쪽 외부 조인과 동일한 결과를 반환합니다. 조인 재정의에서 한 개의 오른쪽 외부 조인만 사용합니다. 오른쪽 외부 조인을 두 개 이상 작성하려는 경우 소스 리소스의 순서를 반대로 지정하고 조인 유형을 왼쪽 외부 조인으로 변경해 봅니다.

오른쪽 외부 조인을 다른 조인과 함께 사용하려면 조인 재정의의 끝에 오른쪽 외부 조인을 입력합니다.

오른쪽 외부 조인을 작성하려면 다음과 같은 구문을 사용합니다.

```
{ source1 RIGHT OUTER JOIN source2 on join_condition }
```

다음 표에는 조인 재정의의 오른쪽 외부 조인 구문이 나와 있습니다.

구문	설명
<i>source1</i>	소스 리소스 이름입니다. 데이터 통합 서비스가 이 리소스에서 조인 조건과 일치하는 행을 반환합니다.
<i>source2</i>	소스 리소스 이름입니다. 오른쪽 외부 조인을 사용하면 데이터 통합 서비스가 이 리소스의 모든 행을 반환합니다.
<i>join_condition</i>	조인 조건입니다. 소스 데이터베이스에서 지원되는 구문을 사용합니다. AND 연산자를 사용하여 여러 조인 조건을 결합할 수 있습니다.

PreSQL 및 PostSQL 명령

사용자 지정된 데이터 개체 또는 관계형 데이터 개체 인스턴스에서 SQL 명령을 작성할 수 있습니다. SQL 명령을 실행하여 insert, update 및 delete 같은 SQL 문을 실행할 수 있습니다. 데이터 통합 서비스가 소스 관계형 리소스에 대해 SQL 명령을 실행합니다.

매핑을 실행하면 데이터 통합 서비스는 소스를 읽기 전에 소스 데이터베이스에 대해 PreSQL 명령을 실행합니다. 매핑에서 소스를 사용하기 전에 소스를 업데이트해야 하는 경우 PreSQL 명령을 사용할 수 있습니다. 예를 들어 소스의 레코드를 삭제하고 소스를 업데이트하여 소스 데이터베이스의 최신 레코드를 로드하는 PreSQL 명령을 구성할 수 있습니다. 매핑을 실행하면 매핑에 사용되는 소스에 최신 레코드가 포함됩니다.

데이터 통합 서비스는 대상에 쓴 후 소스 데이터베이스에 대해 PostSQL 명령을 실행할 수도 있습니다. 예를 들어 레코드를 매핑 대상에 로드한 후 소스의 레코드를 삭제하려는 경우 PostSQL 명령을 사용할 수 있습니다. 매핑이 대상에 쓴 후 레코드를 삭제하도록 PostSQL 명령을 구성할 수 있습니다.

PreSQL 및 PostSQL 명령을 구성할 때는 다음 지침을 사용하십시오.

- 데이터베이스 유형에 유효한 명령을 사용합니다. 데이터 통합 서비스는 데이터베이스가 중첩 명령을 허용해도 이러한 명령을 허용하지 않습니다.
- 세미콜론을 사용하여 여러 개의 문을 구분합니다. 데이터 통합 서비스는 각 문 후에 커밋을 실행합니다.
- 데이터 통합 서비스는 /* ... */ 내에 포함된 세미콜론은 무시합니다.
- 주석 외부에서 세미콜론을 사용해야 하는 경우 역슬래시(\)를 사용하여 이스케이프할 수 있습니다. 세미콜론을 이스케이프하면 데이터 통합 서비스가 역슬래시를 무시하고 세미콜론을 문 구분 기호로 사용하지 않습니다.
- Developer tool은 PreSQL 및 PostSQL 명령에 포함된 SQL의 유효성을 검사하지 않습니다.

사전 및 사후 매핑 SQL 명령 추가

사용자 지정된 데이터 개체 또는 관계형 데이터 개체 인스턴스에서 사전 및 사후 매핑 SQL 명령을 추가할 수 있습니다. 매핑에서 데이터 개체를 사용하면 데이터 통합 서비스가 SQL 명령을 실행합니다.

1. 사용자 지정된 데이터 개체를 엽니다.
2. **읽기** 보기를 선택합니다.
3. 출력 변환을 선택합니다.
4. **고급** 속성을 선택합니다.
5. **PreSQL** 필드에 사전 매핑 SQL 명령을 입력합니다.
6. **PostSQL** 필드에 사후 매핑 SQL 명령을 입력합니다.
7. 사용자 지정된 데이터 개체를 저장합니다.

사용자 지정 쿼리 작성

사용자 지정 쿼리를 작성하여 소스에서 데이터를 읽기 위한 특수 **SELECT** 문을 실행할 수 있습니다. 사용자 지정 쿼리는 데이터 통합 서비스가 소스 데이터를 읽기 위해 실행하는 기본 쿼리를 재정의합니다.

1. 사용자 지정된 데이터 개체 또는 관계형 데이터 개체 인스턴스를 엽니다.
2. **읽기** 보기를 선택합니다.
3. 출력 변환을 선택합니다.
4. **쿼리** 속성을 선택합니다.
5. 고급 쿼리를 선택합니다.
6. **사용자 지정 쿼리 사용**을 선택합니다.
데이터 통합 서비스가 소스 데이터를 읽기 위해 실행하는 쿼리를 표시합니다.
7. 쿼리를 변경하거나 사용자 지정 쿼리로 대체합니다.
8. 사용자 지정 쿼리를 관계형 데이터 소스에 푸시하려면 **사용자 지정 쿼리를 데이터베이스로 푸시**를 선택합니다.
데이터 통합 서비스는 기본적으로 사용자 지정 쿼리를 데이터베이스로 푸시하지 않습니다. 쿼리가 데이터베이스에 대해 유효한 하위 쿼리를 구성하는 경우 사용자 지정 쿼리를 푸시하도록 선택합니다.
9. 데이터 개체를 저장합니다.

비관계형 데이터 개체

비관계형 데이터 개체를 가져와서 매핑, 맵셋 또는 프로파일에서 사용할 수 있습니다. 비관계형 데이터 개체는 비관계형 데이터 소스를 사용하는 실제 데이터 개체입니다.

다음과 같은 연결 유형에 대해 비관계형 데이터 개체를 가져올 수 있습니다.

- Adabas
- IMS
- 순차
- VSAM

비관계형 데이터 개체를 가져오면 **Developer** 도구가 **PowerExchange** 데이터 맵에서 개체의 메타데이터를 읽습니다. 데이터 맵은 제품에서 **SQL** 언어를 사용하여 데이터에 액세스할 수 있도록 비관계형 레코드를 관계형 테이블과 연결합니다. 데이터 맵을 작성하려면 **PowerExchange Navigator**를 사용합니다.

개체를 가져온 후 개체의 비관계형 작업을 읽기, 쓰기 또는 조회 변환으로 매핑 및 맵렛에 포함할 수 있습니다. 각 비관계형 작업은 데이터 맵이 정의하는 관계형 테이블에 해당합니다. 관계형 테이블의 열에 대해 하나 이상의 비관계형 레코드에 있는 필드의 매핑을 확인하려면 **개체 탐색기** 보기에서 비관계형 작업을 두 번 클릭합니다.

데이터 맵에 대한 자세한 내용은 *PowerExchange Navigator 가이드*를 참조하십시오.

참고:

- **Informatica 9.0.1**에서 작성한 비관계형 데이터 개체로 작업하려면 먼저 개체를 업그레이드해야 합니다. 비관계형 데이터 개체를 업그레이드하려면 **infacmd pwx UpgradeModels** 명령을 실행합니다.
- 업데이트 전략 변환을 포함하는 매핑을 실행하고 비관계형 대상에 데이터를 쓰는 경우 오류 메시지와 함께 매핑이 실패할 수 있습니다. 메시지는 대상에 대한 데이터 맵에 키가 정의되어 있지만 대상 테이블에 키가 없다 내용이 표시됩니다.

비관계형 데이터 개체 가져오기

매핑, 맵렛 또는 프로필에서 사용할 하나 이상의 비관계형 데이터 개체를 가져옵니다.

비관계형 데이터 개체를 가져오려면 먼저 데이터베이스 또는 데이터 집합에 대한 연결을 구성해야 합니다. 또한 개체에 대한 데이터 맵도 작성해야 합니다.

1. **Object Explorer** 보기에서 프로젝트 또는 폴더를 선택합니다.
2. **파일 > 새로 만들기 > 데이터 개체**를 클릭하십시오.
3. **비관계형 데이터 개체**를 선택하고 **다음**을 클릭합니다.
새 비관계형 데이터 개체 대화 상자가 나타납니다.
4. 필요에 따라 실제 데이터 개체의 이름을 입력합니다.
5. **연결** 옵션 옆의 찾아보기를 클릭하고 연결을 선택합니다.
6. 데이터 맵 옵션 옆에 있는 **찾아보기**를 클릭합니다.
데이터 맵 선택 대화 상자가 나타납니다.
7. 필요에 따라 필터 조건을 입력합니다.
8. **리소스** 영역에서 필요에 따라 연결 및 스키마를 확장하고 원하는 데이터 맵을 선택합니다.
9. **확인**을 클릭하여 **데이터 맵 선택** 대화 상자를 닫고 **마침**을 클릭합니다.

Object Explorer 보기에서 프로젝트 또는 폴더의 **실제 데이터 개체** 아래에 있는 연결 이름에 비관계형 데이터 개체와 해당하는 비관계형 작업이 나타납니다.

참고: 또한 **연결 탐색기** 보기를 사용하여 비관계형 데이터 개체를 가져올 수 있습니다.

관계형 데이터 작업에서 읽기, 쓰기 또는 조회 변환 작성

비관계형 데이터 작업을 읽기, 쓰기 또는 조회 변환으로 매핑 또는 맵렛에 추가할 수 있습니다.

1. 읽기, 쓰기 또는 조회 변환을 작성하려는 매핑 또는 맵렛을 엽니다.
2. **개체 탐색기** 보기에서 하나 이상의 비관계형 데이터 작업을 선택합니다.
3. 비관계형 데이터 작업을 매핑 편집기로 끌어서 놓습니다.
매핑에 추가 대화 상자가 나타납니다.
4. **읽기, 쓰기 또는 조회** 옵션을 선택합니다.

독립 데이터 개체로 자동으로 선택됩니다.

5. **확인**을 클릭합니다.

Developer 도구가 각 비관계형 데이터 작업에 대해 읽기, 쓰기 또는 조회 변환을 작성합니다.

WSDL 데이터 개체

WSDL 데이터 개체는 WSDL 파일을 소스로 사용하는 실제 데이터 개체입니다. WSDL 데이터 개체를 사용하여 웹 서비스 또는 웹 서비스 소비자 변환을 작성할 수 있습니다. WSDL 파일을 가져와서 WSDL 데이터 개체를 작성합니다.

WSDL 데이터 개체를 가져온 후 **개요** 및 **고급** 보기에서 일반 및 고급 속성을 편집할 수 있습니다. **WSDL** 보기에 WSDL 파일 콘텐츠가 표시됩니다.

WSDL을 가져올 때 다음 지침을 고려하십시오.

- WSDL 파일은 WSDL 1.1과 호환 가능해야 합니다.
- WSDL 파일은 유효해야 합니다.
- 웹 서비스 또는 웹 서비스 소비자 변환에 포함할 작업에서 문서/리터럴 인코딩을 사용해야 합니다. WSDL 파일의 모든 작업에서 문서/리터럴 이외의 인코딩 유형을 사용할 경우 WSDL 가져오기가 실패합니다.
- Developer tool이 WSDL 파일이 참조하는 스키마에 액세스할 수 있어야 합니다.
- WSDL 파일에 스키마가 포함되거나 외부 스키마가 있는 경우 Developer tool이 WSDL 데이터 개체 내에 포함된 스키마를 작성합니다.
- WSDL 파일이 다른 WSDL 파일을 가져올 경우 Developer tool이 두 개의 WSDL을 결합하여 WSDL 데이터 개체를 작성합니다.
- WSDL 파일에서 여러 작업을 정의할 경우 Developer tool이 모든 작업을 WSDL 데이터 개체에 포함합니다. WSDL 데이터 개체에서 웹 서비스를 작성할 경우 하나 이상의 작업을 포함할지 선택할 수 있습니다.

WSDL 데이터 개체 개요 보기

WSDL 데이터 개체 **개요** 보기에는 WSDL 및 WSDL의 작업에 대한 일반 정보가 표시됩니다.

다음 테이블에는 WSDL 데이터 개체에 대해 구성하는 일반 속성이 설명되어 있습니다.

속성	설명
이름	WSDL 데이터 개체의 이름입니다.
설명	WSDL 데이터 개체에 대한 설명입니다.

다음 테이블에는 WSDL 데이터 개체에 정의된 작업의 열이 설명되어 있습니다.

속성	설명
작업	WSDL이 작업에 대한 메시지 형식 및 프로토콜을 정의하는 위치입니다.
입력	작업 입력과 연관된 WSDL 메시지 이름입니다.

속성	설명
출력	작업 출력과 연관된 WSDL 메시지 이름입니다.
결함	작업 결함과 연관된 WSDL 메시지 이름입니다.

WSDL 데이터 개체 고급 보기

WSDL 데이터 개체 **고급** 보기에 WSDL 데이터 개체의 고급 속성이 표시됩니다.

다음 테이블에는 WSDL 데이터 개체의 고급 속성이 설명되어 있습니다.

속성	설명
연결	웹 서비스 소비자 변환에 대한 기본 웹 서비스 연결입니다.
파일 위치	WSDL 파일이 있는 위치입니다.

WSDL 데이터 개체 가져오기

WSDL에서 웹 서비스를 작성하거나 웹 서비스 소비자 변환을 작성하려면 WSDL 데이터 개체를 가져오십시오. WSDL 파일 또는 WSDL 위치를 가리키는 URI에서 WSDL 데이터 개체를 가져올 수 있습니다. SOAP 1.1 또는 SOAP 1.2 바인딩 작업이나 둘 다를 포함하는 WSDL 파일에서 WSDL 데이터 개체를 가져올 수 있습니다.

1. **파일 > 새로 만들기 > 데이터 개체**를 클릭합니다.
2. **WSDL 데이터 개체**를 선택하고 **다음**을 클릭합니다.
새 **WSDL 데이터 개체** 대화 상자가 나타납니다.
3. **WSDL** 옵션 옆에 있는 **찾아보기** 옵션을 클릭하고 WSDL 위치를 입력합니다. 그런 다음 **확인**을 클릭합니다.
WSDL 위치를 입력할 경우 WSDL 파일을 찾아보거나 WSDL에 대한 URI를 입력할 수 있습니다.
참고: URI에 영어 이외의 문자가 포함된 경우 가져오기가 실패할 수 있습니다. URI를 브라우저의 주소 표시줄에 복사합니다. 위치를 다시 브라우저에 복사합니다. Developer 도구는 브라우저의 인코딩된 URI를 허용합니다.
4. WSDL 이름을 입력합니다.
5. **위치** 옵션 옆에 있는 **찾아보기**를 클릭하여 WSDL 데이터 개체를 가져올 프로젝트 또는 폴더 위치를 선택합니다.
6. **다음**을 클릭하여 WSDL의 작업을 확인합니다.
7. **마침**을 클릭합니다.
개체 탐색기 보기의 프로젝트 또는 폴더에서 **실제 데이터 개체** 아래에 데이터 개체가 표시됩니다.

WSDL 동기화

WSDL 파일이 변경될 경우 WSDL 데이터 개체를 동기화할 수 있습니다. WSDL 데이터 개체를 동기화할 경우 Developer 도구가 WSDL 파일에서 개체 메타데이터를 다시 가져옵니다.

WSDL 데이터 개체를 사용하여 웹 서비스 또는 웹 서비스 소비자 변환을 작성할 수 있습니다. WSDL 데이터 개체를 업데이트할 경우 Developer 도구에서 WSDL을 참조하는 개체를 업데이트하고, 사용자가 해당 개체를 열 때 변경됨으로 표시합니다. Developer 도구가 새 WSDL을 이전 WSDL과 비교할 때 이름 특성을 통해 WSDL 구성 요소를 식별합니다.

이름 특성이 변경되지 않은 경우 **Developer** 도구는 WSDL 구성 요소를 참조하는 개체를 업데이트합니다. 예를 들어, WSDL 파일을 편집하고 단순 요소 "CustID"의 유형을 `xs:string`에서 `xs:integer`로 변경하십시오. WSDL 데이터 개체를 동기화할 경우 **Developer** 도구가 CustID 요소를 참조하는 모든 웹 서비스 및 웹 서비스 소비자 변환의 요소 유형을 업데이트합니다.

이름 특성이 변경된 경우 WSDL 구성 요소를 참조하는 개체를 사용자가 열 때 **Developer** 도구가 해당 개체를 변경됨으로 표시합니다. 예를 들어, WSDL을 편집하고 요소 이름을 "Resp"에서 "RespMsg"로 변경하십시오. 그런 다음 WSDL을 동기화합니다. 요소를 참조하는 웹 서비스를 열 경우 **Developer** 도구가 편집기의 웹 서비스 이름에 별표를 표시하여 웹 서비스에 변경 사항이 포함되어 있음을 나타냅니다. **Developer** 도구에서 웹 서비스의 요소 이름을 업데이트하지만 새 요소가 포트에 매핑되는 방법을 확인할 수 없습니다. Resp 요소가 입력 변환 또는 출력 변환의 포트에 매핑된 경우 RespMsg 요소를 해당 포트에 매핑해야 합니다.

Developer 도구가 WSDL 데이터 개체를 업데이트하기 전에 WSDL 파일의 유효성을 검사합니다. WSDL 파일에 오류가 포함된 경우 **Developer** 도구가 파일을 가져오지 않습니다.

WSDL 데이터 개체 동기화

WSDL 파일이 변경될 경우 WSDL 데이터 개체를 동기화하십시오.

1. **개체 탐색기** 보기에서 WSDL 데이터 개체를 마우스 오른쪽 단추로 클릭하고, **동기화**를 선택합니다.

WSDL 데이터 개체 동기화 대화 상자가 나타납니다.

2. **WSDL** 필드 옆에 있는 **찾아보기**를 클릭하고 WSDL 위치를 입력합니다. 그런 다음 **확인**을 클릭합니다.

WSDL 위치를 입력할 경우 WSDL 파일을 찾아보거나 WSDL에 대한 URI를 입력할 수 있습니다.

참고: URI에 영어 이외의 문자가 포함된 경우 가져오기가 실패할 수 있습니다. URI를 브라우저의 주소 표시줄에 복사합니다. 위치를 다시 브라우저에 복사합니다. **Developer** 도구는 브라우저의 인코딩된 URI를 허용합니다.

3. WSDL 이름 및 위치를 확인합니다.
4. **다음**을 클릭하여 WSDL의 작업을 확인합니다.
5. **마침**을 클릭합니다.

Developer 도구에서 WSDL을 참조하는 개체를 업데이트하고, 사용자가 해당 개체를 열었을 때 변경된 상태로 표시합니다.

인증서 관리

클라이언트 인증이 필요한 URL에서 WSDL 데이터 개체 및 스키마 개체를 가져오려면 **Developer** 도구에서 인증서를 사용해야 합니다.

기본적으로 URL을 호스트하는 서버가 트러스트된 인증서를 사용할 경우 **Developer** 도구는 클라이언트 인증이 필요한 URL에서 개체를 가져옵니다. URL을 호스트하는 서버에서 트러스트되지 않은 인증서를 사용할 경우 **Developer** 도구에 트러스트되지 않은 인증서를 추가하십시오. **Developer** 도구에 트러스트되지 않은 인증서를 추가하지 않은 경우 **Developer** 도구가 개체를 가져올 수 없습니다. 개체를 가져올 URL의 서버 관리자에게 인증서 파일 및 암호를 요청하십시오.

Developer 도구에 추가한 인증서가 **Developer** 도구 시스템에서 수행하는 가져오기에 적용됩니다. **Developer** 도구는 모델 리포지토리에 인증서를 저장하지 않습니다.

Informatica Developer 인증서 속성

트러스트되지 않은 인증서를 사용하여 클라이언트를 인증해야 하는 URL에서 개체를 가져오려는 경우 Developer tool에 인증서를 추가해야 합니다.

다음 테이블에는 인증서 속성이 설명되어 있습니다.

속성	설명
호스트 이름	URL을 호스트하는 서버의 이름입니다.
포트 번호	URL의 포트 번호입니다.
인증서 파일 경로	클라이언트 인증서 파일의 위치입니다.
암호	클라이언트 인증서 파일의 암호입니다.

Informatica Developer에 인증서 추가

인증서를 추가할 때는 트러스트되지 않은 인증서를 통한 클라이언트 인증이 필요한 URL에서 개체를 가져올 때 Developer tool에서 사용할 인증서 속성을 구성해야 합니다.

1. 창 > 기본 설정을 클릭합니다.
2. Informatica > 웹 서비스 > 인증서를 선택합니다.
3. 추가를 클릭합니다.
4. 인증서 속성을 구성합니다.
5. 확인을 클릭합니다.

동기화

실제 데이터 개체의 소스가 변경되는 경우 해당 개체를 동기화할 수 있습니다. 실제 데이터 개체를 동기화하면 Developer 도구가 선택된 소스에서 개체 메타데이터를 다시 가져옵니다.

모든 실제 데이터 개체를 동기화할 수 있습니다. 관계형 데이터 개체 또는 사용자 지정된 데이터 개체를 동기화하는 경우 Developer 도구에서 정의하는 키 관계를 유지하거나 덮어쓸 수 있습니다.

해당 소스가 변경되는 경우 사용자 지정된 데이터 개체를 동기화하도록 구성할 수 있습니다. 예를 들어 사용자 지정된 데이터 개체가 관계형 데이터 개체를 소스로 사용하고 사용자가 관계형 데이터 개체에 열을 추가할 수 있습니다. Developer 도구가 사용자 지정된 데이터 개체에 열을 추가합니다. 해당 소스가 변경될 때 사용자 지정된 데이터 개체를 동기화하려면 사용자 지정된 데이터 개체의 개요 속성에서 입력 및 출력 동기화 옵션을 선택합니다.

실제 데이터 개체를 동기화하려면 개체 탐색기 보기에서 개체를 마우스 오른쪽 단추로 클릭하고 동기화를 선택합니다.

Informatica Developer의 플랫폼 파일 데이터 개체 동기화

Informatica Developer에서 외부 플랫폼 파일 데이터 소스에 대한 변경 내용을 해당 데이터 개체와 동기화할 수 있습니다. 플랫폼 파일 동기화 마법사를 사용하여 데이터 개체를 동기화합니다.

1. Object Explorer 보기에서 플랫폼 파일 데이터 개체를 선택합니다.

2. 마우스 오른쪽 단추를 클릭하고 **동기화**를 선택합니다.
다음 이미지는 데이터 개체에 대한 동기화 옵션을 보여 줍니다.



플랫 파일 데이터 개체 동기화 마법사가 나타납니다.

3. 플랫 파일 데이터 개체 동기화 마법사에서 **기존 플랫 파일 선택** 필드에 있는 플랫 파일 경로를 확인합니다.
다음 이미지는 플랫 파일 데이터 개체 동기화 마법사를 보여 줍니다.



4. **다음**을 클릭합니다.
5. 필요에 따라 코드 페이지, 형식, 구분자로 분리된 형식 속성 및 열 속성을 선택합니다.
6. **마침**을 클릭한 다음 **확인**을 클릭합니다.

Informatica Developer의 관계형 데이터 개체 동기화

Informatica Developer에서 관계형 데이터 소스의 외부 데이터 소스 변경 내용을 해당 데이터 개체와 동기화할 수 있습니다. 외부 데이터 소스 변경 내용으로는 열 추가/변경/제거, 규칙 변경 등이 있습니다.

1. **개체 탐색기** 보기에서 관계형 데이터 개체를 선택합니다.
2. 마우스 오른쪽 단추를 클릭하고 **동기화**를 선택합니다.
작업을 확인하라는 메시지가 표시됩니다.
3. 동기화 프로세스를 완료하려면 **확인**을 클릭합니다.
동기화 프로세스 상태 메시지가 나타납니다.
4. **동기화가 완료되었습니다.** 메시지가 표시되면 **확인**을 클릭합니다.
메시지에는 데이터 개체에 대한 메타데이터 변경 내용의 요약이 표시됩니다.

실제 데이터 개체 문제 해결

관계형 데이터 개체 또는 사용자 지정된 데이터 개체 소스 변환을 미리 보려고 하지만 미리보기가 실패합니다.

리소스 소유자 이름이 올바른지 확인하십시오.

관계형 리소스를 가져올 때 테이블을 가져올 사용자 이름과 스키마가 일치하지 않으면 **Developer tool**이 소유자 이름을 가져옵니다. 테이블을 가져온 사용자 이름과 스키마는 일치하지만 데이터베이스 기본 스키마의 이름이 다를 경우 데이터 통합 서비스가 테이블이 존재하지 않는 데이터베이스 기본 스키마에 대해 미리보기 쿼리를 실행하므로 미리보기가 실패합니다.

관계형 데이터 개체 또는 소스 변환을 업데이트하고 올바른 리소스 소유자 이름을 입력하십시오. 소유자 이름은 관계형 데이터 개체 또는 소스 변환에서 **고급** 속성에 나타납니다.

Databricks 실제 개체에서 매핑 유효성 검사가 실패합니다.

Databricks에서 실제 데이터 개체를 가져올 때 이스케이프 문자를 사용한 경우 매핑 유효성 검사가 실패합니다. 개체 이름에 이스케이프 문자를 사용하지 마십시오.

제 6 장

플랫 파일 데이터 개체

이 장에 포함된 항목:

- [플랫 파일 데이터 개체 개요, 87](#)
- [소스 파일 이름 생성, 88](#)
- [플랫 파일 데이터 개체 개요 속성, 88](#)
- [플랫 파일 데이터 개체 고급 속성, 89](#)
- [제어 파일, 95](#)
- [런타임에 열 업데이트, 96](#)
- [제어 파일에서 열 메타데이터 생성, 97](#)
- [Excel에서 플랫 파일 데이터 개체로 복사, 99](#)
- [플랫 파일 데이터 개체 작성, 101](#)

플랫 파일 데이터 개체 개요

플랫 파일 실제 데이터 개체를 작성하여 매핑, 맵셋 또는 프로필에 포함할 수 있습니다. 플랫 파일 데이터 개체는 매핑과 맵셋에 읽기, 쓰기 및 조회 변환으로 추가할 수 있습니다. 플랫 파일 데이터 개체에서 프로필을 작성하고 실행할 수 있습니다.

플랫 파일 데이터 개체는 구분자로 분리되거나 고정 너비일 수 있습니다. 이진 데이터를 포함하지 않는 고정 너비 플랫 파일과 구분자로 분리된 플랫 파일에서 플랫 파일 데이터 개체를 작성할 수 있습니다.

런타임 시 플랫 파일 데이터 소스의 변경 내용을 처리하도록 플랫 파일 데이터 개체를 구성할 수 있습니다. 플랫 파일 또는 제어 파일에 있는 정보를 사용하여 플랫 파일 데이터 개체의 열 이름을 생성할 수도 있습니다. 제어 파일에는 열 이름, 전체 자릿수, 소수 자릿수 및 처리할 바이트 수 같은 정보가 포함됩니다.

플랫 파일 데이터 개체를 작성한 후에는 다음과 같은 보기를 사용하여 플랫 파일 속성을 구성합니다.

개요 보기

플랫 파일 데이터 개체의 이름과 설명을 구성하고 열 속성을 업데이트합니다.

매개 변수 보기

플랫 파일 데이터 개체의 매개 변수를 작성합니다.

고급 보기

데이터 통합 서비스가 플랫 파일에서 데이터를 읽고 쓸 때 사용하는 형식 및 런타임 속성을 구성합니다.

플랫 파일 데이터 개체를 읽기 또는 쓰기 변환으로 매핑에 추가하면 해당 플랫 파일 데이터 개체의 형식과 런타임 속성을 **속성** 보기에서 확인할 수 있습니다. 매핑 내의 플랫 파일 속성은 편집할 수 없습니다.

소스 파일 이름 생성

플랫 파일 데이터 개체에 파일 이름 열을 추가할 수 있습니다. 파일 이름 열을 사용하면 데이터의 특정 레코드가 포함된 소스 파일을 식별할 수 있습니다. 이는 데이터를 여러 소스에서 가져올 때 유용합니다.

플랫 파일 데이터 개체의 개요 보기에서 파일 이름 열을 구성할 수 있습니다. 파일 이름 열은 플랫 파일 데이터 개체에서 선택적 열입니다. 플랫 파일 데이터 개체에서 파일 이름 열 포트가 포함된 각 소스 행에 소스 파일 이름을 쓰도록 매핑을 구성할 수 있습니다. 파일 이름 열에는 정규화된 경로 및 파일 이름이 포함되어 있습니다.

파일 이름 열 포트는 소스 데이터 개체의 마지막 열로 표시됩니다. 소스 데이터 개체에서 하나의 파일 이름 열 포트만 추가할 수 있습니다. 포트에 **FileName**과 같은 이름이 있고 파일 이름 열 포트를 생성하려고 하는 경우 파일 이름 열 포트의 이름이 **FileName1**로 지정됩니다.

예를 들어 매핑은 파일 목록이 포함된 소스 파일을 사용합니다. 파일 목록의 파일 이름은 조직 전체의 부서 이름을 나타냅니다. 예를 들어 **SYSA_Finance.txt** 파일에는 재무 부서의 데이터가 포함되어 있습니다. 매핑에서 문자열 함수를 사용하여 파일 이름 열의 출력에서 부서 이름을 추출할 수 있습니다. 추출된 부서 이름을 사용하여 각 부서에 대해 다르게 데이터를 처리할 수 있습니다.

플랫 파일 데이터 개체 개요 속성

데이터 통합 서비스는 데이터를 플랫 파일에서 읽거나 플랫 파일에 쓸 때 개요 속성을 사용합니다. 개요 속성은 플랫 파일 데이터 개체에 적용되는 일반 속성이 포함됩니다. 또한 플랫 파일 데이터 개체의 열에 적용되는 열 속성도 포함됩니다. **Developer** 도구는 플랫 파일의 개요 속성을 **개요** 보기에 표시합니다.

다음 표에서는 플랫 파일에 대해 구성하는 일반 속성에 대해 설명합니다.

속성	설명
이름	플랫 파일 데이터 개체의 이름입니다.
설명	플랫 파일 데이터 개체에 대한 설명입니다.

다음 표에서는 플랫 파일에 대해 구성하는 열 속성에 대해 설명합니다.

속성	설명
이름	열 또는 파일 이름 열 포트의 이름입니다.
원시 유형	열의 원시 데이터 유형입니다.
처리할 바이트(고정 너비 플랫 파일)	데이터 통합 서비스가 열에 대해 읽거나 쓰는 바이트의 수입니다.
전체 자릿수	숫자 데이터 유형의 최대 유효 자릿수 또는 문자열 데이터 유형의 최대 문자 수입니다. 숫자 데이터 유형의 경우 전체 자릿수에 소수 자릿수가 포함됩니다.
소수 자릿수	숫자 값의 소수점 뒤 최대 자릿수입니다.

속성	설명
형식	숫자 및 날짜/시간 데이터 유형의 열 형식입니다. 숫자 데이터 유형의 경우 형식이 1000 단위 구분 기호와 소수 구분 기호를 정의합니다. 기본값은 1000 단위 구분 기호와 소수 구분 기호 마침표(.)가 없습니다. 날짜/시간 데이터 유형의 경우 형식이 년, 월, 일, 시의 표시 형식을 정의합니다. 또한 필드 너비도 정의합니다. 기본값은 "A 19 YYYY-MM-DD HH24:MI:SS"입니다.
표시 유형	데이터 통합 서비스가 열에서 데이터를 읽을 수 있는지 또는 열에 데이터를 쓸 수 있는지 결정합니다. 예를 들어 표시 유형이 읽기일 경우 데이터 통합 서비스가 열에서 데이터를 읽을 수 있지만 데이터를 열에 쓸 수는 없습니다. 플랫 파일 데이터 개체의 경우 이 속성은 읽기 전용입니다. 표시 유형은 항상 읽기 및 쓰기입니다.
설명	열에 대한 설명입니다.

플랫 파일 데이터 개체 고급 속성

데이터 통합 서비스는 데이터를 플랫 파일에서 읽거나 플랫 파일에 쓸 때 고급 속성을 사용합니다. 고급 속성은 형식 및 런타임 속성이 포함됩니다. **Developer tool**은 플랫 파일의 고급 속성을 **고급** 보기에 표시합니다.

고급 보기에는 축소하거나 확장할 수 있는 속성 섹션이 포함되어 있습니다. 열 형식 섹션은 구분자로 분리된 열 형식을 구성했는지 아니면 고정 너비 열 형식을 구성했는지에 따라 다르게 표시됩니다.

다음 이미지는 **고급** 보기의 속성 섹션을 보여 줍니다.



형식 속성

Developer tool은 고급 보기의 **형식** 섹션에 플랫폼 파일 데이터 개체의 형식 속성을 표시합니다.

다음 표에는 플랫폼 파일 데이터 개체에 대해 구성 가능한 형식 속성이 설명되어 있습니다.

속성	설명
코드 페이지	플랫폼 파일 데이터 개체의 코드 페이지입니다. 소스 파일의 경우 대상 코드 페이지의 하위 집합인 소스 코드 페이지를 사용합니다. 조 회 파일의 경우 소스 코드 페이지의 상위 집합 및 대상 코드 페이지의 하위 집합인 코드 페이지를 사용합니다. 대상 파일의 경우 소스 코드 페이지의 상위 집합인 코드 페이지를 사용합니다. 기본값은 "MS Windows Latin 1 (ANSI), Latin 1의 상위 집합"입니다.
날짜/시간 형식	날짜/시간 값의 표시 형식 및 필드 너비를 정의합니다. 기본값은 "19 YYYY-MM-DD HH24:MI:SS"입니다.
1000 단위 구분 기호	숫자 값의 1000 단위 구분 기호입니다. 기본값은 없음입니다.
소수 구분 기호	숫자 값의 소수 구분 기호입니다. 기본값은 마침표(.)입니다.
열 형식	플랫폼 파일의 형식(구분자로 분리됨 또는 고정 너비)입니다.

열 형식: 구분자로 분리된 속성

구분자로 분리된 플랫폼 파일의 경우, Developer tool은 고급 보기의 **열 형식: 구분자로 분리됨** 섹션에 구분자로 분리된 속성을 표시합니다.

다음 표에는 플랫폼 파일 데이터 개체에 대해 구성 가능한 구분자로 분리된 속성이 설명되어 있습니다.

속성	설명
구분자	데이터의 열을 구분하는 데 사용되는 문자입니다. 구분자 필드에서 문자를 선택하거나 속성에 매개 변수를 할당합니다. 구분자는 인쇄 가능한 문자여야 하며, 선택한 경우, 텍스트 한정자 및 이스케이프 문자와 달라야 합니다. 기본값은 쉼표입니다.
텍스트 한정자	텍스트 문자열의 경계를 정의하는 따옴표 문자입니다. 따옴표 문자를 선택하면 Developer tool이 따옴표 쌍 안의 구분자를 무시합니다. 기본값은 따옴표 사용 안 함입니다.
가져오기 시작 행	데이터 통합 서비스가 데이터 가져오기를 시작하는 행입니다. 헤더 행을 건너뛰려면 이 옵션을 선택합니다. 기본값은 1입니다.
행 구분자	데이터의 행을 구분하는 문자의 8진수 코드입니다. 기본값은 줄 바꿈 \012 LF(\n)입니다. 참고: 행 구분자는 소스 데이터 읽기에 적용됩니다. 데이터 통합 서비스가 대상 파일에 쓰는 경우 항상 기본 구분자인 \n이 사용됩니다.

속성	설명
이스케이프 문자	따옴표로 묶이지 않은 문자열에서 구분자가 이스케이프 문자 다음에 오는 문자일 경우에 구분자를 이스케이프하는 데 사용되는 문자입니다. 이스케이프 문자를 지정하면 데이터 통합 서비스가 구분자 문자를 문자열에 포함된 일반 문자로 읽습니다. 참고: 소스 파일에 따옴표 또는 이스케이프 문자가 포함되어 있지 않으면 매핑 성능이 다소 향상될 수 있습니다.
데이터에 이스케이프 문자 유지	출력 문자열에 이스케이프 문자를 포함합니다. 기본값은 비활성화됩니다.
연속 구분자를 하나로 처리	데이터 통합 서비스가 하나 이상의 연속 열 구분자를 하나로 처리하도록 합니다. 그렇지 않으면 데이터 통합 서비스가 두 개의 연속 구분자를 null 값으로 읽습니다. 기본값은 비활성화됩니다.

열 형식: 고정 너비 속성

고정 너비 플랫폼 파일의 경우, Developer tool은 고급 보기의 **열 형식: 고정 너비** 섹션에 고정 너비 속성을 표시합니다.

다음 표에는 플랫폼 파일 데이터 개체에 대해 구성 가능한 고정 너비 속성이 설명되어 있습니다.

속성	설명
Null 문자 유형	Null 문자 유형(텍스트 또는 이진)입니다.
Null 문자 값	null 값을 나타내는 데 사용되는 문자입니다. null 문자는 파일 코드 페이지의 올바른 문자이거나 0 - 255의 이진 값일 수 있습니다.
Null 문자 반복	소스 파일의 경우 데이터 통합 서비스가 단일 필드의 null 문자 반복을 하나의 null 값으로 읽도록 합니다. 대상 파일의 경우 데이터 통합 서비스가 가능한 한 많은 null 문자를 대상 필드로 쓰도록 합니다. 이 옵션을 활성화하지 않으면 데이터 통합 서비스가 필드의 시작 부분에 null 문자 하나를 입력하여 null 값을 나타냅니다. 기본값은 비활성화됩니다.
가져오기 시작 행	데이터 통합 서비스가 데이터 가져오기를 시작하는 행입니다. 헤더 행을 건너뛰려면 이 옵션을 선택합니다. 기본값은 1입니다.
레코드 간 건너뛰기 바이트 수	한 행의 마지막 열과 다음 행의 첫 번째 열 사이의 바이트 수입니다. 데이터 통합 서비스는 캐리지 리턴 문자 또는 줄 바꿈 문자를 읽지 않도록 각 행의 끝에 입력된 바이트 수를 건너뛰습니다. UNIX 파일의 경우 1을 입력하고, DOS 파일의 경우 2를 입력합니다. 기본값은 2입니다.
순차 정렬	데이터 통합 서비스가 마지막 열의 줄바꿈 또는 캐리지 리턴 문자를 열의 끝으로 읽습니다. 파일에서 각 행의 마지막 열을 줄이기 위해 열 바꿈 또는 캐리지 리턴을 사용하는 경우 이 옵션을 선택합니다. 기본값은 비활성화됩니다.

속성	설명
후행 공백 제거	문자열 값에서 후행 공백을 제거합니다. 기본값은 비활성화됩니다.
사용자 정의 시프트 상태	열 속성에서 소스 열에 대한 시프트 상태를 선택할 수 있도록 합니다. 소스 파일에 다중 바이트 및 단일 바이트 데이터가 모두 포함되었지만 SI(Shift-In) 및 SO(Shift-Out) 키가 포함되지 않은 경우 이 옵션을 선택합니다. 다중 바이트 파일 소스에 shift 키가 포함되지 않은 경우 플랫폼 파일 데이터 개체에서 각 열에 대해 shift 키를 선택해야 합니다. 데이터 통합 서비스가 각 문자를 올바르게 읽을 수 있게 하려면 각 열에 대해 shift 키를 선택하십시오. 기본값은 비활성화됩니다.

런타임: 읽기 속성

Developer tool은 고급 보기의 **런타임: 읽기** 섹션에 플랫폼 파일 소스의 런타임 속성을 표시합니다. 데이터 통합 서비스는 플랫폼 파일에서 데이터를 읽을 때 이 정보를 사용합니다.

다음 표에는 플랫폼 파일 데이터 개체에 대해 구성 가능한 읽기 속성이 설명되어 있습니다.

속성	설명
입력 유형	소스 입력의 유형입니다. 다음 유형의 소스 입력을 정의할 수 있습니다. <ul style="list-style-type: none"> - 파일. 플랫폼 파일 소스를 나타냅니다. - 명령. 셸 명령에서 생성되는 파일 목록 또는 소스 데이터를 나타냅니다.
소스 유형	파일 속성이 동일한 파일의 소스 유형을 나타냅니다. 다음 소스 유형 중 하나를 선택할 수 있습니다. <ul style="list-style-type: none"> - 직접. 소스 데이터를 포함하는 소스 파일입니다. - 간접. 파일 목록을 포함하는 소스입니다. 데이터 통합 서비스가 파일 목록을 읽고 순차적 순서로 파일을 읽습니다. - 디렉터리. 디렉터리에 있는 소스 파일입니다. 소스 파일 디렉터리 속성의 디렉터리 위치를 지정해야 합니다. 데이터 통합 서비스가 오름차순으로 파일을 읽습니다. 데이터 통합 서비스는 하위 디렉터리의 파일을 읽지 않습니다.
소스 파일 이름	플랫폼 파일 소스의 파일 이름입니다.
소스 파일 디렉터리	플랫폼 파일 소스가 있는 디렉터리입니다. Informatica 서비스를 호스팅하는 시스템에서 이 디렉터리에 액세스할 수 있어야 합니다. 기본값은 SourceDir 시스템 매개 변수입니다.
동시 읽기 분할	여러 파티션에서 소스 파일의 입력 행을 읽는 순서입니다. 데이터 통합 서비스가 매핑에 대해 파티션을 작성하지 않는 경우 이 값은 무시됩니다. 다음 옵션 중 하나를 선택합니다. <ul style="list-style-type: none"> - 처리량 최적화. 데이터 통합 서비스가 입력 행 순서를 유지하지 않습니다. - 상대 순서 유지. 데이터 통합 서비스가 각 파티션에서 읽는 행에 대해 입력 행 순서를 유지합니다. - 절대 순서 유지. 데이터 통합 서비스가 모든 파티션에서 읽는 모든 행에 대해 입력 행 순서를 유지합니다.
연결 유형	연결 유형입니다. 다음 옵션 중에서 선택하십시오. <ul style="list-style-type: none"> - 없음. 소스 파일에 연결이 필요하지 않습니다. - Hadoop 파일 시스템. 소스 파일이 HDFS에 있습니다. 기본값은 없음입니다.

속성	설명
명령	<p>소스 파일 데이터를 생성하는 데 사용되는 명령입니다. 명령을 사용하여 매핑이 실행될 때 플랫폼 파일 데이터를 생성 또는 변환하고 명령의 표준 출력을 플랫폼 파일 판독기에 보냅니다. 플랫폼 파일 판독기는 표준 출력을 플랫폼 파일 소스 데이터로 읽습니다. 명령을 사용하여 소스 데이터를 생성하면 플랫폼 파일 소스를 준비할 필요가 없습니다.</p> <p>사전 매핑 명령을 사용하여 플랫폼 파일 소스를 생성하는 대신 명령 또는 스크립트를 사용하여 소스 데이터를 데이터 통합 서비스에 직접 보낼 수 있습니다. 명령을 사용하여 파일 목록을 생성할 수도 있습니다.</p> <p>예를 들어 디렉터리 목록을 파일 목록으로 사용하려면 다음과 같은 명령을 사용합니다.</p> <pre>cd MySourceFiles; ls sales-records-Sep-*-2005.dat</pre>
문자열 null 잘라내기	<p>문자열 값의 첫 번째 null 문자와 이 문자 이후의 모든 문자를 스트립합니다.</p> <p>구분자로 분리된 플랫폼 파일의 문자열에 null 문자가 포함된 경우 이 옵션을 활성화합니다. 이 옵션을 활성화하지 않으면 데이터 통합 서비스는 문자열에 null 문자가 포함된 모든 행에 대해 행 오류를 생성합니다.</p> <p>기본값은 비활성화됩니다.</p>
순차 정렬 버퍼 길이	<p>데이터 통합 서비스가 각 행에 대해 읽는 바이트 수입니다.</p> <p>이 속성은 전체 행 크기와 함께 데이터 통합 서비스가 행을 삭제하는지 여부를 결정합니다. 행이 순차 정렬 버퍼 길이 또는 전체 행 크기 중에 더 큰 값을 초과하면 데이터 통합 서비스가 해당 행을 삭제하고 매핑 로그 파일에 씁니다. 전체 행 크기를 확인하려면 열 전체 자릿수와 구분자를 추가한 후 전체에 각 문자의 최대 바이트를 곱합니다.</p> <p>기본값은 1024입니다.</p>
런타임 열 이름 생성	<p>런타임에 열 메타데이터를 생성하는 방법을 결정합니다.</p> <p>다음 옵션 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - 자동으로(Field1, Field2...). 데이터 통합 서비스가 데이터에서 열의 위치에 기반하여 열 메타데이터를 포함합니다. 플랫폼 파일의 열 메타데이터는 사용자가 플랫폼 파일 데이터 개체의 형식 속성에 포함된 가져오기 시작 행 특성에 지정하는 행에 기반하여 런타임에 구성됩니다. - 데이터 파일 헤더(첫 번째 행)에서. 플랫폼 파일의 첫 번째 행에는 데이터 통합 서비스가 열 메타데이터를 결정하는 데 사용하는 헤더 행이 포함되어 있습니다. - 제어 파일에서. 데이터 통합 서비스가 열 이름, 전체 자릿수, 데이터 유형, 소수 자릿수 및 처리할 바이트 수 같은 제어 파일 내의 데이터에 기반하여 열 메타데이터를 구성합니다.
제어 파일 이름	<p>제어 파일의 이름입니다. 제어 파일에서 런타임 열 이름을 생성하는 경우 필수 항목입니다.</p>
제어 파일 디렉터리	<p>제어 파일이 있는 디렉터리입니다. 제어 파일에서 런타임 열 이름을 생성하는 경우 필수 항목입니다.</p>
기본 필드 유형	<p>런타임에 생성되는 추가 포트의 데이터 유형입니다.</p>
기본 전체 자릿수	<p>런타임에 생성되는 추가 포트의 전체 자릿수입니다.</p>
기본 소수 자릿수	<p>런타임에 생성되는 추가 포트의 소수 자릿수입니다.</p>
제약 조건	<p>데이터 행의 값이 충족해야 하는 조건부 식입니다. 식 편집기를 사용하여 TRUE로 평가되는 식을 입력하십시오. 데이터 통합 서비스는 제약 조건을 읽으면서 TRUE로 평가되지 않는 행을 삭제합니다.</p> <p>예를 들어 소스 플랫폼 파일에 AGE 열이 있다고 가정해 보겠습니다. 이 경우 플랫폼 파일 데이터 개체에 AGE < 70인 제약 조건을 설정할 수 있습니다. 그러면 데이터 통합 서비스는 AGE < 70인 제약 조건을 사용하여 소스 플랫폼 파일에서 행을 읽습니다. 데이터 통합 서비스는 AGE >= 70인 행을 읽을 경우 해당 행을 삭제합니다.</p>

런타임: 쓰기 속성

Developer tool은 고급 보기의 **런타임: 쓰기** 섹션에 플랫폼 파일 대상의 런타임 속성을 표시합니다. 데이터 통합 서비스는 플랫폼 파일에 데이터를 쓸 때 이 정보를 사용합니다.

다음 표에는 플랫폼 파일 데이터 개체에 대해 구성 가능한 쓰기 속성이 설명되어 있습니다.

속성	설명
존재하는 경우 추가	출력 데이터를 대상 파일 및 거부 파일에 추가합니다. 이 옵션을 선택하지 않으면 데이터 통합 서비스가 데이터를 대상 파일 및 거부 파일에 쓰기 전에 이 파일들을 자릅니다. 이러한 파일이 없으면 데이터 통합 서비스가 파일을 작성합니다. 기본값은 비활성화됩니다.
없을 경우 디렉터리 작성	대상 디렉터리가 없을 경우 대상 디렉터리를 작성합니다. 기본값은 비활성화됩니다.
헤더 옵션	파일 대상에 헤더 행을 작성합니다. 다음 옵션을 선택할 수 있습니다. - 머릿글 없음. 플랫폼 파일 대상에 헤더 행을 작성하지 않습니다. - 출력 필드 이름. 출력 포트 이름을 사용하여 파일 대상에 헤더 행을 작성합니다. - 헤더 명령 출력 사용. 헤더 명령 필드의 명령을 사용하여 헤더 행을 생성합니다. 예를 들어 명령을 사용하여 파일 대상의 헤더 행에 날짜를 추가할 수 있습니다. 기본값은 헤더 없음입니다.
헤더 명령	파일 대상에서 헤더 행을 생성하는 데 사용되는 명령입니다.
바닥글 명령	파일 대상에서 바닥글 행을 생성하는 데 사용되는 명령입니다.
출력 유형	매핑의 대상 유형입니다. 대상 데이터를 플랫폼 파일에 쓰려면 파일을 선택합니다. 데이터를 명령으로 출력하려면 명령을 선택합니다.
명령	대상 데이터를 처리하는 데 사용되는 명령입니다. UNIX에서는 유효한 UNIX 명령이나 셸 스크립트를 사용합니다. 예를 들어 UNIX에서 대상 데이터로부터 압축 파일을 생성하려면 다음 명령을 사용합니다. <code>compress -c - > MyTargetFiles/MyCompressedFile.Z</code> Windows에서는 유효한 DOS 명령이나 배치 파일을 사용합니다. 플랫폼 파일 기록기가 플랫폼 파일 대상 대신 명령에 데이터를 보냅니다. 예를 들어 Windows에서 cmd를 대상 명령으로 사용하여 파일 시스템의 준비 데이터를 방지하고 모든 보안 위반을 방지합니다. 변환 태스크를 데이터 통합 서비스 대신 명령에 푸시하여 매핑 성능을 향상시킬 수 있습니다. 명령을 사용하여 대상 데이터를 정렬하거나 압축할 수도 있습니다.
병합 명령	모든 대상 파티션에 대한 병합 데이터를 처리하는 데 사용되는 병합 명령입니다. 데이터 통합 서비스가 병합 데이터를 처리하는 명령에 대해 동시 병합 유형을 사용해야 합니다. 병합 명령이 대상 데이터의 순서를 유지하지 않을 수 있습니다.
출력 파일 디렉터리	플랫폼 파일 대상의 출력 디렉터리입니다. Informatica 서비스를 호스팅하는 시스템에서 이 디렉터리에 액세스할 수 있어야 합니다. 여러 개의 파티션이 플랫폼 파일 대상에 쓰는 경우 성능을 높이려면 여러 디렉터리를 세미콜론으로 구분하여 입력합니다. 기본값은 TargetDir 시스템 매개 변수입니다.

속성	설명
출력 파일 이름	플랫 파일 대상의 파일 이름입니다. 여러 파티션이 플랫 파일 대상에 쓰고 대상 데이터를 병합하지 않도록 선택하는 경우 각 파티션이 <output_file_name><partition_number>.out라는 별도의 출력 파일에 씁니다.
병합 유형	데이터 통합 서비스가 분할된 대상의 데이터에 대해 수행하는 병합의 유형입니다. 데이터 통합 서비스가 대상에 대해 파티션을 작성하지 않는 경우 이 값을 무시합니다. 다음 옵션 중 하나를 선택합니다. - 병합 안 함. 데이터 통합 서비스가 각 파티션에 대해 별개의 파일에 대상 출력을 동시에 씁니다. - 순차. 데이터 통합 서비스가 각 파티션에 대해 출력 파일을 작성한 다음 매핑의 끝에 모든 출력 파일을 단일 병합 파일로 병합합니다. - 파일 목록. 데이터 통합 서비스가 각 파티션에 대해 대상 파일을 작성하고 각 파일의 경로를 포함하는 파일 목록을 작성합니다. - 동시. 데이터 통합 서비스가 모든 대상 파티션의 데이터를 동시에 병합 파일에 씁니다. 데이터 통합 서비스가 모든 파티션에 대해 동시에 병합 파일에 쓰기 때문에 병합 파일에서 데이터의 정렬 순서가 순차적이지 않을 수 있습니다.
병합 파일 디렉터리	모든 대상 파티션에 대한 병합 파일의 디렉터리입니다. Informatica 서비스를 호스팅하는 시스템에서 이 디렉터리에 액세스할 수 있어야 합니다. 기본값은 TargetDir 시스템 매개 변수입니다.
병합 파일 이름	모든 대상 파티션에 대한 병합 파일의 이름입니다. 기본값은 출력 파일 이름입니다.
연결 유형	연결 유형입니다. 다음 옵션 중에서 선택하십시오. - 없음. 대상 파일에 연결이 필요하지 않습니다. 대상 파일 위치는 출력 파일 디렉터리에 지정됩니다. - Hadoop 파일 시스템. 대상 파일이 HDFS에 있습니다. 기본값은 없음입니다.

제어 파일

데이터 통합 서비스는 제어 파일에 기반하여 플랫 파일의 열 메타데이터를 업데이트할 수 있습니다.

제어 파일은 필드 이름, 데이터 유형, 전체 자릿수 및 소수 자릿수가 포함된 단순 텍스트 파일입니다. 제어 파일의 각 행에는 데이터 열 하나가 포함됩니다.

런타임 시 제어 파일에서 열 이름을 생성하려면 플랫 파일 데이터 개체의 고급 속성에서 **런타임 열 이름 생성** 속성에 대해 **제어 파일에서**를 선택하십시오. 런타임에 열 메타데이터를 가져오려면 읽기 변환도 구성해야 합니다.

제어 파일에는 다음의 데이터 유형을 지정할 수 있습니다.

- Bigint
- 날짜/시간
- 10진수
- 배정밀도
- 정수
- 문자열

- 텍스트
- TimestampwithTZ

제어 파일을 사용하면 플랫폼 파일 데이터 개체에 기반하여 읽기 변환의 런타임 열 이름을 생성하거나, 플랫폼 파일 데이터 개체를 작성할 수 있습니다. 데이터 통합 서비스는 제어 파일에 있는 정보를 사용하여 플랫폼 파일에 대한 열 메타데이터를 구성합니다. 열 이름, 전체 자릿수, 데이터 유형 및 소수 자릿수 특성은 쉼표로 분리됩니다. 줄 바꿈 문자는 전체 열에 걸쳐 정보를 분리합니다.

런타임에 열 업데이트

소스 메타데이터에 대한 변경 내용을 런타임에 수용하도록 플랫폼 파일 데이터 개체를 구성할 수 있습니다. 데이터 통합 서비스는 플랫폼 파일 실제 데이터 개체의 열과 비교하여 열의 개수 또는 순서가 다른 플랫폼 파일에서 데이터를 읽을 수 있습니다.

예를 들어 다른 부서에서 제공한 소스를 가진 매핑이 있습니다. 소스를 제공한 부서는 소스 파일의 열 순서를 보장할 수 없습니다. 경우에 따라 부서에서 파일의 열을 변경하거나 추가 열을 포함할 수 있습니다. 소스 메타데이터에 대한 변경 내용을 허용하도록 플랫폼 파일 데이터 개체를 구성할 수 있습니다.

소스의 모든 행, 플랫폼 파일 헤더 또는 제어 파일에서 자동으로 런타임 열 이름을 생성할 수 있습니다.

입력 규칙을 사용하면 플랫폼 파일 데이터 개체에 기반하여 읽기 변환에서 런타임 열을 매핑의 다음 변환에 프로잭션할 수 있습니다.

데이터 개체 탭에서 **런타임 시 데이터 소스에서 데이터 개체 열 가져오기**를 선택하도록 읽기 변환 데이터 개체 속성을 구성할 수 있습니다. 속성을 구성한 후에는 플랫폼 파일 데이터 개체의 옵션에 따라 데이터 통합 서비스가 데이터를 처리하는 프로세스가 결정됩니다.

자동으로 런타임 열 이름 생성

데이터 통합 서비스는 열 위치에 기반하여 플랫폼 파일의 열 메타데이터를 자동으로 업데이트할 수 있습니다.

데이터 통합 서비스는 사용자가 플랫폼 파일의 형식 속성에서 가져오기를 시작하도록 지정한 열에 기반하여 플랫폼 파일의 열 메타데이터를 동적으로 업데이트합니다. 런타임 열의 열 메타데이터에는 열 위치는 변경되지 않고 기본 데이터 유형, 전체 자릿수 및 소수 자릿수가 사용됩니다.

런타임 시 열 이름을 자동으로 생성하려면 플랫폼 파일 데이터 개체의 고급 속성에서 **런타임 열 이름 생성** 속성에 대해 **자동으로(Field1, Field2...)**를 선택하십시오. 런타임에 열 메타데이터를 가져오려면 읽기 변환도 구성해야 합니다.

예를 들어 런타임 시 변경 내용을 허용하도록 플랫폼 파일 데이터 개체를 구성하려고 합니다. 플랫폼 파일의 데이터가 열 위치에 기반하여 열 메타데이터를 결정합니다. Dept, Name, Place 열이 포함된 플랫폼 파일 데이터 개체를 정의한다고 가정해 보겠습니다. 데이터 통합 서비스에서 런타임 시 플랫폼 파일에 대해 열 메타데이터를 동적으로 구성할 것입니다.

입력 파일에는 다음 정보가 포함됩니다.

```
HR,Bob,Chicago,US,87675
Finance,Mary,California,US,65437,t567,4200
```

형식 속성에서 행 1부터 데이터 가져오기를 시작하도록 플랫폼 파일 데이터 개체를 구성했습니다. 첫 번째 행에 열이 5개 있기 때문에 데이터 통합 서비스는 열 5개를 사용하여 플랫폼 파일 데이터 개체를 처리합니다. 데이터 통합 서비스는 위치(Field4와 Field5)에 기반하여 고유한 이름을 가진 런타임 열을 플랫폼 파일 데이터 개체에 추가적으로 작성합니다. 데이터 통합 서비스는 두 번째 행의 추가 열은 처리하지 않습니다.

플랫 파일 데이터 개체를 매핑에 추가할 경우 런타임에 열 메타데이터를 가져오도록 읽기 변환을 구성해야 합니다. 런타임 시 데이터 통합 서비스는 플랫 파일 데이터 개체를 구성하고 다음 값으로 플랫 파일의 행을 처리합니다.

Dept	Name	Place	Field4	Field5
HR	Bob	Chicago	US	87675
Finance	Mary	California	US	65437

데이터 파일 헤더에서 런타임 열 이름 생성

데이터 통합 서비스는 헤더 행을 사용하여 열 메타데이터를 결정할 수 있습니다. 플랫 파일의 첫 번째 행이 헤더입니다.

런타임 열의 열 메타데이터에는 기본 데이터 유형, 전체 자릿수 및 소수 자릿수가 사용되며, 열 위치는 런타임에 변경할 수 있습니다.

런타임 시 데이터 파일 헤더에서 열 이름을 생성하려면 플랫 파일 데이터 개체의 고급 속성에서 **런타임 열 이름 생성** 속성에 대해 **데이터 파일 헤더(첫 번째 행)**에서 선택하십시오. 런타임에 열 메타데이터를 가져오려면 읽기 변환도 구성해야 합니다.

예를 들어 런타임 시 변경 내용을 허용하도록 플랫 파일 데이터 개체를 구성하려고 합니다. 플랫 파일의 데이터가 데이터 파일에 기반하여 열 메타데이터를 결정합니다.

Name, Age, Dept 열이 포함된 플랫 파일 데이터 개체를 정의했다고 가정해 보겠습니다. 런타임 시 데이터 파일 헤더에서 열 이름을 생성하도록 플랫 파일 데이터 개체 고급 속성을 구성할 수 있습니다.

데이터 플랫 파일에는 다음 정보가 포함됩니다.

Dept	Name	Place	Country	Phone_No.
HR	Bob	Chicago	US	87675
Finance	Mary	California	US	65437

플랫 파일 데이터 개체를 매핑에 추가할 경우 런타임에 열 메타데이터를 가져오도록 읽기 변환을 구성해야 합니다. 런타임 시 데이터 통합 서비스는 Dept, Name, Place, Country 및 Phone_No 열이 포함된 플랫 파일 데이터 개체를 생성합니다. Age 열은 플랫 파일 헤더 행에 표시되지 않기 때문에 런타임 시 구성된 플랫 파일 데이터 개체에는 Age 열이 없습니다.

데이터 통합 서비스는 매핑 규칙에 따라 Country 및 Phone Number 같은 런타임 열을 다운스트림 변환에 전달합니다.

제어 파일에서 열 메타데이터 생성

제어 파일은 플랫 파일 데이터 개체를 작성하는 기반으로 사용할 수 있는 단순 텍스트 파일입니다. 플랫 파일 데이터 개체에 기반하여 런타임 시 읽기 변환에 대해 열을 업데이트하는 경우에도 제어 파일을 사용할 수 있습니다.

데이터 통합 서비스는 런타임에 열을 생성할 때 플랫 파일 데이터 개체 속성에 지정된 10진수 및 1000 단위 구분 기호를 사용합니다. 열이 플랫 파일 데이터 개체에 있는 경우 제어 파일의 열 이름만 지정해야 합니다.

플랫 파일 소스의 고급 보기에서 런타임 속성에 소스 유형을 간접 유형으로 지정하면 런타임에 제어 파일을 사용하여 열 이름을 생성할 수 있습니다.

구분자로 분리된 플랫 파일 또는 고정 너비 플랫 파일에 대해 제어 파일을 작성할 수 있습니다.

플랫 파일 데이터 개체의 런타임 속성에 제어 파일 이름과 제어 파일 디렉터리를 매개 변수화할 수 있습니다.

데이터 통합 서비스는 제어 파일에 지정하는 열 메타데이터와 입력 데이터에 기반하여 제어 파일을 처리합니다.

런타임 시 제어 파일에서 열 이름을 생성하고 소스 데이터 개체에 파일 이름 열이 포함되어 있을 때 데이터 통합 서비스가 중복 파일 이름으로 유효성 검사 오류를 기록합니다.

제어 파일 형식

구분자로 분리된 플랫폼 파일과 고정 너비 플랫폼 파일은 사용하는 제어 파일 형식이 서로 다릅니다. 제어 파일을 사용하면 런타임 시 열 메타데이터를 생성하거나, 플랫폼 파일 데이터 개체를 작성할 수 있습니다.

다음 형식 중 하나를 사용하여 제어 파일을 작성할 수 있습니다.

구분자로 분리된 제어 파일

구분자로 분리된 제어 파일에서 각 행의 형식은 다음과 같습니다.

```
[column name],<data type>,<precision>,<scale>
```

다음 예에서는 구분자로 분리된 플랫폼 파일 제어 파일을 보여 줍니다.

```
Dept,String,10
Name
Place,String,20
Country
Phone number,string,30
```

고정 너비 제어 파일

고정 너비 제어 파일에서 각 행의 형식은 다음과 같습니다.

```
[column name],<data type>,<precision>,<scale>,<number of bytes to process>
```

다음 예에서는 구분자로 분리된 플랫폼 파일 제어 파일을 보여 줍니다.

```
Dept,String,10
Name
Place,String,20
Country
Phone number,string,30
```

고정 너비 소스의 경우 제어 파일에는 처리할 바이트 수를 위한 추가 열이 포함됩니다. 처리할 바이트 수를 지정하지 않을 경우 데이터 통합 서비스는 전체 자릿수에 대해 지정된 값을 처리할 바이트 수로 사용합니다.

런타임 속성 매개 변수화

플랫폼 파일 데이터 개체의 런타임 속성을 사용하면 제어 파일에서 추가 열의 기본 소수 자릿수, 전체 자릿수 및 데이터 유형 값을 매개 변수화하거나 지정할 수 있습니다. 플랫폼 파일 데이터 개체의 런타임 속성에 제어 파일 이름과 제어 파일 디렉터리를 매개 변수화할 수 있습니다. 실제 데이터 개체 속성의 고급 탭에서 매개 변수를 구성합니다. 실제 데이터 개체에서 변환을 작성할 경우에는 매핑 매개 변수를 사용하여 매개 변수 기본값을 재정의할 수 있습니다.

제어 파일의 런타임 처리

매핑을 개발할 때는 플랫폼 파일에서 데이터를 읽는 방법을 결정하는 데이터 개체 읽기 속성을 정의합니다. 데이터 통합 서비스는 제어 파일을 기반으로 하는 플랫폼 파일 데이터 개체의 열을 처리할 수 있습니다.

데이터 통합 서비스는 제어 파일에 기반하여 열 메타데이터를 구성할 때 다음과 같은 데이터 처리 조건을 적용합니다.

사용자가 플랫폼 파일 데이터 개체에 대해 지정하지 않은 열 속성에 대해서는 데이터 통합 서비스가 기본값을 적용합니다.

제어 파일에 런타임 열 속성을 지정하지 않으면 데이터 통합 서비스가 플랫폼 파일 데이터 개체의 기본 런타임 속성을 적용합니다. 데이터 유형, 전체 자릿수 또는 소수 자릿수가 없는 추가 열이 제어 파일에 있으면 데이터 통합 서비스가 기본 데이터 유형, 전체 자릿수 및 소수 자릿수를 사용합니다.

제어 파일에 표시되지 않지만 플랫폼 파일 데이터 개체에는 있는 열에 대해서는 데이터 통합 서비스가 해당 열의 데이터를 NULL로 처리합니다.

플랫폼 파일 데이터 개체의 열이 제어 파일에 없으면 데이터 통합 서비스가 데이터 미리보기 동안 해당 데이터를 NULL로 처리합니다. 런타임에는 열이 제어 파일에 없기 때문에 데이터 통합 서비스가 해당 열을 처리할 수 없습니다.

소스 메타데이터가 제어 파일과 일치하지 않는 경우 데이터 통합 서비스가 매핑에 실패합니다.

제어 파일 기반의 소스 메타데이터가 입력 데이터와 일치하지 않는 경우 데이터 통합 서비스가 데이터를 처리하지 못합니다. 예를 들어 소스의 제어 파일에 있는 데이터 유형이 소스에 있는 데이터 유형과 호환되지 않으면 데이터 통합 서비스가 매핑에 실패합니다.

제어 파일에 대한 규칙 및 지침

제어 파일을 사용할 때 다음 지침을 고려하십시오.

- 제어 파일은 데이터 통합 서비스가 실행되는 시스템이 액세스할 수 있어야 합니다.
- 제어 파일을 플랫폼 파일 데이터 개체로 가져온 후에는 데이터 개체에 있는 열의 데이터 유형, 전체 자릿수 및 소수 자릿수를 변경할 수 없습니다. 제어 파일에서 추가 열의 열 위치와 열 메타데이터는 변경할 수 있습니다.
- 제어 파일에 기반하여 런타임 시 열 이름을 생성할 경우, 제어 파일의 메타데이터가 조회 캐시에 반영됩니다. 매핑을 한 번 실행한 후 추가 열이 있는 동일한 제어 파일을 업데이트할 경우에는 조회 변환의 런타임 속성에서 **조회 소스에서 다시 캐시** 속성을 선택해야 합니다.
- 제어 파일에는 날짜 형식 열을 지정할 수 없습니다. 데이터 통합 서비스는 고급 보기의 플랫폼 파일 데이터 개체 고급 속성에 지정된 플랫폼 파일 날짜 형식을 사용합니다. 플랫폼 파일 소스에 날짜 형식이 여러 개 포함된 경우 데이터 통합 서비스는 고급 보기에 지정된 유일한 날짜 형식만 모든 날짜에 사용합니다.
- 제어 파일에는 이진 데이터 유형을 지정할 수 없습니다. 제어 파일에 **TimestampwithTZ** 데이터 유형이 포함되어 있는 경우, 제어 파일 내에 데이터 유형의 전체 자릿수와 소수 자릿수가 각각 36과 9로 설정되었는지 확인해야 합니다.

Excel에서 플랫폼 파일 데이터 개체로 복사

플랫폼 파일 속성을 Excel에서 구성하고 Developer tool의 플랫폼 파일 데이터 개체로 복사할 수 있습니다. 플랫폼 파일 속성에는 열 이름, 원시 유형, 정밀도 및 배율이 포함됩니다. 열이 많이 포함된 플랫폼 파일 데이터 개체를 개발하거나 편집해야 하는 경우 이렇게 할 수 있습니다.

참고: 플랫폼 파일 데이터 개체에 값을 복사하기 전에 각 셀의 값이 유효한지 반드시 확인해야 합니다. 예를 들어, 문자열 유형은 "0"이 아닌 배율 값을 가질 수 없습니다. 정밀도 값은 단어일 수 없으며 유형 값은 숫자일 수 없습니다. 정보가 올바르지 않으면 오류 메시지가 표시됩니다.

Excel에서 플랫폼 파일 데이터 개체 편집

플랫폼 파일 데이터 개체의 많은 부분을 편집하기 위해 Developer tool에서 모든 값을 변경할 필요가 없습니다. 대신에 플랫폼 파일 열을 Excel에 복사하고, 자동 채우기를 사용하여 모든 값을 동시에 변경한 다음 플랫폼 파일을 Developer tool에 다시 붙여 넣습니다(바깥쪽).

1. Developer tool에서 메타데이터를 복사하려면 플랫폼 파일 데이터 개체 내에서 마우스 오른쪽 단추를 클릭하고 **모두 선택**을 클릭합니다.
2. 메타데이터를 Excel 스프레드시트로 복사합니다.
3. Excel 스프레드시트 내에서 데이터를 변경합니다.

4. Excel에서 메타데이터를 복사합니다.
5. 플랫폼 파일 데이터 개체를 변경 내용으로 업데이트하려면 플랫폼 파일 열 내에서 마우스 오른쪽 단추를 클릭하고 붙여넣기(바꾸기)를 클릭합니다.

메타데이터를 플랫폼 파일 데이터 개체에 복사

Excel에서 메타데이터를 생성한 다음 Developer tool에서 플랫폼 파일 데이터 개체에 복사할 수 있습니다.

1. 플랫폼 파일 데이터 개체를 포함하는 매핑을 생성합니다.
2. Excel에서 플랫폼 파일 데이터 개체에 대한 메타데이터를 정의합니다.
3. Excel에서 메타데이터를 복사합니다.
4. 메타데이터를 플랫폼 파일 데이터 개체로 이동하려면 열 내에서 마우스 오른쪽 단추를 클릭하고 붙여넣기(바꾸기)를 클릭합니다. 다음 이미지는 샘플 Excel 테이블 및 메타데이터를 Developer tool에 복사한 후 생성



된 플랫폼 파일 데이터 개체를 보여줍니다.

메타데이터를 Developer tool에 복사한 후에도 플랫폼 파일 데이터 개체를 계속 변경할 수 있습니다.

예: Excel에서 데이터 개체 편집

플랫폼 파일 데이터 개체를 개발하면서 모든 정밀도 값을 10으로 변경해야 합니다. 각 필드를 개별적으로 변경하는 대신 Excel을 통해 전역 변경을 수행합니다.

메타데이터를 Developer tool에서 Excel로 복사하고, Excel에서 메타데이터를 변경하고, Developer tool의 플랫폼 파일 열로 다시 복사합니다. Excel을 사용하면 각 필드를 개별적으로 변경하지 않아도 됩니다.

다음 이미지는 플랫폼 파일을 Excel로 이동하고 자동 채우기를 사용하여 특정한 값을 변경한 다음 메타데이터를 Developer tool의 플랫폼 파일 데이터 개체에 다시 복사하는 프로세스를 보여줍니다.



플랫 파일 데이터 개체 작성

플랫 파일 데이터 개체는 빈 데이터 개체로 작성하거나, 기존 플랫 파일에서 작성하거나, 제어 파일에서 작성할 수 있습니다. 구분자로 분리된 플랫 파일 데이터 개체 또는 고정 너비 플랫 파일 데이터를 원하는 방법으로 작성할 수 있습니다.

플랫 파일 데이터 개체를 작성할 경우 다음과 같은 방법으로 데이터 개체를 작성할 수 있습니다.

빈 데이터 개체

Developer tool에서 데이터 개체 열과 행을 정의하려면 빈 플랫 파일 데이터 개체를 작성합니다.

기존 플랫 파일에서

데이터 개체에 포함할 메타데이터가 정의되어 있는 플랫 파일이 이미 있는 경우에는 기존 플랫 파일에서 플랫 파일 데이터 개체를 작성합니다.

제어 파일에서

제어 파일에 기반하여 데이터 개체 열과 행을 정의하려면 제어 파일에서 플랫 파일 데이터 개체를 작성합니다.

비어 있는 플랫 파일 데이터 개체 작성

Developer tool에서 데이터 개체 열과 행을 정의하려면 빈 플랫 파일 데이터 개체를 작성합니다.

1. **Object Explorer** 보기에서 프로젝트나 폴더를 선택합니다.
2. **파일 > 새로 만들기 > 데이터 개체**를 클릭하십시오.
3. **실제 데이터 개체 > 플랫 파일 데이터 개체**를 선택하고 **다음**을 클릭합니다.
새 플랫 파일 데이터 개체 대화 상자가 나타납니다.
4. **반 항목으로 작성**을 선택합니다.
5. 데이터 개체의 이름을 입력합니다.
6. 필요에 따라 **찾아보기**를 클릭하여 데이터 개체의 프로젝트 또는 폴더를 선택합니다.
7. **다음**을 클릭합니다.
8. 처리하려는 데이터의 코드 페이지와 일치하는 코드 페이지를 선택합니다.
9. **구분자로 분리됨** 또는 **고정 너비**를 선택합니다.
10. **고정 너비**를 선택한 경우 **마침**을 클릭합니다. **구분자로 분리됨**을 선택한 경우 **다음**을 클릭합니다.
11. 다음과 같은 구분자로 분리된 형식 속성을 구성합니다.

속성	설명
구분자	데이터의 열을 구분하는 데 사용되는 문자입니다. 이스케이프 문자 또는 텍스트 한정자와 동일한 구분자를 입력하면 예기치 않은 결과가 나타날 수 있습니다.
텍스트 한정자	텍스트 문자열의 경계를 정의하는 따옴표 문자입니다. 따옴표 문자를 선택하면 Developer tool이 따옴표 쌍 안의 구분자를 무시합니다.

12. **마침**을 클릭합니다.
편집기에서 빈 데이터 개체가 열립니다. **개요** 보기에서 데이터 개체의 열을 정의합니다.

기존 플랫폼 파일에서 플랫폼 파일 데이터 개체 작성

데이터 개체에 포함할 메타데이터가 정의되어 있는 플랫폼 파일이 이미 있는 경우에는 기존 플랫폼 파일에서 플랫폼 파일 데이터 개체를 작성합니다.

1. **Object Explorer** 보기에서 프로젝트나 폴더를 선택합니다.
2. **파일 > 새로 만들기 > 데이터 개체**를 클릭하십시오.
3. **실제 데이터 개체 > 플랫폼 파일 데이터 개체**를 선택하고 **다음**을 클릭합니다.
새 플랫폼 파일 데이터 개체 대화 상자가 나타납니다.
4. **기존 플랫폼 파일에서 작성**을 선택합니다.
5. **찾아보기**를 클릭하고 파일이 포함된 디렉터리를 찾습니다.
6. **열기**를 클릭합니다.
마법사가 선택한 파일의 이름과 동일하게 데이터 개체의 이름을 지정합니다.
7. 필요에 따라 데이터 개체 이름을 편집합니다.
8. 필요에 따라 **찾아보기**를 클릭하여 데이터 개체의 프로젝트 또는 폴더를 선택합니다.
9. **다음**을 클릭합니다.
10. 처리하려는 데이터의 코드 페이지와 일치하는 코드 페이지를 선택합니다.
11. **구분자로 분리됨** 또는 **고정 너비**를 선택합니다.
12. 필요에 따라 미리 볼 최대 행 수를 편집합니다.
13. **다음**을 클릭합니다.
14. 플랫폼 파일이 구분자로 분리된 플랫폼 파일인지 또는 고정 너비 플랫폼 파일인지에 따라 형식 속성을 구성합니다.
 - 구분자로 분리된 플랫폼 파일의 경우 다음과 같은 속성을 구성합니다.

속성	설명
구분자	데이터의 열을 구분하는 데 사용되는 문자입니다. 이스케이프 문자 또는 텍스트 한정자와 동일한 구분자를 입력하면 예기치 않은 결과가 나타날 수 있습니다.
텍스트 한정자	텍스트 문자열의 경계를 정의하는 따옴표 문자입니다. 따옴표 문자를 선택하는 경우 Developer tool이 따옴표 쌍 안의 구분자를 무시합니다.
첫 줄에서 열 이름 가져오기	선택하면 Developer tool이 첫 번째 행의 데이터를 열 이름에 사용합니다. 열 이름이 첫 번째 행에 표시되는 경우 이 옵션을 선택합니다. Developer tool이 올바르게 읽지 않은 필드 이름에 "FIELD_" 접두사를 추가합니다.
행 구분자	줄 바꿈 문자를 지정합니다. 목록에서 선택하거나 문자를 입력합니다. 역슬래시(\)를 사용하여 8진수 코드의 시작을 나타냅니다. 단일 문자를 사용하려면 해당 문자를 입력합니다. 항목 앞에 백슬래시가 없을 경우 데이터 통합 서비스가 첫 번째 문자만 사용합니다. 문자는 단일 바이트 문자여야 하고 코드 페이지의 다른 문자에 해당 바이트를 포함할 수 없습니다. 기본 값은 줄 바꿈 \012 LF(\n)입니다.
이스케이프 문자	따옴표로 묶이지 않은 문자열에 포함된 열 구분자 바로 앞에 오는 문자 또는 따옴표로 묶인 문자열에서 따옴표 바로 앞에 오는 문자입니다. 이스케이프 문자를 지정하면 데이터 통합 서비스가 구분자를 일반 문자로 읽습니다.

속성	설명
가져오기 시작 행	데이터 통합 서비스가 파일을 가져올 때 읽기 시작하는 행 번호입니다. 예를 들어 두 번째 행에서 시작하도록 지정하는 경우 데이터 통합 서비스가 읽기 전에 Developer tool이 첫 번째 행을 건너뛰니다.
연속 구분자를 하나로 처리	선택하면 데이터 통합 서비스가 하나 이상의 연속 열 구분자를 하나로 읽습니다. 그렇지 않으면 데이터 통합 서비스가 두 개의 연속 구분자를 null 값으로 읽습니다.
데이터에 이스케이프 문자 유지	출력 문자열에 이스케이프 문자를 포함합니다.

- 고정 너비 플랫폼 파일의 경우 다음과 같은 속성을 구성합니다.

속성	설명
첫 줄에서 열 이름 가져오기	선택하면 Developer tool이 첫 번째 행의 데이터를 열 이름에 사용합니다. 열 이름이 첫 번째 행에 표시되는 경우 이 옵션을 선택합니다.
가져오기 시작 행	데이터 통합 서비스가 파일을 가져올 때 읽기 시작하는 행 번호입니다. 예를 들어 두 번째 행에서 시작하도록 지정하는 경우 데이터 통합 서비스가 읽기 전에 Developer tool이 첫 번째 행을 건너뛰니다.
열 나누기	고정 너비 파일에서 열 나누기를 구성합니다. 나누기 편집 을 클릭하여 열 나누기를 편집합니다. 또는 마법사의 지시에 따라 파일 미리보기 창에서 열 나누기를 조작합니다. 열 나누기를 끌어서 이동할 수 있습니다. 또는 열 나누기를 두 번 클릭하여 삭제할 수 있습니다.

15. **다음**을 클릭하여 플랫폼 파일 데이터 개체를 미리 봅니다.

16. **마침**을 클릭합니다.

편집기에서 데이터 개체가 열립니다.

제어 파일에서 플랫폼 파일 데이터 개체 작성

제어 파일에 기반하여 데이터 개체 열과 행을 정의하려면 제어 파일에서 플랫폼 파일 데이터 개체를 작성합니다.

- Object Explorer** 보기에서 프로젝트나 폴더를 선택합니다.
- 파일 > 새로 만들기 > 데이터 개체**를 클릭하십시오.
- 실제 데이터 개체 > 플랫폼 파일 데이터 개체**를 선택하고 **다음**을 클릭합니다.
새 플랫폼 파일 데이터 개체 대화 상자가 나타납니다.
- 제어 파일에서 작성**을 선택합니다.
- 찾아보기**를 클릭하고 제어 파일이 포함된 디렉터리로 이동합니다.
- 열기**를 클릭합니다.
마법사가 선택한 제어 파일의 이름과 동일하게 데이터 개체의 이름을 지정합니다.
- 필요에 따라 데이터 개체 이름을 편집합니다.
- 필요에 따라 **찾아보기**를 클릭하여 데이터 개체의 프로젝트 또는 폴더를 선택합니다.

9. **다음**을 클릭합니다.
10. 처리하려는 데이터의 코드 페이지와 일치하는 코드 페이지를 선택합니다.
11. **구분자로 분리됨** 또는 **고정 너비**를 선택합니다.
12. **고정 너비**를 선택한 경우 **마침**을 클릭합니다. **구분자로 분리됨**을 선택한 경우 **다음**을 클릭합니다.
13. 다음과 같은 구분자로 분리된 형식 속성을 구성합니다.

속성	설명
구분자	데이터의 열을 구분하는 데 사용되는 문자입니다. 이스케이프 문자 또는 텍스트 한정자와 동일한 구분자를 입력하면 예기치 않은 결과가 나타날 수 있습니다.
텍스트 한정자	텍스트 문자열의 경계를 정의하는 따옴표 문자입니다. 따옴표 문자를 선택하면 Developer tool이 따옴표 쌍 안의 구분자를 무시합니다.

14. **마침**을 클릭합니다.
편집기에서 데이터 개체가 열립니다.

제 7 장

데이터의 논리적 보기

이 장에 포함된 항목:

- [데이터의 논리적 보기 개요, 105](#)
- [데이터의 논리적 보기 개발, 106](#)
- [논리적 데이터 개체 모델, 107](#)
- [논리적 데이터 개체 모델 속성, 108](#)
- [논리적 데이터 개체, 114](#)
- [논리적 데이터 개체 매핑, 118](#)

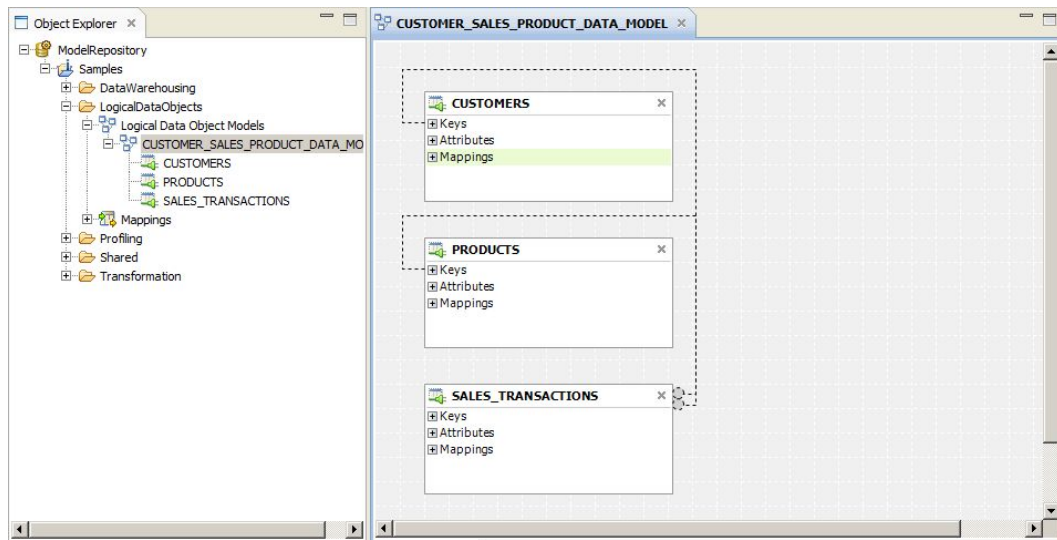
데이터의 논리적 보기 개요

데이터의 논리적 보기는 엔터프라이즈에 상주하는 데이터의 표현입니다. 데이터의 논리적 보기에는 논리적 데이터 모델, 논리적 데이터 개체 및 논리적 데이터 개체 매핑이 포함됩니다.

데이터의 논리적 보기를 통해 다음과 같은 목적을 달성할 수 있습니다.

- 다양한 비즈니스 요구에 맞춰 데이터를 다시 정의할 필요가 없도록 엔터프라이즈 전체에서 공통 데이터 모델을 사용합니다. 또한 데이터 특성이 변경될 경우 이 변경 사항을 한 번 적용하고 한 개의 매핑을 사용하여 해당 변경 사항을 이 데이터를 사용하는 모든 데이터베이스에 적용할 수 있습니다.
- 단일 보기에서 데이터의 관련 소스를 찾아 데이터를 제공합니다. 데이터는 관계형 데이터베이스 및 플랫폼 파일 등 엔터프라이즈의 다양한 위치에 상주합니다. 한 보기에서 모든 데이터 소스에 액세스하고 데이터를 제공할 수 있습니다.
- 논리적 데이터를 관계형 테이블로 노출하여 재사용을 촉진합니다.

다음 그림은 관련 논리적 데이터 개체의 샘플을 보여 줍니다.



논리적 데이터 개체 모델 예제

논리적 데이터 개체 모델을 작성하여 엔터프라이즈에서 논리적 항목의 표현을 설명합니다. 예를 들어 논리적 데이터 개체 모델을 작성하여 서로 다른 소스의 계정 데이터를 단일 보기에서 제공할 수 있습니다.

American Bank가 California Bank를 인수합니다. 인수 후 American Bank는 다음과 같은 목표를 세웁니다.

- 두 은행의 데이터를 비즈니스 인텔리전스 보고서(예: 상위 10명 고객에 관한 보고서)에서 제공합니다.
- 두 은행의 데이터를 중앙의 데이터 웨어하우스로 통합합니다.

일반적으로 American Bank는 개발 환경에서 데이터를 중앙의 데이터 웨어하우스로 통합하고 데이터를 확인한 후 데이터 웨어하우스를 프로덕션 환경으로 이동합니다. 이 프로세스는 몇 개월 이상이 걸릴 수 있습니다. 그 후 프로덕션 환경에서 데이터 웨어하우스에 대해 비즈니스 인텔리전스 보고서를 실행할 수 있습니다.

American Bank의 개발자는 Developer 도구를 사용하여 고객, 계정, 지점 및 엔터프라이즈의 다른 데이터에 대한 모델을 작성할 수 있습니다. 개발자는 American Bank 및 California Bank의 관계형 소스를 고객의 단일 보기로 연결할 수 있습니다. 그런 다음 중앙의 데이터 웨어하우스를 작성하기 전에 비즈니스 인텔리전스 보고서에 데이터를 사용할 수 있도록 만들 수 있습니다.

데이터의 논리적 보기 개발

엔터프라이즈에서 데이터에 액세스하고 데이터를 사용하는 방법을 표현하는 데이터의 논리적 보기를 개발합니다.

데이터의 논리적 보기를 개발한 후 데이터 서비스에 추가하여 최종 사용자에게 가상 데이터를 제공할 수 있습니다.

데이터의 논리적 보기를 개발하기 전에 논리적 데이터 개체 매핑에서 사용할 실제 데이터 개체를 정의할 수 있습니다. 또한 실제 데이터 소스를 프로파일링하여 데이터 품질을 분석할 수 있습니다.

1. 논리적 데이터 모델을 작성하거나 가져옵니다.
2. 필요한 경우 논리적 데이터 개체를 논리적 데이터 개체 모델에 추가하고 개체 간의 관계를 정의합니다.

3. 논리적 데이터 개체에서 데이터를 읽거나 논리적 데이터 개체에 데이터를 쓸 논리적 데이터 개체 매핑을 작성합니다. 논리적 데이터 개체 매핑에 데이터를 변환하는 변환 논리가 포함될 수 있습니다. 변환에는 데이터의 유효성을 검사하고 데이터를 정리하는 데이터 품질 변환이 포함될 수 있습니다.
4. 논리적 데이터 개체 매핑의 출력을 봅니다.

논리적 데이터 개체 모델

논리적 데이터 개체 모델은 엔터프라이즈 내 데이터의 구조와 사용을 설명합니다. 이 모델은 논리적 데이터 개체를 포함하고 이러한 개체 간의 관계를 정의합니다.

논리적 데이터 개체 모델을 정의하여 엔터프라이즈 내 데이터에 대한 통합 모델을 작성합니다. 엔터프라이즈의 데이터는 관계형 데이터베이스 및 플랫폼 파일 등 서로 다른 여러 소스 시스템에 상주할 수 있습니다. 논리적 데이터 개체 모델은 소스 시스템에 관계없이 비즈니스 측면에서 데이터를 표현합니다. 논리적 데이터 개체 모델을 작성하여 데이터를 연구하고 데이터 특성을 설명하고 특성 간 관계를 정의합니다.

예를 들어 **American Bank**의 고객 계정 데이터는 **Oracle** 데이터베이스에 있고 **California Bank**의 고객 계정 데이터는 **IBM DB2** 데이터베이스에 있습니다. 고객 및 계정 간의 관계를 정의하는 고객 계정의 통합 모델을 작성하려고 합니다. 논리적 데이터 개체 모델을 작성하여 관계를 정의합니다.

모델링 도구에서 논리적 데이터 개체 모델을 가져올 수 있습니다. 또한 모델링 도구에서 작성한 **XSD** 파일에서 논리적 데이터 개체 모델을 가져올 수도 있습니다. 또는 **Developer** 도구에서 수동으로 논리적 데이터 개체 모델을 작성할 수 있습니다.

논리적 데이터 개체 모델을 프로젝트 또는 폴더에 추가하고 모델 리포지토리에 저장합니다.

최종 사용자가 논리적 데이터 개체에 대해 **SQL** 쿼리를 실행할 수 있도록 하려면 개체를 **SQL** 데이터 서비스에 포함합니다. 논리적 데이터 개체를 가상 테이블의 소스로 설정합니다. 최종 사용자가 웹을 통해 논리적 데이터 개체에 액세스할 수 있도록 하려면 개체를 웹 서비스에 포함합니다. 논리적 데이터 개체를 작업의 소스로 설정합니다.

논리적 데이터 개체 모델 작성

논리적 데이터 개체 모델을 작성하여 엔터프라이즈 내 데이터의 구조와 사용을 정의합니다. 논리적 데이터 개체 모델을 작성할 때 논리적 데이터 개체를 추가할 수 있습니다. 실제 데이터 개체를 각각의 논리적 데이터 개체와 연결합니다. **Developer** 도구가 모델의 각 논리적 데이터 개체에 대해 논리적 데이터 개체 읽기 매핑을 작성합니다.

1. **Object Explorer** 보기에서 프로젝트 또는 폴더를 선택합니다.
2. **파일 > 새로 만들기 > 논리적 데이터 개체 모델**을 클릭합니다.
새로 만들기 대화 상자가 나타납니다.
3. **논리적 데이터 개체 모델**을 선택하고 **다음**을 클릭합니다.
새 논리적 데이터 개체 모델 대화 상자가 나타납니다.
4. 논리적 데이터 개체 모델의 이름을 입력합니다.
5. 논리적 데이터 개체를 작성하려면 **다음**을 클릭합니다. 빈 논리적 데이터 개체 모델을 작성하려면 **마침**을 클릭합니다.
다음을 클릭하면 **Developer** 도구가 논리적 데이터 개체를 모델에 추가하라는 메시지를 표시합니다.
6. 논리적 데이터 개체를 작성하려면 **새로 만들기** 단추를 클릭합니다.
Developer 도구가 논리적 데이터 개체를 목록에 추가합니다.

7. 이름 옆에 이름을 입력합니다.
8. 필요한 경우 데이터 개체 열에서 열기 단추를 클릭하여 실제 데이터 개체를 논리적 데이터 개체와 연결합니다.
데이터 개체 선택 대화 상자가 나타납니다.
9. 실제 데이터 개체를 선택하고 확인을 클릭합니다.
10. 6 - 9 단계를 반복하여 논리적 데이터 개체를 더 추가합니다.
11. 마침을 클릭합니다.
논리적 데이터 개체 모델이 편집기에서 열립니다.

모델링 도구에서 논리적 데이터 개체 모델 가져오기

모델링 도구 또는 XSD 파일에서 논리적 데이터 개체 모델을 가져올 수 있습니다. 논리적 데이터 개체 모델을 가져와서 구조 및 데이터의 기존 모델을 엔터프라이즈에서 사용합니다.

1. 논리적 데이터 개체 모델을 가져오려는 프로젝트 또는 폴더를 선택합니다.
2. 파일 > 새로 만들기 > 논리적 데이터 개체 모델을 클릭합니다.
새 논리적 데이터 개체 모델 대화 상자가 나타납니다.
3. 데이터 모델의 논리적 데이터 개체 모델을 선택합니다.
4. 다음을 클릭합니다.
5. 모델 유형 필드에서 논리적 데이터 개체 모델을 가져오려는 모델링 도구를 선택합니다.
6. 논리적 데이터 개체 모델의 이름을 입력합니다.
7. 찾아보기를 클릭하고 논리적 데이터 개체 모델의 위치를 선택합니다.
8. 다음을 클릭합니다.
9. 가져오려는 파일을 찾아 선택하고 열기를 클릭합니다.
10. 가져오기 속성을 구성합니다.
11. 다음을 클릭합니다.
12. 논리적 데이터 개체를 논리적 데이터 개체 모델에 추가합니다.
13. 마침을 클릭합니다.
논리적 데이터 개체가 편집기에 표시됩니다.

논리적 데이터 개체 모델 속성

모델링 도구에서 논리적 데이터 개체 모델을 가져오는 경우 모델링 도구와 관련된 속성을 제공합니다.

CA ERwin Data Modeler 가져오기 속성

CA ERwin Data Modeler에서 논리적 데이터 개체 모델을 가져오는 경우 가져오기 속성을 구성합니다.

다음 표에는 CA ERwin Data Modeler에서 모델을 가져올 때 구성하는 속성이 설명되어 있습니다.

속성	설명
UDP 가져오기	<p>사용자 정의 속성을 가져오는 방법을 지정합니다.</p> <p>다음 옵션 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - 메타데이터로. 명시적 값을 속성 값 개체로 가져옵니다. 명시적 값이 내보내지지 않습니다. - 메타데이터로, 기본값 마이그레이션. 명시적 값 및 암시적 값을 속성 값 개체로 가져옵니다. - 설명에서, 기본값 마이그레이션. 암시적 값인 경우에도 속성 이름 및 값을 개체 설명 속성에 추가합니다. - 둘 다, 기본값 마이그레이션. 암시적 값인 경우에도 UDP 값을 메타데이터 및 개체의 설명으로 가져옵니다. <p>기본값은 메타데이터로입니다.</p>
관계 이름 가져오기	<p>ERwin에서 관계 이름을 가져오는 방법을 지정합니다.</p> <p>다음 옵션 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - 원본 관계 이름 - 원본 관계 설명 <p>기본값은 원본 관계 이름입니다.</p>
ID 가져오기	<p>개체의 고유 ID를 NativeId 속성으로 설정할지 여부를 지정합니다.</p>
제목 영역 가져오기	<p>ERwin에서 제목 영역을 가져오는 방법을 지정합니다.</p> <p>다음 옵션 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - 다이어그램으로 가져옴 - 패키지 및 다이어그램으로 가져옴 - 패키지 및 다이어그램으로 가져옴(각 항목마다 하나의 제목 영역이 있다고 가정) - 제목 영역을 가져오지 않음 <p>기본값은 다이어그램으로 가져옴입니다.</p>
열 순서를 가져온 출처	<p>테이블의 열 위치를 가져오는 방법을 지정합니다.</p> <p>다음 옵션 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - 열 순서. ERwin 실제 보기에 표시된 열 순서입니다. - 실제 순서. SQL DDL에서 생성된 데이터베이스의 열 순서입니다. <p>기본값은 실제 순서입니다.</p>
소유자 스키마 가져오기	<p>소유자 스키마를 가져올지 여부를 지정합니다.</p>

IBM Cognos Business Intelligence Reporting - Framework Manager 가져오기 속성

IBM Cognos Business Intelligence Reporting - Framework Manager에서 논리적 데이터 개체 모델을 가져오는 경우 가져오기 속성을 구성합니다.

다음 표에는 IBM Cognos Business Intelligence Reporting - Framework Manager에서 모델을 가져올 때 구성하는 속성이 설명되어 있습니다.

속성	설명
폴더 표현	Framework Manager의 폴더를 표현할 방법을 지정합니다. 다음 옵션 중 하나를 선택합니다. - 무시. 폴더를 무시합니다. - 플랫. 폴더를 다이어그램으로 나타내지만 계층 구조를 유지하지 않습니다. - 계층. 폴더를 다이어그램으로 나타내고 계층 구조를 유지합니다. 기본값은 무시입니다.
패키지 표현	Cognos Framework Manager의 패키지를 표현할 방법을 지정합니다. 다음 옵션 중 하나를 선택합니다. - 무시. 제목 영역을 무시합니다. - 제목 영역. 패키지를 제목 영역으로 표현합니다. - 모델. 패키지를 모델로 표현합니다. 기본값은 무시입니다.
리버스 엔지니어 관계	Developer 도구가 두 dbQueries 간의 관계를 참조 무결성 제약 조건으로 계산할지 여부를 지정합니다.
테이블 디자인 수준	가져오는 테이블의 디자인 수준을 제어할 방법을 지정합니다. 다음 옵션 중 하나를 선택합니다. - 논리 및 실제. 테이블이 모델의 논리 보기와 실제 보기에 표시됩니다. - 실제. 테이블이 모델의 실제 보기에만 표시됩니다. 기본값은 실제입니다.
사용 속성 무시	쿼리 항목의 사용 속성을 사용할지 여부를 지정합니다.

SAP BusinessObjects Designer 가져오기 속성

SAP BusinessObjects Designer에서 논리적 데이터 개체 모델을 가져오는 경우 가져오기 속성을 구성합니다.

다음 표에는 SAP BusinessObjects Designer에서 모델을 가져올 때 구성하는 속성이 설명되어 있습니다.

속성	설명
시스템	BusinessObjects 리포지토리의 이름입니다. BusinessObjects 버전 11.x 및 12.x(XI)의 경우 Central Management Server의 이름을 입력합니다. BusinessObjects 버전 5.x 및 6.x의 경우 Supervisor 응용 프로그램에서 정의한 리포지토리의 이름을 입력합니다.
인증 모드	로그인 인증 모드입니다. 이 매개 변수는 SAP BusinessObjects Designer 11.0 이상에 적용할 수 있습니다. 다음 인증 모드 중 하나를 선택합니다. - Enterprise. Business Objects Enterprise 로그인 - LDAP. LDAP 서버 인증 - Windows AD. Windows Active Directory 서버 인증 - Windows NT. Windows NT 도메인 서버 인증 - 독립 실행형. 독립 실행형 인증 기본값은 엔터프라이즈입니다.
사용자 이름	BusinessObjects 서버의 사용자 이름입니다. 버전 11.x 및 12.x(XI)의 경우 BusinessObjects 그룹의 멤버여야 합니다.
암호	BusinessObjects 서버의 암호입니다.
자동 실행	대화형 모드 또는 자동 모드 중 어떤 모드에서 실행할지 지정합니다. 기본값은 자동입니다.
실행 후 닫기	Developer 도구가 모델 가져오기를 완료한 후 BusinessObjects를 닫을지 여부를 지정합니다.
테이블 디자인 수준	가져온 테이블의 디자인 수준을 지정합니다. 다음 옵션 중 하나를 선택합니다. - 논리 및 실제. 테이블이 모델의 논리 보기와 실제 보기에 표시됩니다. - 실제. 테이블이 모델의 실제 보기에 표시됩니다. 기본값은 실제입니다.
조인을 외래 키로 변환	모델의 단순 SQL 조인을 외래 키 관계로 변환합니다. 데이터베이스 디자인 도구와 같이 구조적 관계형 메타데이터만 지원하는 도구에 모델을 내보내려는 경우 매개 변수를 선택합니다.
클래스 표현	클래스 및 하위 클래스의 트리 구조를 가져오는 방법을 지정합니다. Developer 도구는 각 클래스를 CWM OLAP 표준에서 정의하는 차원으로 가져옵니다. 또한 Developer 도구는 클래스 및 하위 클래스를 CWM 및 UML 표준에서 정의하는 패키지 트리로 가져옵니다. 다음 옵션 중 하나를 선택합니다. - 플랫 구조로. Developer 도구는 패키지를 작성하지 않습니다. - 단순화된 트리 구조로. Developer 도구가 각 클래스와 하위 클래스에 대해 패키지를 작성합니다. - 전체 트리 구조로. Developer 도구가 각 클래스에 대해 패키지를 작성합니다. 기본값은 플랫 구조로입니다.

속성	설명
값 목록 포함	Developer 도구가 개체와 관련된 값의 목록을 가져오는 방식을 제어합니다.
차원 속성 변환	<p>차원 이름, 설명 및 역할을 기본 테이블에 전송하고 특성 이름, 설명 및 데이터 유형을 기본 열로 전송하는 방법을 지정합니다.</p> <p>다음 옵션 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - 비활성화됨. 속성 전송이 수행되지 않습니다. - 활성화됨. 차원 개체 및 관계형 개체 간에 직접 일치가 있을 경우 속성 전송이 수행됩니다. Developer 도구가 차원 이름을 관계형 이름으로 마이그레이션합니다. - 활성화됨(이름 유지). 차원 개체 및 관계형 개체 간에 직접 일치가 있을 경우 속성 전송이 수행됩니다. Developer 도구가 관계형 이름을 유지합니다. <p>기본값이 비활성화됩니다.</p>

SAP PowerDesigner CDM 가져오기 속성

SAP PowerDesigner CDM에서 논리적 데이터 개체 모델을 가져오는 경우 가져오기 속성을 구성합니다.

다음 테이블에는 SAP PowerDesigner CDM에서 모델을 가져올 때 구성하는 속성이 설명되어 있습니다.

속성	설명
UDP 가져오기	<p>사용자 정의 속성을 가져오는 방법을 지정합니다.</p> <p>다음 옵션 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - 메타데이터로. 명시적 값을 속성 값 개체로 가져옵니다. 명시적 값이 내보내지지 않습니다. - 메타데이터로, 기본값 마이그레이션. 명시적 값 및 암시적 값을 속성 값 개체로 가져옵니다. - 설명에서, 기본값 마이그레이션. 암시적 값인 경우에도 속성 이름 및 값을 개체 설명 속성에 추가합니다. - 둘 다, 기본값 마이그레이션. 암시적 값인 경우에도 UDP 값을 메타데이터 및 개체의 설명으로 가져옵니다. <p>기본값은 메타데이터로입니다.</p>
연관 클래스 가져오기	Developer tool이 연관 클래스를 가져올지 여부를 지정합니다.
ID 가져오기	개체의 고유 ID를 NativeId 속성으로 설정할지 여부를 지정합니다.
볼륨 정보를 설명 필드에 추가	발생 횟수 정보를 가져오고 설명 속성에 추가합니다.
텍스트 서식 제거	<p>서식 있는 텍스트를 제거할지 또는 유지할지 여부를 지정합니다.</p> <p>모델이 PowerDesigner 7.0 또는 7.5에서 생성된 경우 이 옵션을 선택합니다.</p> <p>모델이 PowerDesigner 8.0 이상에서 생성된 경우 이 옵션을 선택 취소합니다.</p>

SAP PowerDesigner PDM 가져오기 속성

SAP PowerDesigner PDM에서 논리적 데이터 개체 모델을 가져올 때 가져오기 속성을 구성합니다.

다음 테이블에는 SAP PowerDesigner PDM에서 모델을 가져올 때 구성하는 속성이 설명되어 있습니다.

속성	설명
UDP 가져오기	<p>사용자 정의 속성을 가져오는 방법을 지정합니다.</p> <p>다음 옵션 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - 메타데이터로. 명시적 값을 속성 값 개체로 가져옵니다. 명시적 값이 내보내지지 않습니다. - 메타데이터로, 기본값 마이그레이션. 명시적 값 및 암시적 값을 속성 값 개체로 가져옵니다. - 설명에서, 기본값 마이그레이션. 암시적 값인 경우에도 속성 이름 및 값을 개체 설명 속성에 추가합니다. - 둘 다, 기본값 마이그레이션. 암시적 값인 경우에도 UDP 값을 메타데이터 및 개체의 설명으로 가져옵니다. <p>기본값은 메타데이터로입니다.</p>
ID 가져오기	개체의 고유 ID를 NativeId 속성으로 설정할지 여부를 지정합니다.
볼륨 정보를 설명 필드에 추가	발생 횟수 정보를 가져오고 설명 속성에 추가합니다.
텍스트 서식 제거	<p>서식 있는 텍스트를 제거할지 또는 유지할지 여부를 지정합니다.</p> <p>모델이 PowerDesigner 7.0 또는 7.5에서 생성된 경우 이 옵션을 선택합니다.</p> <p>모델이 PowerDesigner 8.0 이상에서 생성된 경우 이 옵션을 선택 취소합니다.</p>

XSD 가져오기 속성

모델링 도구에서 내보낸 XSD 파일의 논리적 데이터 개체 모델을 가져올 수 있습니다.

다음 표에는 XSD 파일에서 모델을 가져올 때 구성하는 속성이 설명되어 있습니다.

속성	설명
요소 콘텐츠 이름	<p>XSD 파일의 텍스트 콘텐츠(예: #PCDATA)를 보유하는 특성입니다.</p> <p>기본값은 메타데이터로입니다.</p>
축소 수준	<p>클래스를 축소할 때를 지정합니다. 선택하는 값에 따라 Developer 도구가 XSD 파일의 요소 및 특성을 모두 또는 일부 가져올지 여부가 결정됩니다.</p> <p>다음 옵션 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - 없음. 모든 XSD 요소는 클래스가 되고 모든 XSD 특성은 특성이 됩니다. - 비어 있음. 빈 클래스만 상위 클래스로 축소됩니다. - 단일 특성. 단일 특성이 있고 하위 요소가 없는 XSD 요소만 상위 클래스로 축소됩니다. - 하위 요소 없음. 하위 요소가 없는 모든 XSD 요소가 상위 클래스로 축소됩니다. - 모두. 모든 축소 가능한 XSD 요소가 상위 클래스로 축소됩니다. <p>기본값은 모두입니다.</p>
축소 별	Developer 도구가 들어오는 xlink와 함께 XML 요소를 상위 클래스로 축소할지 여부를 지정합니다.

속성	설명
클래스 유형	Developer 도구가 요소가 상위 요소로 축소되는 클래스 유형을 작성할지 여부를 지정합니다.
임의	'xs:any' pseudo 요소에 대해 클래스 또는 항목을 작성할지 여부를 지정합니다.
ID 생성	기본 및 외래 키를 작성하기 위해 추가 특성을 작성할지 여부를 지정합니다. 기본적으로 Developer 도구는 추가 특성을 생성하지 않습니다.
다음 방식으로 대체 그룹 가져오기	상속을 표현할 방법을 지정합니다. 다음 옵션 중 하나를 선택합니다. - 일반화. 상속을 일반화로 표현합니다. - 롤 다운. 상속된 특성을 하위 클래스에 복제합니다. 기본값은 롤 다운입니다.
경로 포함	포함된 스키마 파일(있을 경우)이 있는 디렉터리의 경로입니다.
UDP 네임스페이스	사용자 정의 속성으로 가져올 특성이 있는 네임스페이스입니다.

논리적 데이터 개체

논리적 데이터 개체는 엔터프라이즈에서 논리적 항목을 설명하는 논리적 데이터 개체 모델의 개체입니다. 특성과 키를 포함하며 특성 간의 관계를 설명합니다.

데이터 개체 모델에서 서로 관련되는 논리적 데이터 개체를 포함합니다. 예를 들어 논리적 데이터 개체 **Customer**와 **Account**가 국립 은행의 논리적 데이터 개체 모델에 나타납니다. 논리적 데이터 개체 모델은 고객 및 계정 간의 관계를 설명합니다.

모델에서 논리적 데이터 개체 **Account**에는 **Account_Number** 특성이 포함됩니다. **Account_Number**는 계정을 고유하게 식별하기 때문에 기본 키입니다. **Customer** 데이터 개체가 각 고객에 대해 해당 계정을 참조해야 하므로 **Account**는 논리적 데이터 개체 **Customer**와 관계를 가집니다.

실제 데이터 개체를 논리적 데이터 개체 모델 편집기로 끌어서 논리적 데이터 개체를 작성할 수 있습니다. 또는 논리적 데이터 개체를 작성하고 특성 및 키를 정의할 수 있습니다.

논리적 데이터 개체 속성

논리적 데이터 개체에는 데이터 개체 및 해당 개체와 논리적 데이터 개체 모델의 다른 논리적 데이터 개체 간 관계를 정의하는 속성이 포함됩니다.

논리적 데이터 개체 속성은 편집기의 편집기 탭에 있습니다. 다음 그림은 논리적 데이터 개체 편집기를 보여 줍니다.

Overview

General

Name: customer
Description:
Read Mapping: customer_Read_Mapping Add... Delete
Write Mapping: Add... Delete

customer

Name	Type
CUSTID	decimal
Status	string
LastName	string
FirstName	string
CUSTID1	decimal

Attributes

	Name	Type	Primary...	Precision	Scale	Nullable	Lower	Upper	Queryable	Description
1	CUSTID	decimal	✓	2	0		1	1		
2	Status	string		8	0	✓	1	1		
3	LastName	string		15	0	✓	1	1		
4	FirstName	string		9	0	✓	1	1		
5	CUSTID1	decimal	✓	2	0		1	1		

Overview
Keys
Relationships
Access
Read Mapping
Advanced

다음 테이블에는 논리적 데이터 개체 편집기에 표시되는 정보가 설명되어 있습니다.

탭 이름	설명
개요	일반 영역에는 논리적 데이터 개체의 개체 이름, 설명, 읽기 및 쓰기 매핑(해당하는 경우)이 포함됩니다. 특성 영역에는 논리적 데이터 개체의 데이터 구조가 표시됩니다.
키	논리적 데이터 개체에서 하나 이상의 특성이 기본 키 또는 고유 키가 될 수 있습니다.
관계	논리적 데이터 개체 간의 연관입니다.
액세스	논리적 데이터 개체 및 해당 데이터 개체의 각 특성에 대한 액세스 유형입니다.
읽기 매핑	논리적 데이터 개체에 연결된 논리적 데이터 개체 읽기 매핑입니다. 논리적 데이터 개체에 읽기 매핑이 포함되어 있지 않은 경우 읽기 매핑 탭이 표시되지 않습니다.
쓰기 매핑	논리적 데이터 개체에 연결된 논리적 데이터 개체 쓰기 매핑입니다. 논리적 데이터 개체에 쓰기 매핑이 포함되어 있지 않은 경우 쓰기 매핑 탭이 표시되지 않습니다.

특성 관계

관계는 하나 이상의 논리적 데이터 개체의 기본 또는 외래 키 특성 간의 연관입니다.

특성 간에 다음과 같은 유형의 관계를 정의할 수 있습니다.

식별

한 특성이 다른 특성과의 연관을 통해 식별되는 경우 두 특성 간의 관계입니다.

예를 들어 논리적 데이터 개체 **Branch**의 **Branch_ID** 특성과 논리적 데이터 개체 **Customer**의 **Branch_Location** 특성 간의 관계가 식별될 수 있습니다. 지점 ID가 지점 위치에 대해 고유하기 때문입니다.

식별되지 않음

한 특성을 다른 특성에 대해 독립적으로 식별하는 두 특성 간의 관계입니다.

예를 들어 **Account** 논리적 데이터 개체의 **Account_Type** 특성과 **Customer** 논리적 데이터 개체의 **Account_Number** 특성 간의 관계가 식별되지 않음일 수 있습니다. 계정 유형을 계정 번호와 연관시키지 않고도 식별할 수 있기 때문입니다.

관계를 정의하면 논리적 데이터 개체 모델은 특성 간에 실선으로 식별 관계를 나타냅니다. 또한 특성 간에 점선으로 식별되지 않음 관계를 나타냅니다.

논리적 데이터 개체 작성

논리적 데이터 개체 모델에서 논리적 데이터 개체를 작성하여 엔터프라이즈의 논리적 항목을 정의할 수 있습니다.

1. **파일 > 새로 만들기 > 논리적 데이터 개체**를 클릭합니다.
2. 논리적 데이터 개체 이름을 입력합니다.
3. 논리적 데이터 개체에 대한 논리적 데이터 개체 모델을 선택하고 **마침**을 클릭합니다.
논리적 데이터 개체가 논리적 데이터 개체 모델 편집기에 나타납니다.
4. 논리적 데이터 개체를 선택하고 **속성** 보기를 클릭합니다.
5. 일반 영역의 **개요** 탭에서 필요한 경우 논리적 데이터 개체 이름 및 설명을 편집할 수 있습니다.
6. 특성 영역의 **개요** 탭에서 특성을 작성하고 해당 데이터 유형 및 전체 자릿수를 지정할 수 있습니다.

Overview

General

Name:

Description:

Read Mapping:

Write Mapping:

customer

Name	Type
CUSTID	decimal
Status	string
LastName	string
FirstName	string

Attributes

	Name	Type	Primary...	Precision	Scale	Nullable	Lower	Upper	Querya...	Description	>>
1	CUSTID	decimal	<input type="checkbox"/>	2	0	<input checked="" type="checkbox"/>	1	1	<input type="checkbox"/>		
2	Status	string	<input type="checkbox"/>	8	0	<input checked="" type="checkbox"/>	1	1	<input type="checkbox"/>		
3	LastName	string	<input type="checkbox"/>	15	0	<input checked="" type="checkbox"/>	1	1	<input type="checkbox"/>		
4	FirstName	string	<input type="checkbox"/>	9	0	<input checked="" type="checkbox"/>	1	1	<input type="checkbox"/>		

Overview | Keys | Relationships | Access | Read Mapping | Advanced

7. 키 탭에서 필요한 경우 데이터 개체의 기본 키와 고유 키를 지정합니다.

Keys

Keys: Key

Name: Key

Description:

Available Attributes:

- Status
- LastName
- FirstName

Selected Attributes:

- CUSTID

Buttons: Add..., Remove

Navigation: Overview | Keys | Relationships | Access | Read Mapping | Advanced

8. 관계 탭에서 필요한 경우 논리적 데이터 개체 간의 관계를 작성합니다.

Relationships

Relationships: Relationship

Name: Relationship

Description:

Relationship Type:

- ☒ Identifying
- ☐ Non-Identifying

Referenced Key:

Key: [customer.Key](#)

Attributes:

- CUSTID

Logical Data Object:

Available Attributes:

- CUSTID
- Status
- LastName
- FirstName

Selected Attributes:

- CUSTID1

Buttons: Add..., Remove

Navigation: Overview | Keys | Relationships | Access | Read Mapping | Advanced

9. 액세스 탭에서 필요한 경우 논리적 데이터 개체의 액세스 유형과 데이터 개체의 각 특성을 편집합니다.

기본값은 읽기 전용입니다.

Access

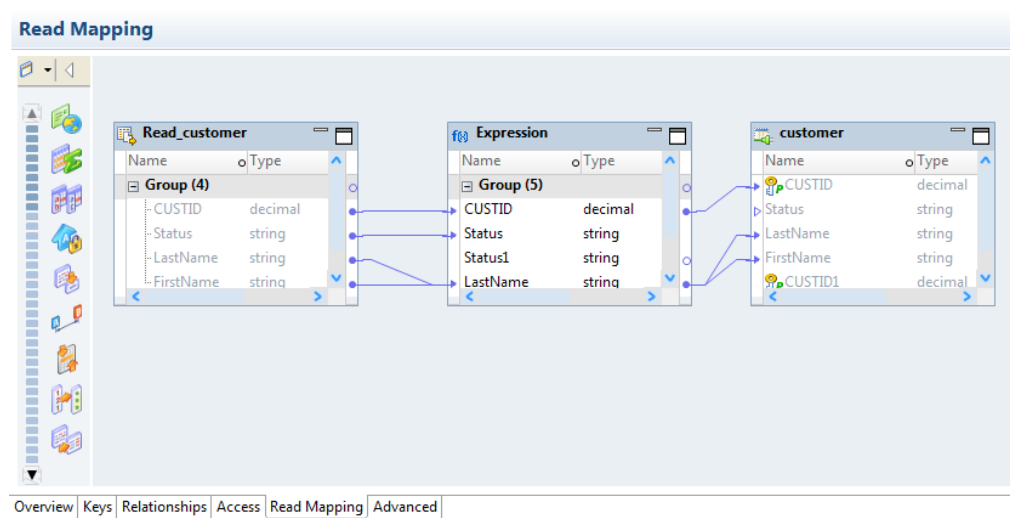
Write Access: Read Only

Attributes:

	Name	Type	Write Access
1	CUSTID	decimal	Read Only
2	Status	string	Read Only
3	LastName	string	Read Only
4	FirstName	string	Read Only
5	CUSTID1	decimal	Read Only

Overview
Keys
Relationships
Access
Read Mapping
Advanced

10. 읽기 매핑 탭에서 필요한 경우 논리적 데이터 개체 읽기 매핑을 작성합니다.



11. 쓰기 매핑 탭에서 필요한 경우 논리적 데이터 개체 쓰기 매핑을 작성합니다.

12. 논리적 데이터 개체를 저장합니다.

논리적 데이터 개체 매핑

논리적 데이터 개체 매핑은 논리적 데이터 개체를 하나 이상의 실제 데이터 개체와 연결하는 매핑입니다. 변환 논리를 포함할 수 있습니다.

논리적 데이터 개체 매핑은 다음 유형 중 하나일 수 있습니다.

- 읽기
- 쓰기

각 논리적 데이터 개체를 하나의 논리적 데이터 개체 읽기 매핑 또는 하나의 논리적 데이터 개체 쓰기 매핑에 연결할 수 있습니다.

논리적 데이터 개체 읽기 매핑

논리적 데이터 개체 읽기 매핑은 입력으로 하나 이상의 실제 데이터 개체와 출력으로 하나의 논리적 데이터 개체를 포함합니다. 매핑에 데이터를 변환하는 매핑 논리가 포함될 수 있습니다.

매핑 논리는 기본 데이터 소스에 액세스하지 않고 데이터에 액세스할 수 있도록 합니다. 또한 둘 이상의 소스에서 전송되는 데이터에 대한 단일 보기를 사용할 수 있도록 합니다.

예를 들어 **American Bank**에 고객 계정에 대한 논리적 데이터 개체 모델이 있다고 가정합니다. 논리적 데이터 개체 모델에 **Customers** 논리적 데이터 개체가 포함됩니다.

American Bank는 **Customers** 논리적 데이터 개체의 두 관계형 데이터베이스에서 고객 데이터를 보기를 원합니다. 논리적 데이터 개체 읽기 매핑을 사용하여 이 작업을 수행하고 **데이터 뷰어** 보기에서 출력을 볼 수 있습니다.

논리적 데이터 개체 쓰기 매핑

논리적 데이터 개체 쓰기 매핑은 논리적 데이터 개체를 입력으로 포함합니다. 이 매핑은 논리적 데이터 개체에서 대상으로 쓸 수 있도록 합니다.

매핑에 데이터를 변환하는 매핑 논리가 포함될 수 있습니다. 매핑은 기본 데이터 대상에 액세스하지 않고 실행됩니다. 매핑은 변환된 데이터를 대상에 쓰지 않고 이러한 데이터의 단일 보기를 제공합니다.

논리적 데이터 개체 매핑 작성

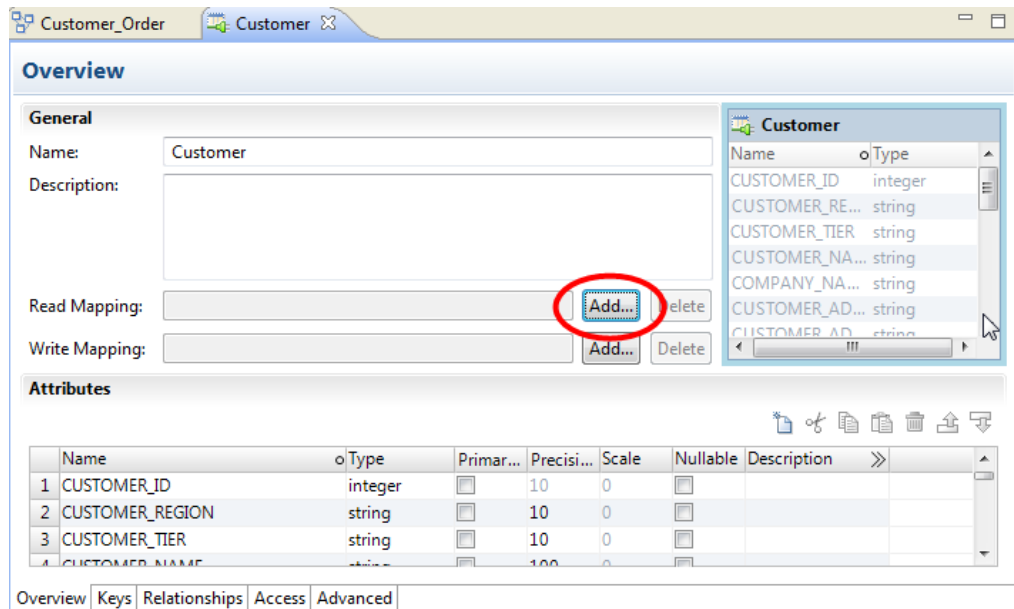
논리적 데이터 개체 매핑을 작성하여 실제 데이터 개체에서 논리적 데이터 개체에 연결하고 데이터를 변환할 수 있습니다.

1. **Object Explorer** 보기에서 매핑을 추가할 논리적 데이터 개체를 두 번 클릭합니다.

논리적 데이터 개체 편집기가 열립니다.

2. 일반 영역의 **개요** 탭에서 **추가**를 클릭하여 읽기 매핑 또는 쓰기 매핑을 추가합니다.

다음 이미지는 **추가** 단추를 보여 줍니다.



3. 매핑에 대한 이름을 입력한 다음 **마침**을 클릭합니다.

매핑이 읽기 또는 쓰기 매핑 중 어느 것인지에 따라 논리적 데이터 개체가 매핑 입력 또는 출력으로 편집기에 표시됩니다.

4. 실제 데이터 개체의 데이터를 논리적 데이터 개체에 연결합니다.

- a. 매핑을 편집하려면 **읽기 매핑** 또는 **쓰기 매핑**을 클릭합니다.

매핑 편집기 색상표가 열립니다.

- b. **Object Explorer** 보기에서 연결하려는 실제 데이터 개체를 찾아 매핑 편집기 색상표로 끌어옵니다.

매핑에 추가 대화 상자가 열립니다.

- c. 읽기 매핑, 쓰기 매핑 또는 조회 매핑을 작성하도록 선택합니다.

- d. 읽기 매핑을 작성하도록 선택한 경우 읽기 매핑의 개체 액세스를 관련 데이터 개체 또는 독립 데이터 개체로 지정합니다.

- e. **확인**을 클릭합니다.

사용자가 **고객** 플랫폼 파일 데이터 개체를 매핑 편집기 색상표로 끌고 읽기 매핑으로 지정하면 다음 이미지가 결과를 표시합니다.



5. 필요한 경우 재사용 가능 개체를 매핑에 추가합니다.

논리적 데이터 개체 또는 다른 모델 리포지토리 개체를 추가할 수 있습니다.

- a. 매핑 편집기를 마우스 오른쪽 단추로 클릭하고 **재사용 가능 개체 추가**를 선택합니다.

- b. 모델 리포지토리의 개체 중 하나를 선택하고 **확인**을 클릭합니다.

- c. 재사용 가능 개체를 읽기 매핑, 쓰기 매핑 또는 조회 매핑으로 지정하도록 선택합니다.

- d. 재사용 가능 개체를 읽기 매핑으로 지정하도록 선택한 경우 읽기 매핑의 개체 액세스를 관련 데이터 개체 또는 독립 데이터 개체로 지정합니다.

- e. **확인**을 클릭합니다.

6. 필요한 경우 추가 개체 및 변환을 매핑에 추가하고 매핑 개체 간의 링크를 작성한 다음 **데이터 뷰어** 보기를 클릭하고 매핑을 실행합니다.

결과가 **출력** 섹션에 표시됩니다.

제 8 장

데이터 보기

이 장에 포함된 항목:

- [데이터 보기 개요, 121](#)
- [구성, 122](#)
- [데이터 내보내기, 128](#)
- [개체 종속성, 129](#)
- [로그, 130](#)
- [유효성 검사 기본 설정, 131](#)
- [Developer tool에서 작업 모니터링, 132](#)

데이터 보기 개요

매핑을 실행하거나, 프로필 결과를 보거나, 소스 데이터를 보거나, 변환의 데이터를 미리 보거나, SQL 쿼리를 실행하거나, 웹 서비스 메시지를 미리 보거나, 개체에 대한 종속성을 볼 수 있습니다.

다음 보기를 사용하여 데이터 및 매핑을 조작하고 봅니다.

매핑 편집기 보기

매핑 편집기를 사용하여 매핑을 실행하고 프로필 결과를 봅니다. 소스에서 대상으로 출력을 이동하고 데이터를 변환하는 매핑을 실행합니다. 명령줄 또는 **실행** 대화 상자에서 매핑을 실행할 수 있습니다.

데이터 뷰어 보기

데이터 뷰어 보기를 사용하여 소스 데이터를 보고, 변환을 위한 데이터를 미리 보고, SQL 쿼리를 실행하거나, 웹 서비스 메시지를 미리 봅니다.

도메인에 둘 이상의 데이터 통합 서비스가 포함되는 경우 데이터를 보려면 기본 데이터 통합 서비스를 선택해야 합니다. 또한 데이터를 볼 때 사용할 다른 데이터 통합 서비스를 추가할 수 있습니다. 데이터를 볼 때 **Developer tool**이 적용하는 설정을 제어하는 구성을 작성할 수 있습니다.

데이터 통합 서비스 또는 **Spark** 엔진에서 실행되는 매핑 내의 데이터를 미리 볼 수 있습니다. 소스 및 변환을 매핑의 미리 보기 지점으로 선택할 수 있습니다.

데이터 통합 서비스는 미리 보기 지점을 기준으로 원시 환경에서 또는 **Spark** 엔진에서 작업을 실행할지 결정합니다. 미리 보기 지점은 데이터를 보기 위해 선택한 매핑의 개체입니다. **Spark** 엔진에서 데이터 미리 보기에 대한 자세한 내용은 *Data Engineering Integration 사용자 가이드*를 참조하십시오.

참고: 데이터 뷰어에서 미리 볼 수 있는 최대 행 수는 100,000개입니다.

데이터 뷰어 보기에서 데이터를 보는 경우 데이터를 파일로 내보낼 수 있습니다. 로그 이벤트를 보여 주는 로그에 액세스할 수도 있습니다.

개체 종속성 보기

개체 종속성 보기를 사용하여 모델 리포지토리 개체를 보거나 수정하거나 삭제할 때 개체 종속성을 봅니다.

구성

구성은 사용자가 매핑을 실행하거나, 데이터를 미리 보거나, SQL 쿼리를 실행하거나, 웹 서비스 메시지를 미리 볼 때 **Developer** 도구가 적용하는 설정 그룹입니다.

구성은 기본 데이터 통합 서비스, 소스에서 읽을 행의 수, 기본 날짜/시간 형식 및 최적화 수준 같은 설정을 제어합니다. 사용자가 작성하는 구성은 **Developer** 도구의 설치에 적용됩니다.

다음과 같은 구성을 작성할 수 있습니다.

- 데이터 뷰어 구성. **데이터 뷰어** 보기에서 출력을 미리 볼 때 **Developer** 도구가 적용하는 설정을 제어합니다.
- 매핑 구성. **구성 실행** 대화 상자 또는 명령줄을 통해 매핑을 실행할 때 **Developer** 도구가 적용하는 설정을 제어합니다.
- 웹 서비스 구성. **데이터 뷰어** 보기에서 웹 서비스의 출력을 미리 볼 때 **Developer** 도구가 적용하는 설정을 제어합니다.

구성 속성

사용자가 출력을 미리 보거나 매핑을 실행할 때 **Developer** 도구는 구성 속성을 적용합니다. **실행** 대화 상자에서 **데이터 뷰어** 보기 또는 매핑의 구성 속성을 설정합니다.

데이터 통합 서비스 속성

Developer tool은 데이터 뷰어, 매핑 및 웹 서비스 구성에 대한 데이터 통합 서비스 탭을 표시합니다.

다음 표에는 데이터 통합 서비스에 대해 구성하는 속성이 나와 있습니다.

속성	설명
기본 데이터 통합 서비스 사용	기본 데이터 통합 서비스를 사용하여 매핑을 실행합니다. 기본값은 활성화됨입니다.
데이터 통합 서비스	기본 데이터 통합 서비스를 사용하지 않을 경우 매핑을 실행하는 데이터 통합 서비스를 지정합니다.
사용 가능한 운영 체제 프로파일	데이터 통합 서비스가 운영 체제 프로파일을 사용하도록 활성화되었을 때 매핑을 실행할 운영 체제 프로파일을 지정합니다. Developer tool 은 관리자가 사용자에게 하나 이상의 운영 체제 프로파일을 할당한 경우에만 이 속성을 표시합니다. 데이터 통합 서비스는 사용자에게 할당된 기본 운영 체제 프로파일로 매핑을 실행합니다. 사용 가능한 운영 체제 프로파일 목록에서 운영 체제 프로파일을 변경할 수 있습니다.

소스 속성

Developer tool은 데이터 뷰어, 매핑 및 웹 서비스 구성에 대한 **소스** 탭을 표시합니다.

다음 표에는 소스에 대해 구성하는 속성이 나와 있습니다.

속성	설명
모든 행 읽기	소스의 모든 행을 읽습니다. 기본값은 활성화됨입니다.
최대 행 수 읽기	모든 행을 읽지 않을 경우 소스에서 읽을 최대 행 수를 지정합니다. 참고: 사용자 지정된 데이터 개체에 쓰는 매핑에 대해 이 옵션을 활성화하면 데이터 통합 서비스가 대상에 쓰기 전에 대상 테이블을 자르지 않습니다. 기본값은 1000입니다.
모든 문자 읽기	열의 모든 문자를 읽습니다. 기본값은 비활성화됨입니다.
최대 문자 수 읽기	모든 문자를 읽지 않을 경우 각 열에서 읽을 최대 문자 수를 지정합니다. 데이터 통합 서비스는 SAP 소스에 대해 이 속성을 무시합니다. 기본값은 4000입니다.

결과 속성

Developer 도구는 데이터 뷰어 및 웹 서비스 구성에 대한 **결과** 탭을 표시합니다.

다음 표에는 **데이터 뷰어** 보기의 결과에 대해 구성하는 속성이 나와 있습니다.

속성	설명
모든 행 표시	데이터 뷰어 보기에서 모든 행을 표시합니다. 기본값이 비활성화됩니다.
최대 행 수 표시	모든 행을 표시하지 않을 경우 표시할 최대 행 수를 지정합니다. 기본값은 1000입니다.
모든 문자 표시	열의 모든 문자를 표시합니다. 기본값이 비활성화됩니다.
최대 문자 수 표시	모든 문자를 표시하지 않을 경우 각 열에서 표시할 최대 문자 수를 지정합니다. 기본값은 4000입니다.

메시지 속성

Developer 도구는 웹 서비스 구성에 대한 **메시지** 탭을 표시합니다.

다음 표에는 메시지에 대해 구성하는 속성이 나와 있습니다.

속성	설명
요청 메시지에 대해 최대 문자 수 읽기	입력 메시지에서 처리할 최대 문자 수를 지정합니다.
응답 메시지에 대해 최대 문자 수 표시	출력 메시지에서 표시할 최대 문자 수를 지정합니다.

고급 속성

Developer 도구는 데이터 뷰어, 매핑 및 웹 서비스 구성에 대한 **고급** 탭을 표시합니다.

다음 표에는 고급 속성이 나와 있습니다.

속성	설명
기본 날짜/시간 형식	매핑이 문자열을 날짜로 변환할 때 데이터 통합 서비스에서 사용하는 날짜/시간 형식 기본값은 MM/DD/YYYY HH24:MI:SS입니다.
추적 수준 재정의	<p>매핑에서 각 변환에 사용되는 추적 수준을 재정의합니다. 추적 수준은 데이터 통합 서비스가 매핑 로그 파일에 보내는 정보의 양을 결정합니다.</p> <p>다음 추적 수준 중 하나를 선택합니다.</p> <ul style="list-style-type: none">- 없음. 데이터 통합 서비스는 매핑에서 설정된 추적 수준을 사용합니다.- 간단. 데이터 통합 서비스가 거부된 데이터 알림, 오류 메시지 및 초기화 정보를 기록합니다.- 보통. 데이터 통합 서비스가 변환 행 오류로 인한 초기화/상태 정보, 발생한 오류 및 건너뛴 행을 기록합니다. 매핑 결과를 요약하지만 개별 행 수준이 아닙니다.- 자세한 정보 표시 초기화. 데이터 통합 서비스가 보통 추적 이외에 추가 초기화 세부 정보, 사용되는 데이터 파일과 인덱스의 이름 및 세부 변환 통계를 기록합니다.- 자세한 정보 표시 데이터. 데이터 통합 서비스가 자세한 정보 표시 초기화 추적 외에 매핑으로 전달되는 각 행을 기록합니다. 또한 데이터 통합 서비스가 열의 전체 자릿수에 맞춰 문자열 데이터를 자르는 위치를 기록하고 자세한 변환 통계를 제공합니다. <p>기본값은 없음입니다.</p>
정렬 순서	<p>데이터 통합 서비스에서 매핑의 문자 데이터를 정렬하는 순서입니다.</p> <p>기본값은 이진입니다.</p>

속성	설명
최적화 수준	<p>데이터 통합 서비스가 매핑에 적용하는 최적화 방법을 다음과 같이 제어합니다.</p> <p>자동</p> <p>데이터 통합 서비스가 실행 모드와 매핑 콘텐츠에 기반하여 최적화를 적용합니다.</p> <p>없음</p> <p>데이터 통합 서비스가 최적화를 적용하지 않습니다.</p> <p>최소</p> <p>데이터 통합 서비스가 초기 예측 최적화 방법을 적용합니다.</p> <p>일반</p> <p>데이터 통합 서비스가 초기 예측, 초기 선택, 분기 잘라내기, 푸시인, 글로벌 조건자 최적화 및 조건자 최적화 방법을 적용합니다.</p> <p>전체</p> <p>데이터 통합 서비스가 비용 기반, 초기 예측, 초기 선택, 분기 잘라내기, 조건자, 푸시인, 반 조인 및 데이터십 조인 최적화 방법을 적용합니다.</p> <p>기본값은 자동입니다.</p>
많은 전체 자릿수	<p>많은 전체 자릿수로 매핑을 실행합니다.</p> <p>전체 자릿수가 많은 데이터 값은 정확도가 더 높습니다. 매핑에서 전체 자릿수가 15자리보다 많은 값 등의 큰 숫자 값을 생성하는 경우 정확한 값이 필요하면 많은 전체 자릿수를 활성화합니다. 많은 전체 자릿수를 활성화하면 큰 숫자 값에서 전체 자릿수 손실을 방지할 수 있습니다.</p> <p>기본값은 활성화됨입니다.</p>
클라이언트로 로그 보내기	<p>Developer 도구에서 로그 파일을 볼 수 있도록 합니다. 이 옵션을 비활성화할 경우 로그 파일을 보려면 Administrator 도구를 사용해야 합니다.</p> <p>기본값은 활성화됨입니다.</p>

데이터 뷰어 구성

데이터 뷰어 구성은 **데이터 뷰어** 보기에서 출력을 미리 볼 때 Developer 도구가 적용하는 설정을 제어합니다.

다음과 같은 개체의 출력을 미리 보는 경우 데이터 뷰어 구성을 선택할 수 있습니다.

- 사용자 지정 데이터 개체
- 논리적 데이터 개체
- 논리적 데이터 개체 읽기 매핑
- 실제 데이터 개체
- 매핑 내 소스 및 변환
- 가상 저장 프로시저
- 가상 테이블
- 가상 테이블 매핑

데이터 뷰어 구성 작성

데이터 뷰어 보기에서 출력을 미리 볼 때 Developer 도구가 적용하는 설정을 제어할 데이터 뷰어 구성을 작성합니다.

1. 실행 > 실행 대화 상자 열기를 클릭합니다.

구성 실행 대화 상자가 나타납니다.



2. 데이터 뷰어 구성을 클릭합니다.

- 3.



새로 만들기 단추를 클릭합니다().

구성 실행 대화 상자의 오른쪽 패널에 데이터 뷰어 구성 속성이 표시됩니다.



4. 데이터 뷰어 구성에 대한 이름을 입력합니다.
5. 데이터 뷰어 구성 속성을 구성합니다.
6. 적용을 클릭합니다.
7. 닫기를 클릭합니다.

Developer 도구가 데이터 뷰어 구성을 작성합니다.

매핑 구성

매핑 구성은 구성 실행 대화 상자 또는 명령줄을 통해 매핑을 실행할 때 Developer tool이 사용하는 매핑 배포 속성을 제어합니다.

Developer tool을 통해 실행하는 매핑에 매핑 구성을 적용하려면 구성 실행 대화 상자를 통해 매핑을 실행하거나 고급 옵션을 사용하여 매핑을 실행합니다. 실행 메뉴 또는 매핑 편집기를 통해 매핑을 실행하면 Developer tool이 기본 매핑 배포 속성을 사용하여 매핑을 실행합니다.

명령줄에서 실행하는 매핑에 매핑 배포 속성을 적용하려면 매핑을 응용 프로그램에 추가할 때 매핑 구성을 선택합니다. 선택하는 매핑 구성은 응용 프로그램의 모든 매핑에 적용됩니다.

응용 프로그램을 편집할 때 매핑 배포 속성을 변경할 수 있습니다. 또한 관리자는 **Administrator** 도구를 통해 매핑 배포 속성을 변경할 수 있습니다. 변경 내용을 적용하려면 응용 프로그램을 다시 배포해야 합니다.

infacmd가 포함된 매개 변수 집합 사용에 대한 자세한 내용은 *Developer 매핑 가이드*를 참조하십시오.

매핑 구성 작성

매핑 실행 시 **Developer tool**이 사용하는 매핑 배포 속성을 제어하는 매핑 구성을 생성합니다.

1. **실행 > 실행 대화 상자 열기**를 클릭합니다.
구성 실행 대화 상자가 나타납니다.
2. **매핑 구성**을 클릭합니다.
3. **새로 만들기** 단추.
구성 실행 대화 상자의 오른쪽 패널에 매핑 구성 속성이 표시됩니다.
4. 매핑 구성의 이름을 입력합니다.
5. 매핑 구성 속성을 구성합니다.
6. **적용**을 클릭합니다.
7. **닫기**를 클릭합니다.
Developer tool이 매핑 구성을 작성합니다.

웹 서비스 구성

웹 서비스 구성은 **데이터 뷰어** 보기에서 웹 서비스의 출력을 미리 볼 때 **Developer** 도구가 적용하는 설정을 제어합니다.

특정 웹 서비스에 사용할 설정을 제어하는 웹 서비스 구성을 작성합니다. 작업 매핑의 출력 또는 작업 매핑에 있는 변환을 미리 볼 때 웹 서비스 구성을 선택할 수 있습니다.

참고: 미리 보는 모든 웹 서비스에 적용되는 웹 서비스 구성을 작성하려면 **기본 설정** 대화 상자를 사용하여 기본 웹 서비스 구성을 업데이트합니다.

웹 서비스 구성 작성

데이터 뷰어 보기에서 웹 서비스의 출력을 미리 볼 때 **Developer** 도구가 적용하는 설정을 제어할 웹 서비스 구성을 작성합니다.

1. **실행 > 실행 대화 상자 열기**를 클릭합니다.
실행 대화 상자가 표시됩니다.
2. **웹 서비스 구성**을 클릭합니다.
3. **새로 만들기**를 클릭합니다.
4. 웹 서비스 구성의 이름을 입력합니다.
5. 웹 서비스 구성 속성을 구성합니다.
6. **적용**을 클릭합니다.
7. **닫기**를 클릭합니다.

기본 구성 속성 업데이트

기본 데이터 뷰어, 매핑 및 웹 서비스 구성 속성을 업데이트할 수 있습니다.

1. **창 > 기본 설정**을 클릭합니다.
기본 설정 대화 상자가 표시됩니다.
2. **Informatica > 구성 실행**을 클릭합니다.
3. **데이터 뷰어, 매핑 또는 웹 서비스** 구성을 선택합니다.
4. 기본 데이터 뷰어, 매핑 또는 웹 서비스 구성 속성을 구성합니다.
5. **확인**을 클릭합니다.
Developer 도구가 기본 구성 속성을 업데이트합니다.

구성 문제 해결

이름은 같지만 대/소문자 다른 두 개의 구성을 작성했습니다. 그런데 Developer 도구를 닫고 다시 열면 한 구성이 없습니다.

데이터 뷰어 및 매핑 구성 이름은 대/소문자를 구분하지 않습니다. 이름은 같지만 대/소문자가 다른 여러 구성을 작성할 경우 사용자가 종료할 때 Developer 도구는 구성 중 하나를 삭제합니다. Developer 도구는 두 구성 이름을 고유하게 간주하지 않습니다.

이름이 긴 구성을 작성하려고 했지만 Developer 도구가 파일을 쓸 수 없다는 내용의 오류 메시지를 표시합니다.

Developer 도구는 Developer 도구를 실행하는 시스템의 파일에 데이터 뷰어 및 매핑 구성을 저장합니다. 예를 들어 100자가 넘는 긴 이름으로 구성을 작성하는 경우 Developer 도구가 파일을 하드 드라이브에 저장하지 못할 수 있습니다.

이 문제를 해결하려면 구성 이름을 줄이십시오.

사용자 지정된 데이터 개체에 대한 미리 보기 데이터가 실패했습니다.

사용자 지정된 데이터 개체가 데이터 유형 변환을 수행하는 경우 푸시다운 최적화가 소스에 설정되어 있지 않으면 Developer tool이 데이터 뷰어 보기에 데이터를 표시할 수 없습니다.

이 문제를 해결하려면 처리를 소스로 푸시하도록 푸시다운 최적화를 구성하십시오.

데이터 내보내기

데이터 뷰어 보기에 표시되는 데이터를 TXT 또는 CSV 파일 같은 탭으로 구분된 플랫폼 파일로 내보낼 수 있습니다. 계층 데이터 유형의 데이터를 JSON 파일로 내보낼 수 있습니다. 데이터의 로컬 복사본을 작성하려는 경우 데이터를 내보냅니다.

1. **데이터 뷰어** 보기에서 결과를 마우스 오른쪽 단추로 클릭하고 **데이터 내보내기**를 선택합니다.
2. 파일 이름과 확장자를 입력합니다.
3. 파일을 저장할 위치를 선택합니다.
4. **확인**을 클릭합니다.

개체 종속성

개체를 변경하거나 삭제하기 전에 모델 리포지토리에서 해당 개체가 영향을 주는 모든 개체에 대한 개체 종속성을 확인할 수 있습니다. 해당 개체에 종속된 모든 개체 및 이 개체가 종속된 개체에 대한 개체 종속성을 볼 수 있습니다.

예를 들어 여러 응용 프로그램에 배포된 데이터 개체를 삭제하려고 할 수 있습니다. 하지만 이 데이터 개체를 삭제하면 배포된 응용 프로그램에 영향을 주는지 여부를 알지 못합니다. 개체 종속성을 확인하여 이 데이터 개체가 배포된 응용 프로그램에 영향을 주는지 여부를 파악할 수 있습니다. 데이터 개체를 삭제한 후 응용 프로그램을 다시 배포할 수 있습니다.

개체 종속성 보기에서 개체 종속성을 볼 수 있습니다. 종속 개체가 사용자에게 읽기 권한이 없는 프로젝트에 있을 경우 **Developer** 도구가 해당 개체를 **개체 종속성** 보기에 표시하지 않습니다. **Developer** 도구는 **Developer** 도구가 개체에 액세스할 수 없다는 내용의 메시지를 표시합니다.

개체 종속성 보기

개체 종속성 보기에서 개체에 대한 개체 종속성을 볼 수 있습니다. **개체 탐색기** 보기에서 선택하는 개체에 대한 종속성이나 **연결 탐색기** 보기에서 선택하는 연결에 대한 종속성을 볼 수 있습니다.

다음과 같은 작업을 수행하여 개체 종속성을 확인할 수 있습니다.

순환 종속성 보기.

순환 종속성은 두 개체가 서로 종속되는 경우 발생합니다. 예를 들어 개체 A가 개체 B에 종속되고 개체 B가 개체 A에 종속됩니다. **Developer** 도구는 개체 종속성 트리에서 개체의 두 번째 인스턴스를 발견하면 개체의 추가 인스턴스를 표시하지 않습니다. 대신 **Developer** 도구는 원형 아이콘을 개체에 대해 추가하여 순환 종속성이 발생했음을 나타냅니다.

다운 스트림 또는 업스트림 종속성 보기.

다운스트림 종속성을 보고 선택한 개체에 종속된 개체를 확인할 수 있습니다. 기본적으로 **Developer** 도구는 다운스트림 종속성을 표시합니다.

업스트림 종속성을 보고 선택한 개체가 종속된 개체를 확인할 수 있습니다.

개체 종속성 필터링.

개체 종속성을 필터링하여 종속 개체의 목록을 좁힐 수 있습니다. 개체 유형 또는 프로젝트별로 필터링하도록 선택할 수 있습니다. 예를 들어 특정 개체가 영향을 주는 워크플로우를 확인할 수 있습니다. 개체 유형별로 필터링하고 워크플로우를 선택할 수 있습니다.

개체 종속성 기록 보기.

최근 10개 개체 종속성에 대한 기록을 봅니다.

개체 종속성 보기

종속성을 볼 때 필터를 적용하거나 기록을 지우거나 특정 종속성에 대한 자세한 정보를 선택할 수 있습니다.

1. **개체 탐색기** 보기 또는 **연결 탐색기** 보기에서 개체 또는 연결을 마우스 오른쪽 단추로 클릭하고 **종속성 찾기**를 클릭합니다.

개체 종속성 보기에서 선택한 개체 또는 연결에 대한 개체 종속성 목록이 개체 종속성 트리로 표시됩니다.

2. 개체 종속성 트리에서 개체 또는 연결을 선택하여 개체 또는 연결 속성을 표시합니다.
3. 필요한 경우 종속성을 필터링하려면 **필터** 아이콘을 클릭하고 개체 유형 또는 프로젝트별로 필터링하도록 선택합니다.
4. 다음과 같은 유형의 종속성을 확인할 수 있습니다.

- 원형. Developer 도구는 원형 아이콘을 개체에 대해 추가하여 순환 종속성이 발생했음을 나타냅니다.
 - 업스트림. 업스트림 종속성을 보려면 **업스트림 종속성** 아이콘을 클릭합니다.
 - 다운스트림. 다운스트림 종속성을 보려면 **다운스트림 종속성** 아이콘을 클릭합니다.
5. 필요한 경우 종속성 기록을 보려면 **기록** 아이콘을 클릭하여 개체 종속성 기록을 표시합니다. 필요한 경우 **기록 지우기**를 클릭하여 기록을 지웁니다.

개체 종속성 필터링

개체 종속성 트리를 **유형** 및 **프로젝트**별로 필터링할 수 있습니다. 필터링된 결과에서 일부 개체는 상위 개체에 직접 종속되지 않을 수 있습니다. 이러한 개체는 개체 종속성 트리에서 상위 개체에 대한 전체 경로를 숨기는 더하기(+) 아이콘과 함께 표시됩니다. 경로를 확장하도록 선택하여 상위 개체까지의 모든 개체 수준을 삽입할 수 있습니다.

1. **개체 종속성** 보기에서 **필터** 아이콘을 클릭합니다.
필터 대화 상자가 열립니다.
2. **유형** 탭에서 기본 선택 항목을 그대로 사용하여 개체 유형별로 필터링합니다.
3. 필요한 경우 **프로젝트** 탭을 클릭하고 모델 리포지토리의 모든 프로젝트로 필터링하거나 각 프로젝트별로 필터링합니다.
 - 모델 리포지토리의 모든 프로젝트로 필터링하려면 모든 프로젝트를 포함하는 옵션을 선택합니다.
 - 각 프로젝트별로 필터링하려면 열려 있는 프로젝트를 포함하는 옵션을 선택합니다.
4. 개체 유형 또는 프로젝트를 선택하거나 모든 개체 유형 또는 프로젝트를 선택하거나 선택한 개체 유형 및 프로젝트를 지웁니다.
 - 개체 유형 또는 프로젝트를 선택하려면 각 개체 유형 또는 프로젝트를 선택합니다.
 - 모든 개체 유형 또는 프로젝트를 선택하려면 **모두 선택**을 클릭합니다.
 - 모든 선택한 개체 유형 및 프로젝트를 제거하려면 **선택 안 함**을 클릭합니다.
5. **확인**을 클릭합니다.
필터링된 개체 종속성 트리가 **개체 종속성** 보기에 나타납니다.
6. 필요한 경우 더하기(+) 아이콘이 개체 이름에 대해 나타나면 개체를 마우스 오른쪽 단추로 클릭하고 **경로 확장**을 클릭하여 상위 개체까지의 모든 개체 수준을 삽입합니다.

로그

매핑을 실행하거나, 프로필을 실행하거나, 데이터를 미리 보거나, SQL 쿼리를 실행하면 데이터 통합 서비스가 로그 이벤트를 생성합니다. 로그 이벤트에는 데이터 통합 서비스가 수행한 태스크, 오류 및 로드 요약과 변환 통계가 포함됩니다.

Developer 도구에서 생성된 로그를 보고 로컬 디렉터리에 저장할 수 있습니다.

데이터 뷰어 보기에서 **로그 표시** 단추를 사용하여 로그 이벤트를 볼 수 있습니다.

실행 > 매핑 실행을 통해 매핑을 실행할 경우 **진행률** 보기에서 로그 이벤트를 볼 수 있습니다. Developer 도구에서 로그 이벤트를 열려면 매핑 실행 링크를 클릭하고 **로그로 이동**을 선택합니다.

프로필을 실행하면 **Monitoring** 도구에서 로그 이벤트를 볼 수 있습니다.

로그를 파일로 저장하려면 **파일 > 다른 이름으로 사본 저장**을 클릭하고 디렉터리를 선택합니다. 기본적으로 로그 파일은 다음 디렉터리에 저장됩니다. c:\[TEMP]\AppData\Local\Temp.

로그 파일 형식

로그 파일의 정보는 실행 중 이벤트의 시퀀스에 따라 달라집니다. 로그로 전송되는 정보의 양은 추적 수준에 따라 다릅니다.

매핑을 실행하거나, 프로필을 실행하거나, 데이터를 미리 보거나, SQL 쿼리를 실행하면 데이터 통합 서비스가 다음 정보로 로그 파일을 업데이트합니다.

논리적 DTM 메시지

매핑을 컴파일하고 최적화하고 변환하기 위한 준비와 관련된 정보를 포함합니다. 로그 이벤트와 정보의 양은 구성 속성 집합에 따라 달라집니다.

DTM(Data Transformation 관리자) 메시지

소스에 대한 연결을 설정하고, 데이터를 읽고, 데이터를 변환하고, 데이터를 대상으로 로드하는 것과 관련된 정보를 포함합니다.

로드 요약 및 변환 통계 메시지

소스에서 읽은 행 수, 대상으로 출력한 행 수, 거부된 행 수 및 실행할 시간에 대한 정보를 포함합니다.

유효성 검사 기본 설정

유효성 검사 로그 보기에 표시되는 그룹당 오류 메시지 수를 제한할 수 있습니다. 또한 **유효성 검사 로그** 보기에 새 개체 또는 개체 유형별로 오류 메시지를 그룹화할 수 있습니다.

오류 메시지 그룹화

유효성 검사 로그 보기에서 오류 메시지를 그룹화하여 개체 또는 개체 유형별로 메시지를 구성합니다. 그렇지 않으면 메시지가 사전 순으로 표시됩니다.

유효성 검사 로그 보기에서 오류 메시지를 그룹화하려면 **메뉴 > 그룹 기준**을 선택한 후 **개체** 또는 **개체 유형**을 선택합니다.

오류 메시지 그룹을 제거하려면 **메뉴 > 그룹 기준 > 없음**을 선택합니다. 오류 메시지가 **유효성 검사 로그** 보기에 새 그룹화되지 않고 사전 순으로 나타납니다.

오류 메시지 제한

유효성 검사 로그 보기에 표시되는 그룹당 오류 메시지 수를 제한할 수 있습니다. 이 제한은 그룹에서 표시되는 메시지 수 또는 **유효성 검사 로그** 보기에서 표시되는 총 메시지 수를 결정합니다. 오류 메시지가 사전 순으로 나열되고 제한이 적용될 때 아래에서 위의 순서로 삭제됩니다.

1. **창 > 기본 설정**을 클릭합니다.
기본 설정 대화 상자가 표시됩니다.
2. **Informatica > 유효성 검사**를 선택합니다.
3. 필요한 경우 오류 제한을 설정하고 나타나는 항목의 수를 구성합니다.
기본값은 100입니다.
4. 기본값을 복원하려면 **기본값 복원**을 클릭합니다.
5. **적용**을 클릭합니다.

6. **확인**을 클릭합니다.

Developer tool에서 작업 모니터링

Developer tool에서 모니터링 도구에 액세스하여 응용 프로그램 및 작업의 상태를 모니터링할 수 있습니다. 관리자는 Administrator 도구에서도 응용 프로그램 및 작업을 모니터링할 수 있습니다.

응용 프로그램 및 작업을 모니터링하여 개체에 대한 속성, 런타임 통계 및 런타임 보고서를 확인할 수 있습니다.

작업을 모니터링할 경우 작업에 대한 요약 통계 및 실행 통계를 확인할 수 있습니다. **요약 통계** 보기에는 도메인에 포함된 작업의 상태에 대한 그래프 개요가 표시됩니다.

실행 통계 보기에는 작업에 대한 일반 속성과 상태 정보가 표시됩니다. 예를 들어 작업을 시작한 사용자 및 작업을 완료하는 데 소요된 시간을 확인할 수 있습니다. 매핑 작업을 모니터링할 경우에는 실행한 작업에 대한 처리량 및 리소스 사용량 통계도 확인할 수도 있습니다.

Developer tool에서 응용 프로그램 및 작업을 모니터링하려면 **진행률** 보기를 열고 **보기 메뉴 > 작업 모니터링**을 클릭합니다. 응용 프로그램 및 작업을 실행하는 데이터 통합 서비스를 선택하고 **확인**을 클릭합니다. Monitoring 도구가 열립니다.

제 9 장

응용 프로그램 배포

이 장에 포함된 항목:

- [응용 프로그램 배포 개요, 133](#)
- [응용 프로그램 작성, 134](#)
- [응용 프로그램 속성, 135](#)
- [응용 프로그램에 배포, 137](#)
- [보관 파일에 배포, 138](#)
- [리소스 매개 변수를 사용한 배포, 139](#)
- [런타임 개체 보기, 140](#)
- [응용 프로그램 재배포, 140](#)
- [응용 프로그램 상태 정보, 141](#)
- [응용 프로그램 작성, 배포 및 업데이트 방법, 141](#)

응용 프로그램 배포 개요

응용 프로그램은 실제 데이터 개체, 논리적 데이터 개체, 데이터 서비스, 매핑, 맵렛, 변환, 웹 서비스, 워크플로우를 포함할 수 있는 배포 가능한 개체입니다.

Developer tool 외부에서 응용 프로그램 개체에 액세스할 수 있으려면 개체가 포함된 응용 프로그램을 배포해야 합니다. 새 응용 프로그램을 생성하거나 개체를 배포할 때 응용 프로그램을 생성할 수 있습니다. 응용 프로그램을 배포하는 경우 데이터 구조의 변경 사항에서 응용 프로그램 개체를 격리합니다.

다음과 같은 방법으로 응용 프로그램 개체를 배포할 수 있습니다.

데이터 통합 서비스에 배포.

데이터 통합 서비스에 응용 프로그램을 배포하면 응용 프로그램이 실행되고 최종 사용자가 응용 프로그램에 연결할 수 있습니다.

응용 프로그램의 개체 유형에 따라, 적절한 사용 권한이 있는 최종 사용자는 개체를 쿼리하거나, 웹 서비스에 액세스하거나, 명령줄에서 매핑 또는 워크플로우를 실행할 수 있습니다. 또한 개체를 배포하여 사용자가 타사 클라이언트 도구를 통해 개체를 쿼리하도록 할 수 있습니다.

네트워크 파일 시스템에 배포.

개체를 네트워크 파일 시스템에 배포하면 Developer tool이 응용 프로그램 보관 파일을 생성합니다. 개체를 버전 제어 시스템에서 보관하려면 개체를 보관 파일에 배포합니다.

조직에서 관리자가 데이터 통합 서비스에 개체를 배포해야 하는 경우 관리자는 **Administrator** 도구를 사용하여 데이터 통합 서비스에 응용 프로그램 보관 파일을 배포할 수 있습니다.

또한 응용 프로그램 보관 파일에서 모델 리포지토리의 프로젝트 또는 폴더로 개체를 가져올 수 있습니다.

데이터 통합 서비스에 응용 프로그램 개체 또는 응용 프로그램 보관 파일을 배포하면 **Developer tool**을 사용하여 런타임 응용 프로그램에 연결하고 런타임 개체의 읽기 전용 사본을 볼 수 있습니다.

응용 프로그램 작성

응용 프로그램을 생성하고 배포하여 응용 프로그램에 포함된 기타 개체와 매핑을 실행합니다.

데이터 통합 서비스에 개체를 배포할 때 응용 프로그램을 생성하거나 새 응용 프로그램을 생성할 수 있습니다. 새 응용 프로그램을 생성하는 경우 응용 프로그램에 포함할 개체를 선택합니다.

매핑, 워크플로우, 데이터 개체 및 매개 변수 집합 같은 개체를 추가할 수 있습니다. 응용 프로그램에 논리적 데이터 개체를 직접 추가할 수는 없습니다. 하지만 논리적 데이터 개체를 포함하는 **SQL** 데이터 서비스 및 웹 서비스 용으로 응용 프로그램을 작성할 수 있습니다.

응용 프로그램을 생성할 때 전체 응용 프로그램 또는 증분 응용 프로그램을 생성할 수 있습니다. 전체 응용 프로그램은 응용 프로그램을 업데이트할 때마다 재배포해야 하는 응용 프로그램입니다. 증분 응용 프로그램은 응용 프로그램 패치를 배포하여 업데이트할 수 있는 응용 프로그램입니다. 패치를 사용하면 다른 응용 프로그램 개체가 계속해서 실행되는 동안 응용 프로그램 개체를 업데이트할 수 있지만 전체 응용 프로그램을 재배포하는 것에 비해 실행 시간이 더 길어질 수 있습니다.

응용 프로그램을 생성한 후에는 **Developer tool**에서 응용 프로그램의 유효성을 검사합니다. 응용 프로그램의 유효성은 포함된 개체의 유효성에 달려 있습니다. 예를 들어, 매핑의 개체 간 링크의 구성은 유효한 매핑이 포함되어 있는 응용 프로그램에 대해 유효해야 합니다. 응용 프로그램이 유효하지 않은 경우 유효성 검사 로그 보기 또는 오류 대화 상자에 오류가 표시됩니다.

전체 응용 프로그램

전체 응용 프로그램은 데이터 통합 서비스에 배포할 수 있는 응용 프로그램입니다. 응용 프로그램 또는 응용 프로그램 개체를 편집할 때마다 응용 프로그램을 재배포해야 변경 내용이 적용됩니다.

참고: 응용 프로그램 재배포 중에는 응용 프로그램에 포함된 모든 개체에 액세스할 수 없습니다.

예

검색 기능을 실행하는 하나의 매핑을 작성하고 선택된 검색 결과를 장바구니에 넣는 다른 매핑을 작성합니다. 두 개의 매핑을 포함하는 응용 프로그램을 작성한 다음 이 응용 프로그램을 데이터 통합 서비스에 배포할 수 있습니다. 응용 프로그램 개체의 출력을 테스트한 후 개체를 변경합니다. 응용 프로그램을 업데이트하려면 응용 프로그램을 재배포합니다. 또한 응용 프로그램을 응용 프로그램 보관 파일에 배포합니다. 관리자는 보관 파일을 버전 제어 시스템으로 체크 인합니다.

증분 응용 프로그램

증분 응용 프로그램은 데이터 통합 서비스에 배포한 후 응용 프로그램 패치를 배포하여 업데이트할 수 있는 응용 프로그램입니다.

패치에는 업데이트하려는 개체 또는 배포된 증분 응용 프로그램에 추가하려는 개체가 포함됩니다. 패치를 배포하면 데이터 통합 서비스가 패치 콘텐츠에 따라 배포된 응용 프로그램을 업데이트합니다.

패치를 사용하면 응용 프로그램 가동 시간을 늘릴 수 있습니다. 예를 들어 스트리밍 매핑이 포함된 응용 프로그램이 있는 경우 스트리밍 매핑을 사용해 계속해서 데이터를 처리하면서 응용 프로그램의 다른 개체를 업데이트할 수 있습니다.

응용 프로그램 패치에 대한 자세한 내용은 [장 10, “응용 프로그램 패치 배포” 페이지 148](#) 항목을 참조하십시오.

참고: 증분 응용 프로그램에는 웹 서비스, SQL 데이터 서비스 또는 REST 웹 서비스가 포함될 수 없습니다.

예

매핑 100개를 생성하고 이러한 매핑이 포함되는 증분 응용 프로그램을 생성합니다. 그런 다음 응용 프로그램을 데이터 통합 서비스에 배포합니다. 응용 프로그램 개체의 출력을 테스트한 후 일부 개체를 변경합니다.

런타임 응용 프로그램의 개체를 업데이트하려면 변경한 개체가 포함된 응용 프로그램 패치를 생성합니다. 응용 프로그램 패치를 데이터 통합 서비스에 배포하면 데이터 통합 서비스가 증분 응용 프로그램에 패치를 적용하여 개체를 업데이트합니다. 다른 사용자가 실행 중일 수 있는 다른 응용 프로그램 개체는 중지되지 않습니다. 또한 응용 프로그램 및 응용 프로그램 패치를 응용 프로그램 보관 파일에 배포합니다. 관리자는 보관 파일을 버전 제어 시스템으로 체크 인합니다.

응용 프로그램 속성

응용 프로그램을 작성한 후 **응용 프로그램** 편집기에서 속성을 편집할 수 있습니다.

일반 응용 프로그램 속성

다음 테이블에는 **개요** 탭에서 설정할 수 있는 일반 응용 프로그램 속성이 설명되어 있습니다.

속성	설명
이름	응용 프로그램의 이름입니다.
설명	선택 사항입니다. 응용 프로그램의 설명입니다.
응용 프로그램 유형	응용 프로그램의 유형입니다. 전체 또는 증분입니다.

매핑 배포 속성

다음 테이블에는 응용 프로그램에 매핑이 포함되어 있을 때 **고급** 탭에서 설정할 수 있는 매핑 배포 속성이 설명되어 있습니다.

속성	설명
기본 날짜/시간 형식	매핑이 문자열을 날짜로 변환하는 경우 데이터 통합 서비스가 사용하는 날짜/시간 형식입니다. 기본값은 MM/DD/YYYY HH24:MI:SS입니다.
추적 수준 재정의	매핑에서 각 변환에 사용되는 추적 수준을 재정의합니다. 추적 수준은 데이터 통합 서비스가 매핑 로그 파일에 보내는 정보의 양을 결정합니다. 다음 추적 수준 중 하나를 선택합니다. <ul style="list-style-type: none"> - 없음. 데이터 통합 서비스가 각 변환에 대해 설정된 추적 수준을 재정의하지 않습니다. - 간단. 데이터 통합 서비스가 거부된 데이터 알림, 오류 메시지 및 초기화 정보를 기록합니다. - 보통. 데이터 통합 서비스가 변환 행 오류로 인해 건너뛴 행, 초기화/상태 및 발생한 오류 정보를 기록합니다. 데이터 통합 서비스가 매핑 결과를 요약합니다(개별 행 수준에서는 제외). - 자세한 정보 표시 초기화. 데이터 통합 서비스가 보통 추적 이외에 추가 초기화 세부 정보, 사용되는 데이터 파일과 인덱스의 이름 및 세부 변환 통계를 기록합니다. - 자세한 정보 표시 데이터. 데이터 통합 서비스가 자세한 정보 표시 초기화 추적 외에 매핑으로 전달되는 각 행을 기록합니다. 로그에는 열의 전체 자릿수에 맞도록 데이터 통합 서비스가 문자열 데이터를 자른 위치가 기록되어 있습니다. 로그에는 세부 변환 통계가 포함되어 있습니다. 데이터 통합 서비스는 변환을 처리할 때 블록의 모든 행에 대해 행 데이터를 씁니다. 참고: 워크플로우 로그에서 Spark 엔진 정보를 보려면 Developer tool에서 자세한 정보 표시 초기화 또는 자세한 정보 표시 데이터를 선택합니다. 기본값은 없음입니다.
정렬 순서	데이터 통합 서비스에서 매핑의 문자 데이터를 정렬하는 순서입니다. 기본값은 이진입니다.

속성	설명
최적화 수준	<p>데이터 통합 서비스가 매핑에 적용하는 최적화 방법을 다음과 같이 제어합니다.</p> <p>자동</p> <p>데이터 통합 서비스가 실행 모드와 매핑 콘텐츠에 기반하여 최적화를 적용합니다.</p> <p>없음</p> <p>데이터 통합 서비스가 최적화를 적용하지 않습니다.</p> <p>최소</p> <p>데이터 통합 서비스가 초기 예측 최적화 방법을 적용합니다.</p> <p>일반</p> <p>데이터 통합 서비스가 초기 예측, 초기 선택, 분기 잘라내기, 푸시인, 글로벌 조건자 최적화 및 조건자 최적화 방법을 적용합니다.</p> <p>전체</p> <p>데이터 통합 서비스가 비용 기반, 초기 예측, 초기 선택, 분기 잘라내기, 조건자, 푸시인, 반 조인 및 데이터십 조인 최적화 방법을 적용합니다.</p> <p>기본값은 자동입니다.</p>
높은 정밀도	<p>높은 정밀도로 매핑을 실행합니다.</p> <p>높은 정밀도 데이터 값은 정확도가 더 높습니다. 매핑에서 전체 자릿수가 15자리보다 많은 값 등의 큰 숫자 값을 생성하는 경우 정확한 값이 필요하면 높은 정밀도를 활성화합니다. 높은 정밀도를 활성화하면 큰 숫자 값에서 전체 자릿수 손실을 방지할 수 있습니다.</p> <p>기본값은 활성화됨입니다.</p>

응용 프로그램에 배포

개체를 응용 프로그램 또는 응용 프로그램의 일부인 데이터 서비스로 배포할 수 있습니다.

일부 개체를 웹 서비스 또는 SQL 데이터 서비스로 배포할 수 있습니다. 먼저 응용 프로그램을 작성하고 개체를 추가합니다. 그러면 응용 프로그램을 배포할 때 **Developer tool**에서 개체 기반 서비스를 작성하라는 메시지를 표시합니다. **Developer tool**이 서비스를 응용 프로그램에 추가합니다.

또한 개체를 배포하여 사용자가 타사 클라이언트 도구를 통해 개체를 쿼리하도록 할 수 있습니다.

사용자가 응용 프로그램에 포함된 매핑, 워크플로우 및 기타 개체에 액세스할 수 있도록 하려면 응용 프로그램을 데이터 통합 서비스에 배포합니다.

응용 프로그램을 배포할 때 데이터 통합 서비스가 응용 프로그램 개체를 실행합니다.

매핑 또는 워크플로우를 배포할 때 **Developer tool**이 응용 프로그램을 생성하라는 메시지를 표시합니다.

배포하는 응용 프로그램에 매개 변수 집합을 추가할 수 있습니다. 매개 변수 집합은 매핑과 워크플로우에서 사용할 매개 변수와 매개 변수 값이 들어 있는 모델 리포지토리의 개체입니다. 데이터 통합 서비스에서 워크플로우 또는 매핑을 실행할 때 매개 변수 값을 사용하려면 워크플로우 또는 매핑과 함께 매개 변수 집합을 배포하십시오. 응용 프로그램에 여러 개의 매개 변수 집합을 추가한 다음 매핑이나 워크플로우 실행에 서로 다른 매개 변수 집합을 사용할 수 있습니다.

이름이 같은 응용 프로그램을 동일한 데이터 통합 서비스에 배포하면 데이터 통합 서비스가 배포된 응용 프로그램과 응용 프로그램의 모든 개체를 덮어씁니다.

웹 서비스로 개체 배포

다음과 같은 개체를 웹 서비스로 배포할 수 있습니다.

- 맵렛
- 웹 서비스 소비자 변환을 제외한 변환
- 플랫 파일 데이터 개체
- 관계형 데이터 개체
- 논리적 데이터 개체

개체를 웹 서비스로 배포할 때 **Developer tool**이 개체 기반 웹 서비스를 작성하라는 메시지와 웹 서비스를 포함하는 응용 프로그램을 작성하라는 메시지를 차례로 표시합니다.

개체를 웹 서비스로 배포하는 경우 다음 정보를 입력합니다.

속성	설명
이름	웹 서비스의 이름입니다.
위치	응용 프로그램을 넣을 모델 리포지토리 프로젝트 폴더입니다.
네임스페이스	사용자가 웹 서비스에 액세스하는 URL입니다. 데이터 통합 서비스가 WSDL 파일에 정의된 URL을 재정의합니다.
접두사	대상 네임스페이스의 접두사입니다.

SQL 데이터 서비스로 개체 배포

다음과 같은 데이터 개체를 SQL 데이터 서비스로 배포할 수 있습니다.

- 실제 데이터 개체
- 논리적 데이터 개체

데이터 개체를 SQL 데이터 서비스로 배포할 때 **Developer tool**이 개체 기반 SQL 데이터 서비스를 작성하라는 메시지와 서비스를 포함하는 응용 프로그램을 작성하라는 메시지를 차례로 표시합니다.

개체를 SQL 데이터 서비스로 배포하는 경우 다음 정보를 입력합니다.

속성	설명
이름	웹 서비스의 이름입니다.
위치	응용 프로그램을 넣을 모델 리포지토리 프로젝트 폴더입니다.

보관 파일에 배포

보관 파일은 응용 프로그램 개체 및 메타데이터를 XML 형식으로 저장합니다.

다음 유형의 보관 파일을 생성할 수 있습니다.

응용 프로그램 보관 파일

응용 프로그램 보관 파일은 응용 프로그램의 응용 프로그램 개체 및 메타데이터에 대한 모든 정보를 저장하는 XML 파일입니다. 파일의 확장명은 **.iar**입니다.

응용 프로그램 패치 보관 파일

응용 프로그램 패치 보관 파일은 응용 프로그램 패치에 대한 정보를 저장하는 XML 파일입니다. 이 파일에는 패치에 있는 응용 프로그램 개체, 패치 설명, 해당 증분 응용 프로그램의 이름, 패치에서 상태 정보를 유지 또는 삭제할지 등의 정보가 저장됩니다. 파일의 확장명은 **.piar**입니다.

다음과 같은 이유로 응용 프로그램 보관 파일을 생성할 수 있습니다.

- 응용 프로그램 또는 응용 프로그램 패치 배포. 조직에서 응용 프로그램 또는 응용 프로그램 패치를 데이터 통합 서비스에 배포하는 관리자의 기능을 제한하는 경우 관리자는 데이터 통합 서비스에 보관 파일을 배포할 수 있습니다.

응용 프로그램을 배포하려는 경우 관리자는 **Administrator** 도구 또는 **infacmd dis deployApplication**을 사용할 수 있습니다. 응용 프로그램 패치를 배포하려는 경우 관리자는 **infacmd tools patchApplication**을 사용할 수 있습니다.

- 모델 리포지토리로 응용 프로그램 또는 응용 프로그램 패치 가져오기. 관리자는 **infacmd tools importObjects**를 사용하여 보관 파일을 모델 리포지토리로 가져올 수 있습니다. 이미 모델 리포지토리에 있는 개체가 포함된 보관 파일을 가져오는 경우 **Developer tool**은 새 프로젝트 폴더를 생성합니다.

참고: 응용 프로그램을 가져오는 모델 리포지토리의 릴리스 버전은 응용 프로그램을 보관 파일로 내보낸 버전과 호환되어야 합니다.

- 파일을 다른 시스템에 보관. 예를 들어 모델 리포지토리가 버전 제어 시스템과 통합되어 있지 않은 경우 관리자는 보관 파일을 버전 제어 시스템으로 체크 인하여 프로젝트 감사 또는 개체 복구에 사용할 수 있습니다.

리소스 매개 변수를 사용한 배포

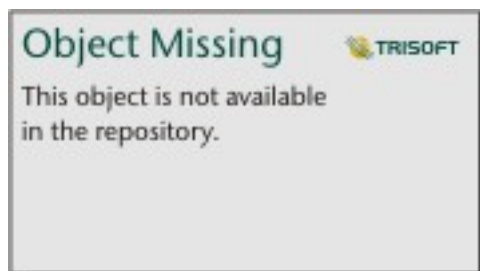
소스, 조회 또는 대상 변환에 리소스 매개 변수가 포함된 매핑을 배포하는 경우 응용 프로그램에서 매핑을 실행하는 데 필요한 데이터 개체의 목록을 볼 수 있습니다.

응용 프로그램을 생성하거나 매핑을 배포할 때 리소스 매개 변수 값의 데이터 개체를 확인할 수 있습니다. 데이터 개체를 보려면 **응용 프로그램 생성** 대화 상자 또는 **배포** 대화 상자에서 **종속** 링크를 클릭합니다. 데이터 개체를 볼 때 응용 프로그램에서 데이터 개체를 제외하도록 선택할 수 있습니다. 런타임 시 리소스 매개 변수의 값을 다른 데이터 개체로 변경합니다.

응용 프로그램에서 데이터 개체를 제외하려면 포함/제외 확인란의 선택을 취소합니다.

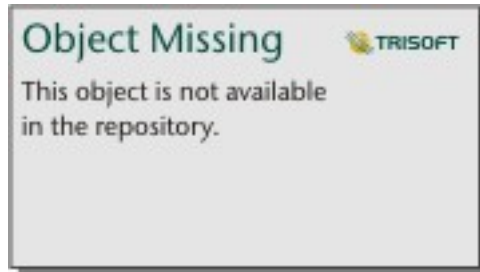
매핑에는 여러 개의 맵셋이 포함될 수 있으며 맵셋에는 다른 맵셋이 포함될 수 있습니다. 매개 변수 계층에서 다른 매개 변수가 리소스 매개 변수 값을 재정의 하는 경우 종속성 목록에 리소스 매개 변수 값이 표시되지 않습니다.

다음 이미지는 **배포** 대화 상자의 **종속** 링크를 보여 줍니다.



종속 링크를 클릭하면 **추가 종속성** 대화 상자가 나타납니다. Developer tool은 응용 프로그램에서 매핑을 실행하는 데 필요한 데이터 개체의 목록을 보여 줍니다.

다음 이미지는 **추가 종속성** 대화 상자를 보여 줍니다.



런타임 개체 보기

배포된 응용 프로그램에서 런타임 개체의 읽기 전용 사본을 보려면 런타임 응용 프로그램에 연결합니다.

다음 이미지는 **Object Explorer** 보기의 런타임 응용 프로그램을 보여 줍니다.



편집기에서 런타임 개체를 열어 응용 프로그램 또는 응용 프로그램 패치가 올바르게 배포되었는지 확인할 수 있습니다. 응용 프로그램을 재배포하거나 응용 프로그램 패치를 배포한 후 런타임 응용 프로그램을 새로 고쳐 업데이트된 개체를 볼 수 있습니다.

또한 테스트 결과나 새로운 릴리스 요구 사항을 기반으로 해당하는 디자인 타임 개체를 편집하는 동안 런타임 개체를 참조할 수도 있습니다. 런타임 개체는 해당 디자인 타임 개체의 데이터 구조 변경으로부터 격리되므로 런타임 개체를 변경하지 않고 디자인 타임 개체를 계속 편집할 수 있습니다.

응용 프로그램 재배포

응용 프로그램의 콘텐츠를 변경한 후 변경 내용을 적용하려면 응용 프로그램을 재배포해야 합니다.

다음과 같은 이유로 응용 프로그램을 재배포할 수 있습니다.

- 전체 응용 프로그램에서 응용 프로그램 개체를 편집, 추가 또는 삭제한 후 변경 내용을 배포된 전체 응용 프로그램에 적용하려고 합니다.
- 전체 응용 프로그램에서 응용 프로그램 개체를 삭제한 후 변경 내용을 배포된 증분 응용 프로그램에 적용하려고 합니다.

- 증분 응용 프로그램에 대한 응용 프로그램 패치에 다수의 개체가 포함되기 때문에 사용자가 패치를 배포하는 것이 비효율적입니다. 패치에 패치 상속으로 인한 다수의 개체가 포함되거나 런타임 응용 프로그램에 영향을 받는 개체가 많은 경우 패치 배포가 비효율적일 수 있습니다.

편집된 응용 프로그램을 배포하는 경우 응용 프로그램 상태 정보를 유지할지 아니면 무시할지 선택합니다. 상태 정보란 매핑 속성, 매핑 출력 및 런타임 개체의 속성을 가리킵니다.

응용 프로그램을 재배포하는 경우 응용 프로그램을 중지하도록 선택해야 합니다. 실행 중인 개체를 중단하지 않으려는 경우 응용 프로그램의 이름을 바꾸거나 응용 프로그램을 다른 서비스에 배포할 수 있습니다.

응용 프로그램을 업데이트하고 네트워크 파일 시스템으로 내보내는 경우 응용 프로그램 보관 파일을 바꾸거나 배포를 취소할 수 있습니다. 응용 프로그램 보관 파일을 바꾸는 경우 **Developer tool**이 응용 프로그램의 개체를 바꾸고 개체 속성을 재설정합니다.

응용 프로그램 상태 정보

응용 프로그램을 재배포하거나 응용 프로그램 패치를 배포하여 배포된 응용 프로그램을 업데이트하는 경우 배포된 응용 프로그램에 대한 상태 정보를 유지하거나 삭제하도록 선택할 수 있습니다.

상태 정보란 매핑 출력이나 시퀀스 생성기 변환과 같은 런타임 개체의 속성 및 매핑 속성을 가리킵니다. 상태 정보를 유지하면 배포된 응용 프로그램의 속성과 이러한 설정이 유지됩니다. 상태 정보를 삭제하면 배포된 응용 프로그램의 속성과 이러한 설정이 삭제됩니다.

예: 구성 가능한 속성 유지 또는 삭제

응용 프로그램에 구성 가능한 런타임 속성이 있는 매핑이 포함되어 있습니다. 높은 정밀도 속성을 **True**로 설정합니다. 응용 프로그램을 배포한 후 매핑을 편집하고 높은 정밀도 속성을 **False**로 변경합니다. 응용 프로그램을 다시 배포하고 **상태 정보 유지**를 선택하면 데이터 통합 서비스가 배포된 응용 프로그램의 상태 정보를 유지하고 런타임 속성에 대한 변경 내용을 인식하지 못합니다. 속성에 대한 변경 내용을 적용하려면 **상태 정보 유지**를 선택 취소해야 합니다.

예: 시퀀스 유지 또는 삭제

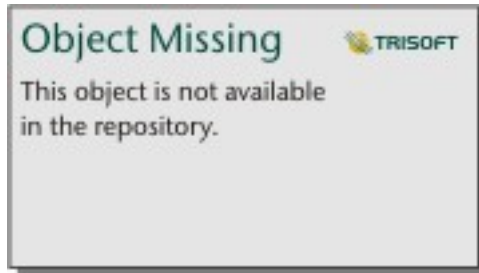
매핑에는 대상 테이블의 행에 대해 고유 키를 생성하는 시퀀스 생성기 변환이 포함되어 있습니다. 매핑이 포함된 응용 프로그램을 배포한 후 매핑이 실행되고 시퀀스 생성기 변환이 행 **1-2000**에 대해 키를 생성합니다. 다음에 매핑이 실행되면 시퀀스 생성기 변환이 **2001**로 시작하는 키를 생성하지만 **10000**에서 시퀀스가 시작하도록 결정할 수 있습니다. 새 시작 값을 지정하려면 시퀀스 생성기 변환을 편집합니다. 응용 프로그램을 다시 배포하고 **상태 정보 유지**를 선택하면 데이터 통합 서비스가 배포된 응용 프로그램의 상태 정보를 유지하고 설정에 대한 변경 내용을 인식하지 못합니다. 매핑이 다시 실행되면 시퀀스가 **2001**에서 다시 시작됩니다. 설정에 대한 변경 내용을 적용하려면 **상태 정보 유지**를 선택 취소해야 합니다.

참고: 응용 프로그램 상태를 유지하든 삭제하든 시퀀스 생성기 변환의 재설정 속성에는 영향을 주지 않습니다.

응용 프로그램 작성, 배포 및 업데이트 방법

매핑과 기타 개체를 작성 및 편집하고 최종 사용자가 액세스할 수 있도록 응용 프로그램에 배포합니다.

다음 이미지는 응용 프로그램 개발 및 배포 프로세스를 보여줍니다.



1. 데이터 액세스 및 변환을 위한 매핑, 워크플로우, 변환 및 기타 개체를 생성합니다.
2. 응용 프로그램을 생성하고 응용 프로그램에 개체를 추가합니다.
3. 응용 프로그램을 데이터 통합 서비스에 직접 배포하거나 응용 프로그램 보관 파일에 배포하도록 선택합니다.
 - 데이터 통합 서비스에서 개체를 실행할 수 있도록 하려면 응용 프로그램을 데이터 통합 서비스에 직접 배포합니다.
 - 관리자가 파일을 보관할 수 있도록 하려면 응용 프로그램을 응용 프로그램 보관 파일에 배포합니다.
4. 보관 파일을 데이터 통합 서비스에 배포합니다.
5. 응용 프로그램을 테스트하여 조직의 요구 사항을 충족하는지 확인합니다.

응용 프로그램을 테스트한 후 응용 프로그램 개체를 편집하고 배포된 응용 프로그램을 업데이트하여 응용 프로그램을 유지 관리할 수 있습니다. 응용 프로그램 개체를 쉽게 편집할 수 있도록 런타임 응용 프로그램에 연결하여 런타임 개체가 디자인되는 방식을 검토할 수 있습니다.

다음 이미지는 응용 프로그램을 유지 관리하는 방법을 보여 줍니다.



1. 출력이 요구 사항을 충족하지 않는 경우 응용 프로그램의 개체를 편집합니다.
2. 응용 프로그램을 재배포하거나 응용 프로그램 패치를 배포하여 응용 프로그램을 업데이트하도록 선택합니다. 응용 프로그램 패치는 응용 프로그램이 증분 응용 프로그램인 경우에만 배포할 수 있습니다.
3. 패치를 배포하도록 선택하는 경우 증분 배포 마법사를 사용하여 패치를 생성합니다.
4. 패치를 데이터 통합 서비스에 배포하고 필요한 경우 응용 프로그램 패치를 보관 파일에 배포합니다.

응용 프로그램 패치 생성 및 배포에 대한 자세한 내용은 "응용 프로그램 배치 배포" 장을 참조하십시오.

참고: 관리자는 명령줄 또는 Administrator 도구를 사용하여 응용 프로그램을 배포하고 기타 관리 태스크를 수행할 수도 있습니다. 이러한 태스크에 대한 자세한 내용은 *Informatica Application Service 가이드*를 참조하십시오.

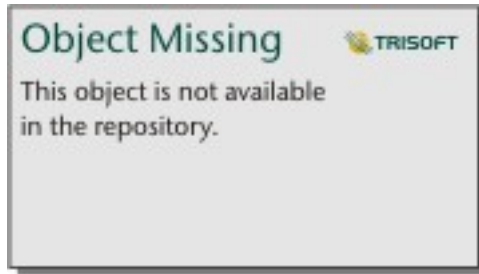
응용 프로그램 작성

응용 프로그램을 작성할 때 응용 프로그램에 포함할 개체를 선택합니다. 최종 사용자가 타사 도구를 통해 데이터에 액세스할 수 있도록 하나 이상의 개체를 배포할 때 응용 프로그램을 작성합니다.

1. **개체 탐색기** 보기에서 프로젝트나 폴더를 선택합니다.

2. 파일 > 새로 만들기 > 응용 프로그램을 클릭합니다.

새 응용 프로그램 대화 상자가 나타납니다.



3. 응용 프로그램의 이름을 입력합니다.

응용 프로그램 이름은 응용 프로그램을 배포할 때 세션 로그 파일 이름에 포함됩니다. 응용 프로그램에 따라 세션 로그 파일 이름에 응용 프로그램 이름, 기능 이름, 연결 ID와 데이터 및 날짜와 시간이 포함될 수 있습니다.

세션 로그 파일 이름은 로그 파일 경로의 로그 디렉터리 경로에 포함됩니다. Windows 운영 체제에서 로그 파일 경로는 259자로 제한됩니다. 응용 프로그램 이름을 지정할 때 이 길이 제한을 고려하십시오.

4. 찾아보기를 클릭하고 응용 프로그램 위치를 선택합니다.

응용 프로그램을 프로젝트 또는 폴더에 작성해야 합니다.

5. 응용 프로그램 유형을 선택합니다. 전체 또는 증분을 선택할 수 있습니다.

응용 프로그램을 생성한 후에는 응용 프로그램 유형을 편집할 수 없습니다.

6. 응용 프로그램에 개체를 추가하려면 추가 단추를 클릭합니다.

개체 추가 대화 상자가 나타납니다.

7. 하나 이상의 개체를 선택하고 확인을 클릭합니다.

Developer tool이 사용자가 선택한 개체를 새 응용 프로그램 대화 상자에 나열합니다.

8. 마침을 클릭하여 응용 프로그램을 작성합니다.

응용 프로그램을 생성한 후 데이터 통합 서비스 또는 응용 프로그램 보관 파일에 배포할 수 있습니다.

응용 프로그램 배포

응용 프로그램을 생성한 후에는 데이터 통합 서비스에 배포하여 응용 프로그램 개체를 실행하거나 응용 프로그램 보관 파일에 배포합니다.

1. Object Explorer 보기에서 응용 프로그램을 마우스 오른쪽 단추로 클릭하고 배포를 선택합니다.



배포 대화 상자가 표시됩니다.

2. 다음 옵션 중 하나를 선택하여 응용 프로그램을 배포합니다.

- 응용 프로그램을 데이터 통합 서비스에 배포하려면 **서비스에 배포**를 선택합니다.
 - 필요한 경우 **찾아보기**를 클릭하여 도메인을 선택합니다.
 - 응용 프로그램을 배포할 데이터 통합 서비스를 선택합니다.
- 응용 프로그램을 응용 프로그램 보관 파일에 배포하려면 **응용 프로그램 보관 파일로 내보내기**를 선택합니다. 그런 다음 **찾아보기** 단추를 클릭하여 파일을 저장하려는 디렉터리를 선택합니다.

응용 프로그램을 데이터 통합 서비스에 배포하는 경우 데이터 통합 서비스는 응용 프로그램과 포함된 개체를 실행합니다. 응용 프로그램 개체의 출력을 테스트하고 요구 사항에 대해 출력의 유효성을 검사할 수 있습니다. 필요한 경우 응용 프로그램 개체를 편집하고 배포된 응용 프로그램을 업데이트합니다.

응용 프로그램을 응용 프로그램 보관 파일에 배포하는 경우 파일을 저장하거나, 다른 모델 리포지토리로 가져올 때 사용하거나, 나중에 파일의 응용 프로그램을 데이터 통합 서비스에 배포할 수 있습니다.

개체 배포

개체를 응용 프로그램으로 데이터 통합 서비스에 배포하거나 개체를 응용 프로그램 보관 파일에 배포하여 개체를 응용 프로그램으로 내보냅니다.

1. 실행 가능한 개체 또는 플랫폼 파일 데이터 개체를 배포하도록 선택합니다.
 - 매핑, 워크플로우 또는 기타 실행 가능한 개체를 배포하려면 개체를 마우스 오른쪽 단추로 클릭하고 **배포**를 선택합니다.
 - 플랫폼 파일 데이터 개체를 배포하려면 마우스 오른쪽 단추로 개체를 클릭하고 개체를 웹 서비스, **SQL** 데이터 서비스 또는 **REST** 웹 서비스로 배포하도록 선택합니다.

Developer tool이 응용 프로그램을 작성하라는 메시지를 표시합니다.
2. 응용 프로그램 이름을 입력합니다.
3. 응용 프로그램의 위치를 선택하려면 기본 위치를 그대로 사용하거나 **찾아보기**를 클릭하고 다른 위치를 선택합니다.
4. 응용 프로그램 유형을 선택합니다. 전체 또는 증분을 선택할 수 있습니다.

플랫폼 파일 데이터 개체를 웹 서비스, **SQL** 데이터 서비스 또는 **REST** 웹 서비스로 배포하는 경우 응용 프로그램은 증분 응용 프로그램일 수 없습니다.

응용 프로그램을 생성한 후에는 응용 프로그램 유형을 편집할 수 없습니다.
5. 응용 프로그램에 개체를 추가하려면 추가 단추를 클릭하고 개체를 선택합니다.
6. **다음**을 클릭합니다.
7. 다음 옵션 중 하나를 선택하여 응용 프로그램을 배포합니다.
 - 응용 프로그램을 데이터 통합 서비스에 배포하려면 **서비스에 배포**를 선택합니다.
 - **Developer tool**에 여러 도메인에 대한 연결 정보가 있을 경우 **찾아보기**를 클릭하고 도메인을 선택합니다.
 - 응용 프로그램을 배포할 데이터 통합 서비스를 선택합니다.
 - 응용 프로그램을 응용 프로그램 보관 파일에 배포하려면 **응용 프로그램 보관 파일로 내보내기**를 선택합니다. 그런 다음 **찾아보기** 단추를 클릭하여 파일을 저장하려는 디렉터리를 선택합니다.
8. 웹 서비스에 개체를 배포하려면 다음 단계를 완료합니다.
 - a. 웹 서비스에 대한 속성을 구성합니다.
 - b. 웹 서비스에 작업을 추가하려면 **다음**을 클릭합니다.

기본적으로 **Developer tool**은 웹 서비스로 배포하는 각 개체에 대해 작업을 작성합니다.

- c. 각 작업, 작업 입력 및 작업 출력을 선택하여 속성을 표시하고 구성합니다.
 - d. **마침**을 클릭합니다.
9. SQL 데이터 서비스에 개체를 배포하려면 다음 단계를 완료합니다.
- a. SQL 데이터 서비스의 이름을 입력합니다.
 - b. 기본 위치를 그대로 사용하거나 **찾아보기**를 클릭하여 SQL 데이터 서비스에 대한 모델 리포지토리와 프로젝트 폴더 위치를 선택합니다.
 - c. **다음**을 클릭합니다.
SQL 데이터 서비스에 가상 테이블 추가 대화 상자가 나타납니다.
 - d. 추가 단추를 클릭합니다.
 - e. 가상 테이블의 이름을 입력합니다.
 - f. **데이터 개체** 열에서 열기 단추를 클릭합니다.
데이터 개체 선택 대화 상자가 나타납니다.
 - g. 실제 데이터 개체를 선택하고 **확인**을 클릭합니다.
 - h. **가상 스키마** 열에 가상 스키마 이름을 입력합니다.
 - i. 데이터 액세스 열의 **읽기**를 선택하여 가상 테이블을 데이터 개체와 연결합니다. 가상 테이블을 데이터 개체와 연결하지 않으려는 경우 **없음**을 선택합니다.
10. REST 웹 서비스에 개체를 배포하려면 다음 단계를 완료합니다.
- a. REST 웹 서비스에 대한 속성을 구성합니다.
 - b. 정의에 포함할 열을 선택합니다.
 - c. HTTP 메서드를 선택합니다.
11. **마침**을 클릭합니다.

Developer tool이 응용 프로그램을 데이터 통합 서비스에 배포합니다. 데이터 통합 서비스는 응용 프로그램과 해당 응용 프로그램에 포함된 개체를 실행합니다.

응용 프로그램 개체의 출력을 테스트하고 요구 사항에 대해 출력의 유효성을 검사할 수 있습니다. 필요한 경우 응용 프로그램 개체를 편집하고 배포된 응용 프로그램을 업데이트합니다.

응용 프로그램 보관 가져오기

보관 파일에 배포된 응용 프로그램 또는 응용 프로그램 패치를 가져와 모델 리포지토리에서 파일 콘텐츠를 사용할 수 있도록 만듭니다.

다른 모델 리포지토리에서 보관된 파일을 가져오거나 버전 제어 시스템에서 검색된 파일을 가져오는 경우 보관 파일을 생성할 때 사용된 응용 프로그램 및 응용 프로그램 개체를 가져올 수 있습니다.

1. **파일 > 가져오기**를 클릭합니다.
가져오기 마법사가 표시됩니다.
2. **Informatica > 응용 프로그램 보관**을 선택합니다.
3. **다음**을 클릭합니다.
4. **찾아보기**를 클릭하여 보관 파일을 선택합니다. 응용 프로그램 보관 파일을 가져오려면 **.iar** 확장명을 사용합니다. 응용 프로그램 패치 보관 파일을 가져오려면 **.piar** 확장명을 사용합니다.
Developer tool이 보관 파일의 콘텐츠를 나열합니다.
5. 응용 프로그램을 가져올 모델 리포지토리를 선택합니다.
6. **마침**을 클릭합니다.

Developer tool이 응용 프로그램을 리포지토리로 가져옵니다.

런타임 응용 프로그램에 연결

편집기를 사용하여 응용 프로그램에서 런타임 개체를 볼 수 있도록 런타임 응용 프로그램에 연결합니다.

1. **파일 > 런타임 응용 프로그램에 연결**을 클릭합니다.
2. 도메인을 선택합니다.
3. 데이터 통합 서비스를 선택합니다.
4. 데이터 통합 서비스에서 배포된 응용 프로그램을 선택합니다.
5. **확인**을 클릭합니다.

런타임 응용 프로그램이 **Object Explorer** 보기에 표시됩니다. 응용 프로그램에서 런타임 개체를 보도록 응용 프로그램을 확장할 수 있습니다.

응용 프로그램 편집

응용 프로그램 개체를 편집하거나 응용 프로그램 개체를 편집하여 개체를 추가 및 삭제하거나 응용 프로그램 속성을 편집합니다.

1. 응용 프로그램 개체를 편집하려면 **개체 탐색기** 보기에서 개체를 엽니다. 필요한 변경을 수행합니다.
2. 응용 프로그램을 편집하려면 **개체 탐색기**에서 마우스 오른쪽 단추로 응용 프로그램을 클릭하고 **열기**를 선택합니다.

응용 프로그램 편집기가 표시됩니다.



3. 응용 프로그램에 개체를 추가하려면 추가 단추를 클릭합니다.
개체 추가 대화 상자가 나타납니다.
4. 하나 이상의 개체를 선택하고 **확인**을 클릭합니다.
5. 삭제할 하나 이상의 개체를 선택하고 삭제 단추를 클릭합니다.
6. 응용 프로그램을 저장합니다.

응용 프로그램을 편집한 후 응용 프로그램을 업데이트하여 변경 내용을 배포된 응용 프로그램으로 전파합니다.

데이터 통합 서비스에 응용 프로그램 재배포

데이터 통합 서비스의 런타임 응용 프로그램을 디자인 타임 응용 프로그램으로 바꾸려면 응용 프로그램을 데이터 통합 서비스에 재배포합니다.

1. **Object Explorer** 보기에서 응용 프로그램을 마우스 오른쪽 단추로 클릭하고 **배포**를 클릭합니다.
배포 대화 상자가 표시됩니다.
2. **서비스에 배포**를 선택합니다.
3. 기본 데이터 통합 서비스가 설정되지 않은 경우 **찾아보기**를 클릭하고 도메인을 선택합니다.
도메인 선택 대화 상자가 나타납니다.

4. 도메인을 선택하고 **확인**을 클릭한 다음 데이터 통합 서비스를 선택합니다.

대상 이름 열이 기본적으로 동일한 이름을 가진 응용 프로그램을 표시합니다. 예를 들어 다음 이미지는 응용 프로그램 **MyApplication**의 배포를 보여 줍니다.



5. 데이터 통합 서비스에서 다른 대상 응용 프로그램을 선택하려면 대상 이름 열을 클릭합니다.
6. 응용 프로그램의 일부인 런타임 개체의 상태를 유지하려면 **상태 정보 유지**를 선택합니다.
7. 필요한 경우 **대상 응용 프로그램을 강제로 중지하여 실행 중인 개체를 중단하십시오**를 선택합니다.

응용 프로그램의 상태가 실행 중이 아닌 경우에만 응용 프로그램을 중지하지 않고 재배포할 수 있습니다. 관리자가 응용 프로그램을 중지하지 않은 경우 관리자에게 응용 프로그램의 강제 중지를 선택해도 되는지 확인하십시오.

8. **마침**을 클릭합니다.

응용 프로그램을 다시 배포한 후에는 조직 또는 프로젝트 요구 사항을 기준으로 응용 프로그램의 유효성을 검사할 수 있습니다.

제 10 장

응용 프로그램 패치 배포

이 장에 포함된 항목:

- [응용 프로그램 패치 배포 개요, 148](#)
- [응용 프로그램 개체 인스턴스, 149](#)
- [Application Object Dependencies, 149](#)
- [충분 배포 마법사, 151](#)
- [응용 프로그램 패치 유형 선택, 153](#)
- [응용 프로그램 개체 영향 결정, 156](#)
- [패치 배포 전략 지정, 160](#)
- [응용 프로그램 패치 배포, 161](#)
- [응용 프로그램 패치 배포의 규칙 및 지침, 162](#)

응용 프로그램 패치 배포 개요

응용 프로그램 패치는 배포된 충분 응용 프로그램의 개체를 업데이트할 때 사용할 수 있는 배포 가능한 개체입니다.

응용 프로그램 패치를 설계하려면 응용 프로그램 개체 간의 종속성과 응용 프로그램 개체의 디자인 타임 및 런타임 인스턴스를 고려하여 생성하려는 응용 프로그램 패치의 유형을 결정해야 합니다. 일부 응용 프로그램 개체는 업데이트하려는 다른 개체와 관련될 수 있으며 업데이트하려는 개체는 다른 응용 프로그램 개체에 영향을 미칠 수 있습니다.

패치 유형을 사용하면 관련 개체를 상속하는 방법을 선택하고 배포된 응용 프로그램에 패치가 미치는 영향의 정도를 결정할 수 있습니다. 패치 상속 및 개체 영향을 관리하려면 충분 배포 마법사의 도구를 사용하여 패치 콘텐츠를 확인하고 데이터 통합 서비스가 배포된 응용 프로그램에 콘텐츠를 적용하는 방법을 파악할 수 있습니다.

참고: 다음과 같은 태스크는 응용 프로그램 패치를 사용하여 수행할 수 없습니다.

- 응용 프로그램 및 매핑 배포 속성(예: 응용 프로그램 설명, 기본 날짜 시간 형식 또는 정렬 순서) 업데이트.
- 배포된 응용 프로그램에 웹 서비스, SQL 데이터 서비스 또는 REST 웹 서비스 추가.
- 배포된 응용 프로그램에서 응용 프로그램 개체 삭제.

응용 프로그램 개체 인스턴스

응용 프로그램의 응용 프로그램 개체에는 디자인 타임 인스턴스와 런타임 인스턴스가 있습니다.

응용 프로그램 개체의 디자인 타임 인스턴스는 개발자가 **Developer tool**에서 생성하고 편집하는 응용 프로그램 개체를 나타냅니다. 개체가 포함된 응용 프로그램을 데이터 통합 서비스에 배포하면 응용 프로그램 개체의 런타임 인스턴스가 생성됩니다. 개체의 런타임 인스턴스는 디자인 타임 인스턴스의 데이터 구조 변경에서 격리되므로 **Developer tool**에서 런타임 개체를 변경하지 않고 디자인 타임 개체를 계속해서 편집할 수 있습니다.

중분 배포 마법사는 응용 프로그램 개체의 디자인 타임 및 런타임 인스턴스를 모두 사용하여 패치 콘텐츠를 배포된 응용 프로그램에 적용하는 방법을 결정합니다. 예를 들어 이 마법사는 선택한 개체의 디자인 타임 및 런타임 인스턴스에 대한 종속성 집합에 따라 패치가 다른 개체를 상속하는 방법을 결정하고 응용 프로그램 개체의 런타임 인스턴스를 가져와 이러한 인스턴스가 어떤 영향을 받는지 결정합니다.

Application Object Dependencies

An application object that you select to include in an application patch might be related to other application objects based on the selected object's set of dependencies.

Objects within an application can contain the following dependencies:

Direct

A direct dependency is an application object that the selected object uses and any direct dependencies of the direct dependency.

The following table lists the direct dependencies for each object type:

Object	Possible Direct Dependencies
Workflow	<ul style="list-style-type: none">- Mapping- Mapplet- Reusable transformation- Logical data object- Physical data object
Mapping	<ul style="list-style-type: none">- Mapplet- Reusable transformation- Logical data object- Physical data object
Mapplet	<ul style="list-style-type: none">- Reusable transformation- Logical data object- Physical data object
Reusable Transformation	<ul style="list-style-type: none">- Logical data object- Physical data object
Logical Data Object	<ul style="list-style-type: none">- Physical data object

Object	Possible Direct Dependencies
Physical Data Object	- A physical data object cannot have a direct dependency.
Parameter Set	- A parameter set cannot have a direct dependency or be a direct dependency.

Indirect

An indirect dependency is an application object that uses the selected object, or an application object that uses a direct dependency of the selected object. Indirect dependencies include any direct dependencies of the indirect dependency.

Remote

A remote dependency is an application object that is an indirect dependency of an indirect dependency for the selected object. Remote dependencies include any direct dependencies of the remote dependency.

An application patch inherits other application objects based on a selected object's dependencies. For example, you can create an application patch that inherits a selected object's direct and indirect dependencies.

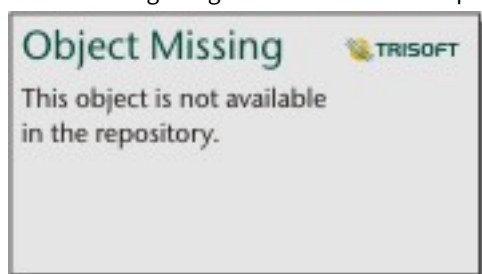
You cannot select mapplets, reusable transformations, and logical data objects in an application patch, but the patch can inherit these objects.

For more information about identifying indirect and remote dependencies, see the following Informatica How-To Library article:

[Identifying Indirect and Remote Dependencies for an Application Patch in the Developer Tool](#)

Example. Dependency Sets

The following image shows the set of dependencies for an application object Mapping A:



The following table describes the set of dependencies for the mapping Mapping A:

Dependency Type	Object	Description
Direct	Reusable Transformation A	The reusable transformation is a direct dependency because the mapping Mapping A uses the reusable transformation.
Direct	Physical Data Object A	The physical data object is a direct dependency because the mapping Mapping A ultimately uses the data object through the reusable transformation Reusable Transformation A .

Dependency Type	Object	Description
Indirect	Workflow A	The workflow is an indirect dependency because the workflow uses the mapping Mapping A .
Indirect	Mapping B	The mapping Mapping B is an indirect dependency because the mapping Mapping B uses a direct dependency of the mapping Mapping A : the reusable transformation Reusable Transformation A .
Indirect	Physical Data Object B	The physical data object is an indirect dependency because it is a direct dependency for the mapping Mapping B which is an indirect dependency for the mapping Mapping A .
Remote	Workflow B	The workflow is a remote dependency because it is an indirect dependency for the mapping Mapping B which is an indirect dependency for the mapping Mapping A .
Remote	Mapping C	The mapping Mapping C is a remote dependency because it is a direct dependency for the workflow Workflow B which is a remote dependency for the mapping Mapping A .

Relationship Priority

Relationships between physical data objects, mappings, and workflows take priority over relationships between other types of objects.

For example, the following image shows a set of dependencies for a data object **Physical Data Object A**:



The mapping **Mapping A** is an indirect dependency because the relationship between the mapping **Mapping A** and the data object **Physical Data Object A** takes priority over the following relationships:

- The relationship between the data object **Physical Data Object A** and the mapplet **Mapplet A**.
- The relationship between the mapplet **Mapplet A** and the mapping **Mapping A**.

Since the mapping **Mapping A** is an indirect dependency, its direct dependency **Mapplet A** is an indirect dependency for the data object **Physical Data Object A**.

증분 배포 마법사

증분 배포 마법사는 응용 프로그램 패치를 생성할 때 사용하는 마법사입니다.

마법사를 실행하려면 증분 응용 프로그램을 마우스 오른쪽 단추로 클릭하고 **응용 프로그램 개체 업데이트**를 선택합니다.

마법사에서 다음 페이지를 사용하여 응용 프로그램 패치를 생성합니다.

1. **응용 프로그램 선택** 페이지에서, 업데이트하려는 런타임 응용 프로그램을 선택합니다.
2. **소스 개체 선택** 페이지에서, 패치에 포함하려는 개체를 선택합니다. 패치가 런타임 응용 프로그램에 미치는 영향의 유형 및 영향의 정도를 미리 볼 수 있습니다.
3. **패치 배포 전략** 페이지에서 패치에 대한 추가적인 세부 정보를 정의하고 패치 배포 방법을 결정합니다.

응용 프로그램 선택

응용 프로그램 선택 페이지에서, 업데이트하려는 런타임 응용 프로그램을 선택합니다.

다음 이미지는 **응용 프로그램 선택** 페이지를 보여 줍니다.



소스 개체 선택

소스 개체 선택 페이지에서, 생성할 패치 유형을 선택하고 패치에 포함할 개체를 선택합니다.

다음 이미지는 **소스 개체 선택** 페이지를 보여 줍니다.



각 개체에 대해 디자인 타임 위치 열을 사용하여 디자인 타임 응용 프로그램의 개체에 대한 경로를 볼 수 있습니다. 마법사의 관련 개체 열에는 패치 유형에 따른 개체의 종속성도 나열됩니다. 예를 들어 직접 및 간접 종속성을 상속하는 패치 유형의 경우 각 개체의 직접 및 간접 종속성이 마법사에 나열됩니다.

페이지의 오른쪽 아래에서 런타임 개체에 미치는 영향의 요약은 미리 볼 수 있습니다. 업데이트 통계에는 수정되거나 영향을 받는 개체의 총 수가 표시됩니다. 추가 통계에는 추가될 개체의 총 수가 표시됩니다. 특정 런타임 개체에 대한 영향을 보려면 응용 프로그램을 미리 봅니다.

다음 이미지는 응용 프로그램 미리 보기를 보여 줍니다.



개체 이름을 일치하고 런타임 위치를 디자인 타임 위치와 일치하여 미리 보기의 런타임 개체 목록을 디자인 타임 개체와 일치할 수 있습니다. 각 개체에 대한 영향에는 개체가 수정되는지, 영향을 받는지, 추가되는지, 변경되지 않는지 여부가 나열됩니다.

패치 배포 전략

패치 배포 전략 페이지에서 패치를 설명하고, 패치 배포 방법을 정의하고, 패치에 상태 정보를 유지할지 선택합니다.

다음 이미지는 **패치 배포 전략** 페이지를 보여 줍니다.



배포 전략을 선택할 때 패치 기록을 통해 배포된 패치를 검토하여 런타임 응용 프로그램을 업데이트할 수도 있습니다.

응용 프로그램 패치 유형 선택

중분 배포 마법사를 사용하는 경우 생성할 응용 프로그램 패치의 유형을 선택할 수 있습니다. 응용 프로그램 패치 유형은 패치가 상속하는 종속성 수준을 결정합니다.

패치는 직접, 간접 및 원격 종속성을 상속할 수 있습니다. 이러한 수준의 종속성이 포함되는 경우 패치가 상속하는 개체가 늘어납니다. 상속된 개체는 패치에 포함되며 패치가 런타임 응용 프로그램에 미치는 전체적인 영향이 증가합니다. 업데이트하려는 응용 프로그램 디자인 및 응용 프로그램 개체에 따라 상속할 종속성을 선택할 수 있습니다.

다음 이미지는 마법사에 표시되는 패치 유형의 목록을 보여 줍니다.

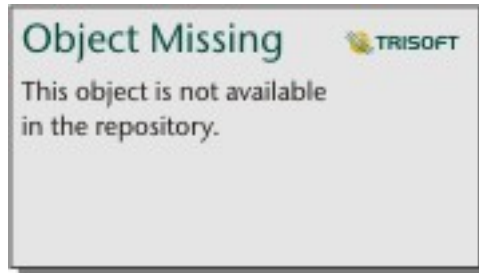


패치를 배포하면 데이터 통합 서비스가 선택한 개체의 런타임 인스턴스 및 런타임 응용 프로그램의 상속된 종속성을 바꾸므로 다른 런타임 개체가 영향을 받을 수 있습니다. 데이터 통합 서비스는 바뀐 런타임 개체를 사용하여 도록 영향을 받는 개체를 업데이트하지만 영향을 받는 개체가 디자인 타임 인스턴스로 바뀌지는 않습니다.

응용 프로그램에 종속성이 높은 개체가 포함되는 경우 응용 프로그램 패치가 상속하는 종속성은 런타임 응용 프로그램의 영향을 받는 개체 수와 반비례할 수 있습니다. 응용 프로그램 패치가 상속하는 종속성의 수가 작으면 런타임 응용 프로그램의 더 많은 개체가 영향을 받을 수 있습니다.

예. 종속성 상속

다음 이미지는 선택한 개체가 Mapping A 매핑인 경우 패치가 상속하는 직접 및 간접 종속성을 보여 줍니다.



패치는 다음과 같은 직접 및 간접 종속성을 상속합니다.

- 직접 종속성. 패치는 선택한 매핑인 Mapping A의 직접 종속성 개체인 Reusable Transformation A 재사용 가능 변환 및 Physical Data Object A 데이터 개체를 상속합니다.
- 간접 종속성. 패치는 선택한 매핑인 Mapping A의 간접 종속성 개체인 Mapping B 매핑 및 Physical Data Object B 데이터 개체를 상속합니다.

패치는 Workflow B 워크플로우 및 Mapping C 매핑을 상속하지 않습니다. 이러한 개체는 선택한 매핑인 Mapping A의 원격 종속성이기 때문입니다.

직접 종속성만 상속

디자인 타임 및 런타임 응용 프로그램에 대한 전체 정보가 없는 경우 런타임 응용 프로그램의 개체에 미치는 영향을 최소화하려면 직접 종속성만 상속하는 패치를 생성합니다.

예

30명의 다른 사용자가 공유하는 응용 프로그램이 있습니다. 여기서 업데이트가 필요한 응용 프로그램 개체는 Kafka 데이터 개체의 데이터를 읽는 단일 스트리밍 매핑입니다. 런타임 응용 프로그램에서 매핑을 테스트한 후 Developer tool에서 매핑을 편집하고 응용 프로그램 패치를 배포하여 런타임 응용 프로그램의 매핑을 업데이트합니다.

응용 프로그램 패치를 생성할 때 매핑을 선택하고 직접 종속성만 상속하도록 선택합니다. 매핑을 선택하는 경우 패치는 매핑에 사용되는 데이터 개체를 상속합니다.

다음 이미지는 증분 배포 마법사에 표시된 상속을 보여 줍니다.



다음 이미지는 선택한 개체와 상속된 간접 종속성을 나타냅니다.



직접 및 간접 종속성 상속

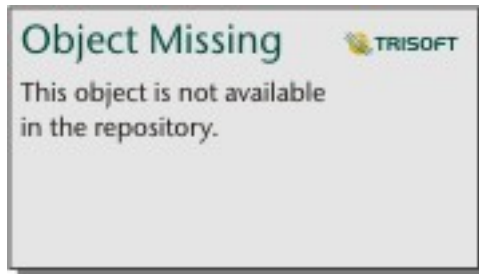
다른 많은 응용 프로그램 개체에 사용되는 응용 프로그램 개체를 편집할 때 변경 내용을 상위 개체에 전파하려면 직접 및 간접 종속성을 상속하도록 선택합니다.

예

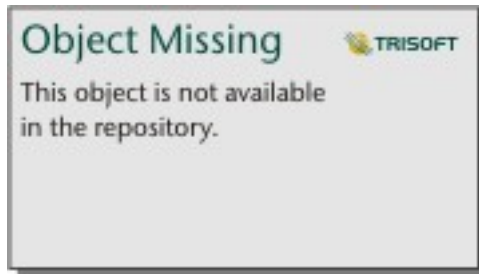
동일한 관계형 데이터 개체를 사용하는 다수의 매핑이 포함된 응용 프로그램이 있습니다. 각 매핑은 다양한 목적으로 데이터 개체의 데이터를 변환합니다. 매핑 실행을 예약하기 위해 데이터 통합 서비스에 응용 프로그램을 배포합니다.

나중에 Developer tool을 사용하여 관계형 데이터 개체를 관계형 데이터베이스와 동기화합니다. 동기화된 데이터를 런타임 응용 프로그램의 매핑에 전파하기 위해 직접 및 간접 종속성을 상속하는 패치를 생성합니다.

패치를 생성할 때 데이터 개체를 사용하는 매핑 중 하나를 선택하고 패치는 이 데이터 개체를 사용하는 다른 모든 매핑을 상속합니다. 다음 이미지는 중분 배포 마법사에 표시된 상속을 보여 줍니다.



다음 이미지는 선택한 개체와 상속된 간접 종속성을 나타냅니다.



직접, 간접 및 원격 종속성 상속

디자인 타임 및 런타임 응용 프로그램에 대한 전체 정보가 있는 경우 직접, 간접 및 원격 종속성을 상속합니다.

응용 프로그램의 단독 개발자이거나 기능 관리자인 경우 응용 프로그램에 대한 전체 정보가 있을 수 있습니다.

디자인 타임 응용 프로그램을 변경하는 경우 모든 변경 내용을 런타임 응용 프로그램에 전파하여 런타임 응용 프로그램이 디자인 타임 응용 프로그램을 미리링하도록 해야 합니다.

예

관리하는 응용 프로그램에는 서로 다른 데이터 개체를 공유하는 매핑을 실행하는 워크플로우가 포함되어 있습니다. 응용 프로그램을 배포하고 출력을 테스트한 후 데이터 개체 중 하나를 편집합니다.

데이터 개체를 사용하는 다른 모든 응용 프로그램 개체로 데이터 개체의 변경 내용을 전파하기 위해 직접, 간접 및 원격 종속성을 상속하는 패치를 생성합니다. 데이터 개체를 선택하면 패치가 데이터 개체를 사용하는 모든 워크플로우 및 매핑을 상속합니다.

다음 이미지는 증분 배포 마법사에 표시된 상속을 보여 줍니다.



다음 이미지는 선택한 개체와 상속된 직접, 간접 및 원격 종속성을 나타냅니다.



참고: 런타임 응용 프로그램에서 **Physical Data Object A** 데이터 개체를 사용하는 매핑 및 워크플로우로 데이터 개체의 변경 내용을 전파하려면 직접 종속성만 상속하는 패치를 사용할 수도 있습니다. 데이터 개체를 선택하면 마법사가 영향을 받는 개체가 되는 매핑 및 워크플로우를 식별하고 데이터 통합 서비스는 패치 배포 중에 이러한 매핑 및 워크플로우를 업데이트합니다.

그러나 직접, 간접 및 원격 종속성을 상속하는 패치는 개체가 업데이트되는 방법에 대한 개선된 투명성을 제공합니다. 이러한 패치를 사용하는 경우 런타임 응용 프로그램의 업데이트된 개체가 디자인 타임 응용 프로그램의 개체와 동일한 방법으로 데이터를 변환할 것이라는 보장이 더 높아집니다.

응용 프로그램 개체 영향 결정

응용 프로그램 패치를 생성하고 배포하는 경우 패치가 런타임 응용 프로그램에 미치는 영향은 수정된 개체와 영향을 받는 모든 개체로 확장됩니다. 여러 사용자가 응용 프로그램을 공유하는 경우 패치는 다른 사용자가 유지 관리하는 개체에 영향을 미칠 수 있습니다.

패치가 런타임 응용 프로그램에 미치는 영향의 정도를 파악하려면 다음과 같은 작업을 수행합니다.

- 개체 영향 미리 보기. 개체 영향은 각 응용 프로그램 개체가 받는 영향을 설명하는 요약입니다.
- 영향을 받는 개체 식별. 영향을 받는 개체는 다른 사용자에게 속하는 개체일 수 있습니다. 영향을 받는 개체를 식별하여 다른 사용자의 개체가 어떻게 변경되는지 확인할 수 있습니다.
- 패치의 범위 검토. 범위는 패치의 영향을 받는 개체의 수에 대한 숫자 요약입니다.

개체 영향 미리 보기

응용 프로그램 패치에 개체가 추가되고 패치가 개체의 종속성을 상속하는 경우 증분 배포 마법사는 런타임 응용 프로그램에 대한 영향을 분석합니다. 각 응용 프로그램 개체에 대한 영향을 미리 보면 패치가 배포된 후 개체의 런타임 인스턴스가 어떻게 변경되는지 확인할 수 있습니다.

미리 보기에는 각 응용 프로그램 개체에 대한 다음과 같은 유형의 영향이 나열됩니다.

- 수정. 런타임 응용 프로그램에 개체의 런타임 인스턴스가 있으며 데이터 통합 서비스는 런타임 인스턴스를 디자인 타임 인스턴스로 바꿉니다.
- 영향받음. 런타임 응용 프로그램의 개체에 대한 런타임 종속성이 수정될 예정이므로 마법사 런타임 응용 프로그램에서 개체를 가져옵니다. 데이터 통합 서비스는 수정된 종속성을 사용하도록 런타임 응용 프로그램의 영향을 받는 개체를 업데이트하지만 바꾸지는 않습니다.
- 추가. 런타임 응용 프로그램에 개체의 런타임 인스턴스가 없으므로 데이터 통합 서비스는 런타임 응용 프로그램에 개체의 런타임 인스턴스를 생성합니다.
- 변경 없음. 개체는 수정되거나 영향을 받거나 런타임 응용 프로그램에 추가되지 않습니다.

다음 이미지는 Developer tool에서 개체 영향을 미리 볼 수 있는 위치를 보여 줍니다.



영향을 받는 개체 식별

영향을 받는 개체는 증분 배포 마법사가 배포된 응용 프로그램의 유지 관리를 위해 런타임 응용 프로그램에서 가져오는 개체입니다. 영향을 받는 개체를 소유한 사용자는 데이터 통합 서비스가 패치를 적용하는 동안 개체를 실행할 수 없게 됩니다.

마법사는 응용 프로그램 패치가 디자인되는 동안 영향을 받는 개체를 가져옵니다. 개체 영향을 미리 보면 영향을 받는 개체를 볼 수 있습니다.

마법사는 다음과 같은 이유로 영향을 받는 개체를 가져옵니다.

변경 내용 전파

사용자가 응용 프로그램 패치에서 개체를 선택하고 패치가 다른 응용 프로그램 개체를 상속하는 경우 마법사는 개체의 변경 내용을 런타임 응용 프로그램의 상위 개체로 전파할 방법을 결정해야 합니다. 마법사는 변경으로 인해 영향을 받는 상위 개체를 식별합니다.

패치가 배포되는 경우 데이터 통합 서비스는 영향을 받는 개체의 런타임 인스턴스를 바꾸지 않고 런타임 직접 종속성에 수행된 변경을 사용하도록 영향을 받는 개체를 업데이트합니다.

참고: 영향을 받는 개체가 워크플로우인 경우 워크플로우는 다음번 실행 시 업데이트됩니다. 예를 들어 매핑이 수정되고 워크플로우가 이 매핑을 매핑 태스크에서 사용하는 경우 워크플로우는 이 수정된 매핑을 후속 실행에서 사용합니다.

유효성

마법사는 런타임 응용 프로그램의 응용 프로그램 개체가 데이터 통합 서비스의 패치 적용 후에 유효한지 확인해야 합니다. 마법사는 영향을 받는 개체에 변경 내용을 전파할 방법을 결정한 후 영향을 받는 개체의 유효성을 검사합니다.

효성을 검사합니다. 영향을 받는 개체가 수정된 개체와 호환되지 않는 경우 이 개체는 유효하지 않을 수 있습니다.

응용 프로그램 패치 유형에 따라 영향을 받는 개체가 발생할 수 있습니다. 예를 들어 직접, 간접 및 원격 종속성을 상속하는 패치를 생성하는 경우 원격 종속성에 패치가 상속하는 직접 종속성이 포함될 수 있습니다. 이 직접 종속성이 런타임 응용 프로그램의 다른 개체에 사용되는 경우 상위 개체는 영향을 받는 개체가 됩니다.

데이터 통합 서비스가 영향을 받는 개체를 업데이트한 후 영향을 받는 개체의 데이터는 응용 프로그램 디자인과 일치하지 않을 수 있습니다. 예를 들어 맵셋을 사용하는 매핑이 포함된 런타임 응용 프로그램에서 수정된 맵셋에 다른 변환이 사용되는 경우 영향을 받는 매핑은 유효할 수 있지만 데이터를 예측할 수 없는 방식으로 변환할 가능성도 있습니다.

가능한 경우 업데이트하려는 개체를 선택하거나 개체를 상속하는 응용 프로그램 패치를 설계하여 영향을 받는 개체의 수를 최소화하고 런타임 응용 프로그램의 개체가 디자인 타임 응용 프로그램의 개체와 동일한 방식으로 데이터를 변환할 수 있도록 하십시오.

예. 변경 내용 전파 및 유효성

데이터 개체를 수정하는 패치를 배포하는 경우 마법사는 데이터 개체를 사용하는 다른 모든 런타임 개체로 데이터 개체의 수정 사항을 전파합니다. 데이터 개체를 사용하는 영향을 받는 런타임 매핑은 데이터 통합 서비스가 패치를 적용한 후 수정된 데이터 개체를 사용하게 됩니다.

매핑에 수정된 데이터 개체가 사용되고 데이터 개체에 포함된 행 수가 패치 적용 전보다 작은 경우 매핑은 더 이상 유효하지 않을 수 있습니다. 이 데이터 개체는 업스트림 및 다운스트림 변환 간의 링크 수가 작기 때문에 매핑의 변환과 호환되지 않을 수 있습니다. 영향을 받는 매핑이 유효하지 않은 경우 패치를 배포할 수 없습니다.

삭제된 개체

디자인 타임 응용 프로그램에서 응용 프로그램 개체를 삭제한 후 패치를 배포하는 경우 데이터 통합 서비스는 런타임 응용 프로그램에서 해당 개체를 삭제하지 않습니다. 일부 경우 삭제된 개체는 영향을 받는 개체일 수 있습니다.

삭제된 개체가 영향을 받는 개체로 식별되는 경우 마법사는 삭제된 개체의 하위 개체에 포함된 변경 내용을 삭제된 개체로 전파합니다. 또한 마법사는 삭제된 개체의 유효성을 검사합니다. 삭제된 개체가 유효하지 않은 경우 패치를 배포할 수 없습니다.

사용자가 삭제된 개체를 실행하고 있지 않은 경우 응용 프로그램을 재배포하여 런타임 응용 프로그램에서 개체를 제거할 수 있습니다.

예제. 영향을 받는 개체 식별

다음 예는 증분 배포 마법사가 런타임 응용 프로그램 개체에서 가져오는, 영향을 받는 개체를 식별하는 방법을 보여 줍니다.

예를 들어 Mapping A와 Mapping B의 두 매핑이 포함된 응용 프로그램이 있습니다. 이러한 매핑은 물리적 데이터 개체 Data Object A와 워크플로우 Workflow B를 공유하며 워크플로우에는 Mapping B 매핑이 사용됩니다. 응용 프로그램을 배포한 후 Mapping A 매핑을 편집하고 런타임 응용 프로그램의 매핑을 업데이트하기로 결정합니다.

직접 종속성만 상속하는 패치를 생성하고 Mapping A 매핑을 선택합니다. 매핑을 선택했으므로 패치는 매핑에 사용되는 데이터 개체인 Data Object A를 상속합니다.

마법사는 데이터 통합 서비스가 매핑 또는 데이터 개체의 런타임 인스턴스를 바꿀 것으로 예상하므로 런타임 응용 프로그램에서 이 매핑 및 데이터 개체를 사용하는 모든 상위 개체를 찾습니다. 데이터 개체는 Mapping B 매핑에 데이터 개체가 사용되므로 마법사는 이 매핑을 영향을 받는 개체로 표시합니다.

Mapping B 매핑은 런타임 응용 프로그램에서 업데이트될 것이고 Workflow B 워크플로우에는 Mapping B 매핑이 사용되므로 Workflow B 워크플로우 또한 영향을 받는 개체입니다.

다음 이미지는 디자인 타임 및 런타임 조건, 선택한 매핑, 패치가 상속하는 종속성 및 영향을 받는 개체를 보여 줍니다.



참고: Mapping B 및 Workflow B 개체의 런타임 인스턴스는 디자인 타임 인스턴스로 바뀌지 않습니다. 디자인 타임 응용 프로그램에서 Mapping B 매핑을 편집한 경우 Mapping B의 런타임 인스턴스는 디자인 타임 인스턴스와 다릅니다.

개체 영향의 범위 식별

응용 프로그램 패치가 배포된 응용 프로그램에 미치는 영향의 범위를 식별하여 패치 배포 중에 가동 중지를 경험하는 개체의 수를 확인합니다. 응용 프로그램의 사용자가 여러 명인 경우 범위가 클수록 이러한 사용자가 사용할 수 있는 개체의 수가 줄어듭니다.

범위를 식별하려면 다음과 같은 개체 유형에 따라 영향 요약의 값을 사용합니다.

- 선택한 개체. 패치에 표시하도록 선택하는 개체와 패치가 상속하는 개체의 합계입니다. 최대값은 디자인 타임 응용 프로그램의 응용 프로그램 개체 수입입니다.
- 업데이트된 개체. 수정되거나 영향을 받는 개체의 합계입니다. 최대값은 런타임 응용 프로그램의 응용 프로그램 개체 수입입니다.
- 추가된 개체. 런타임 응용 프로그램에 추가될 개체 수입입니다.

참고: 디자인 타임 응용 프로그램에 런타임 응용 프로그램과 다른 종속성 집합이 포함되는 경우 선택한 개체의 수는 업데이트된 개체의 수와 다를 수 있습니다.

다음 이미지는 증분 배포 마법사에 표시되는 요약을 보여 줍니다.



패치 배포 전략 지정

조직의 정책에 따라 응용 프로그램 패치의 콘텐츠를 최종 사용자에게 제공하는 방법을 결정하려면 패치 배포 전략을 지정합니다.

다음과 같은 방법으로 응용 프로그램 패치를 배포할 수 있습니다.

- 데이터 통합 서비스에 배포. 패치를 데이터 통합 서비스에 배포하면 데이터 통합 서비스가 패치 콘텐츠에 따라 증분 응용 프로그램을 업데이트합니다. 영향을 받지 않는 개체는 계속해서 실행되며 영향을 받는 개체는 패치 콘텐츠로 업데이트됩니다.
- 보관 파일에 배포. 패치를 보관 파일에 배포하면 패치를 로컬 시스템에 파일로 저장하고 파일을 버전 제어 시스템에 저장한 다음 나중에 데이터 통합 서비스에 배포할 수 있습니다.

패치를 배포하는 경우 응용 프로그램의 상태는 변경되지 않습니다. 예를 들어 중지된 응용 프로그램은 실행을 시작하지 않습니다.

배포를 취소하는 경우 데이터 통합 서비스가 패치 콘텐츠를 배포된 응용 프로그램에 부분적으로 적용할 수 있습니다. 배포를 취소하는 경우 증분 응용 프로그램은 롤백되지 않습니다.

패치 배포 전략 속성

패치 배포 전략 페이지에서 일반 속성과 배포 옵션을 구성할 수 있습니다. 또한 런타임 응용 프로그램의 패치 기록을 검토할 수도 있습니다.

다음 테이블에는 구성할 수 있는 일반 속성이 설명되어 있습니다.

일반 속성	설명
패치 이름	패치의 이름입니다.
패치 설명	선택 사항입니다. 패치에 대한 설명입니다. 패치 설명은 270자를 초과할 수 없습니다.
상태 정보 유지	선택 사항입니다. 상태 정보를 유지할지 아니면 삭제할지를 나타냅니다. 상태 정보란 매핑 출력이나 시퀀스 생성기 변환과 같은 런타임 개체의 속성 및 매핑 속성을 가리킵니다. 기본적으로 상태 정보는 유지됩니다. 참고: 응용 프로그램 상태를 유지하는 삭제하는 시퀀스 생성기 변환의 재설정 속성에는 영향을 주지 않습니다. 상태 정보에 대한 자세한 내용은 “응용 프로그램 상태 정보” 페이지 141 항목을 참조하십시오.

다음 테이블에는 선택 가능한 패치 배포 옵션이 설명되어 있습니다.

배포 옵션	설명
데이터 통합 서비스에 배포	패치를 데이터 통합 서비스에 배포하려면 이 옵션을 선택합니다.
파일에 배포	패치를 응용 프로그램 패치 보관 파일에 배포하려면 이 옵션을 선택합니다.

다음 테이블에는 패치 기록에서 각 패치에 대해 검토할 수 있는 속성이 설명되어 있습니다.

패치 속성	설명
이름	배포된 패치의 이름입니다.
설명	배포된 패치에 대한 설명입니다. 패치가 생성된 시간이 패치 설명의 시작 부분에 추가됩니다.

참고: 기본적으로 패치 기록은 패치가 생성된 시간별로 패치를 나열합니다.

응용 프로그램 패치 배포

중분 응용 프로그램의 개체를 편집하는 경우 응용 프로그램 패치를 생성하고 배포하여 배포된 응용 프로그램의 개체를 업데이트할 수 있습니다. 그런 다음 응용 프로그램 패치를 데이터 통합 서비스 또는 응용 프로그램 패치 보관 파일에 배포합니다.

1. **개체 탐색기** 보기에서 중분 응용 프로그램을 마우스 오른쪽 단추로 클릭하고 **응용 프로그램 개체 업데이트**를 선택합니다.

중분 배포 마법사의 **응용 프로그램 선택** 페이지가 나타납니다.



2. **응용 프로그램 선택** 페이지에서 **찾아보기**를 클릭하고 데이터 통합 서비스에서 업데이트할 런타임 응용 프로그램을 선택합니다.
3. **다음**을 클릭합니다.

마법사가 런타임 응용 프로그램을 가져옵니다. 그런 다음 **소스 개체 선택** 페이지가 나타납니다.



4. **소스 개체 선택** 페이지에서 데이터 통합 서비스를 통해 런타임 응용 프로그램에서 바꾸려는 디자인 타임 응용 프로그램의 개체를 선택합니다.
5. 다음 응용 프로그램 패치 유형 중 하나를 선택합니다.
 - 직접 종속성. 패치는 선택한 개체의 직접 종속성을 상속합니다.
 - 직접 및 간접 종속성. 패치는 선택한 개체의 직접 및 간접 종속성을 상속합니다.

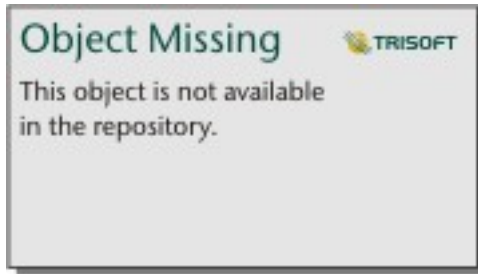
- 직접, 간접 및 원격 종속성. 패치는 선택한 개체의 직접, 간접 및 원격 종속성을 상속합니다.

기본값은 직접 및 간접 종속성입니다.

종속성은 **관련 개체** 열에 나타납니다. 예를 들어 직접 및 간접 종속성을 상속하는 응용 프로그램 패치의 경우 선택한 개체의 직접 및 간접 종속성이 이 열에 나열됩니다.

6. 런타임 응용 프로그램의 각 개체에 미치는 영향을 검토하려면 **응용 프로그램 미리 보기**를 클릭합니다.
7. **다음**을 클릭합니다.

패치 배포 전략 페이지가 나타납니다.



8. 패치 이름을 입력하십시오.
9. 필요에 따라 패치에 대한 설명을 입력합니다.
10. 필요에 따라 상태 정보를 유지할지 여부를 선택합니다.
상태 정보를 유지하지 않기로 선택하면 상태 정보가 삭제됩니다.
11. 다음 배포 옵션 중 하나를 선택합니다.
 - 데이터 통합 서비스에 배포. 데이터 통합 서비스에 응용 프로그램 패치를 배포합니다. 데이터 통합 서비스는 런타임 응용 프로그램에 응용 프로그램 패치를 적용합니다.
 - 파일에 배포. 보관 파일에 응용 프로그램 패치를 배포합니다. 보관 파일에 패치를 배포하는 경우 파일을 저장하려는 로컬 시스템의 디렉터리를 선택합니다.
12. 패치 기록을 검토합니다.
패치 기록에는 증분 응용 프로그램을 업데이트하기 위해 배포된 응용 프로그램 패치가 나열되어 있습니다.

응용 프로그램 패치 배포의 규칙 및 지침

응용 프로그램 패치를 배포하여 증분 응용 프로그램을 업데이트하는 경우 다음 규칙 및 지침을 참조하십시오.

- 개체 탐색기 보기에서 응용 프로그램 개체를 업데이트하도록 선택하면 증분 배포 마법사가 응용 프로그램 종속성을 구성하기 시작합니다. 진행률 보기에서 진행률을 볼 수 있지만 작업을 취소할 수는 없습니다.
- 패치를 생성하는 경우 응용 프로그램의 개체가 아닌 개체는 증분 배포 마법사에 나열되지 않습니다.
예를 들어 응용 프로그램에 매핑은 포함되지만 매핑이 데이터 변환에 사용하는 데이터 개체는 제외될 수 있습니다. 이 경우 데이터 개체는 응용 프로그램 개체가 아닙니다. 패치를 생성하면 데이터 개체를 볼 수 없게 되지만 마법사는 응용 프로그램 개체 간의 종속성을 식별할 때 데이터 개체를 처리합니다.
- 디자인 타임 응용 프로그램에서 응용 프로그램 개체의 이름 또는 프로젝트를 변경하고 응용 프로그램 패치를 사용하여 개체를 업데이트하면 런타임 응용 프로그램에서 서로 다른 경로 아래 개체가 중복됩니다.
- 패치를 처음으로 배포할 때 런타임 응용 프로그램의 응용 프로그램 설명이 비어 있으면 증분 배포 마법사가 응용 프로그램의 패치 기록을 사용하여 설명을 업데이트합니다. 추가 패치를 배포하면 마법사가 설명에 패치 이름을 추가합니다. 마법사는 클라이언트를 사용하여 응용 프로그램 설명이 편집되기 전까지 패치 이름을 추가합니다.

제 11 장

응용 프로그램 패치 배포 예

이 장에 포함된 항목:

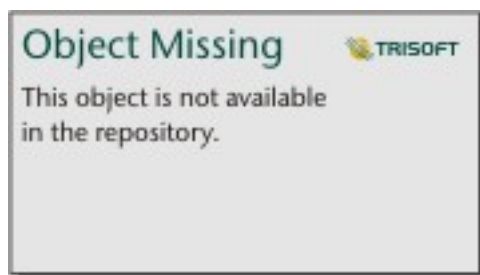
- [응용 프로그램 패치 배포 예 개요, 163](#)
- [패치 유형: 직접, 164](#)
- [패치 유형: 직접 및 간접, 165](#)
- [패치 유형: 직접, 간접 및 원격, 166](#)
- [결론, 167](#)

응용 프로그램 패치 배포 예 개요

디자인 타임 개체의 변경 내용을 관찰하고 변경 내용을 데이터 통합 서비스에 배포된 응용 프로그램의 해당 런타임 개체에 병합하는 DevOps 엔지니어로 가정하겠습니다.

최근 개발자가 증분 응용 프로그램에서 매핑 Mapping A를 편집했습니다. 런타임 응용 프로그램의 매핑을 업데이트하는 패치를 생성하려고 합니다.

다음 이미지는 매핑 Mapping A가 선택된 개체인 Developer tool의 디자인 타임 응용 프로그램과 데이터 통합 서비스의 런타임 응용 프로그램을 보여 줍니다.



응용 프로그램 패치를 생성하려면 패치가 상속할 종속성을 선택해야 합니다. 상속된 개체를 식별하고 개체 영향을 평가하고 배포된 응용 프로그램에 대한 변경 내용의 범위를 평가해야 합니다.

패치 유형: 직접

이 섹션에서는 직접 종속성만 상속하는 응용 프로그램 패치에 대해 설명합니다.

상속된 개체 식별

다음 이미지는 직접 종속성만 상속할 때 패치에 포함하는 개체를 보여 줍니다.



다음 테이블에는 매핑 Mapping A가 선택된 개체일 때 패치가 상속하는 종속성이 나열되어 있습니다.

종속성의 유형	종속성
직접	- Data Object A

개체 영향 평가

다음 이미지는 패치가 배포된 후 런타임 개체에 대한 영향을 보여 줍니다.



변경 내용의 범위 평가

다음 테이블은 패치 배포 동안 영향을 받는 개체 수를 보여 줍니다.

범위 항목	개체 수
선택	2
업데이트됨	수정됨: 2 영향받음: 3* 합계: 5*
추가됨	0

* 재사용 가능 변환 Lookup B는 런타임 시 영향을 받는 개체로 동작하지만 중분 배포 마법사는 재사용 가능 변환을 영향을 받는 개체로 나열하지 않습니다. 중분 배포 마법사에서 영향을 받는 개체 수는 2입니다.

결과

직접 종속성만 상속하는 패치는 런타임 응용 프로그램에 대한 영향이 낮지만 런타임 응용 프로그램에는 영향을 받는 개체가 포함되어 있습니다. 영향을 받는 개체는 수정된 개체에 비해 예측 불가능할 수 있으므로 패치를 적용한 후 응용 프로그램이 데이터를 변환하는 방법이 원활하지 않을 수 있습니다.

패치 유형: 직접 및 간접

이 섹션에서는 직접 및 간접 종속성을 상속하는 응용 프로그램 패치에 대해 설명합니다.

상속된 개체 식별

다음 이미지는 직접 및 간접 종속성을 상속할 때 패치에 포함하는 개체를 보여 줍니다.



다음 테이블에는 매핑 Mapping A가 선택된 개체일 때 패치가 상속하는 종속성이 나열되어 있습니다.

종속성의 유형	종속성
직접	- Data Object A
간접	- Mapping B - Data Object B

개체 영향 평가

다음 이미지는 패치가 배포된 후 런타임 개체에 대한 영향을 보여 줍니다.



패치가 매핑 Mapping B를 간접 종속성으로 상속하기 때문에 매핑은 해당 디자인 타임 종속성을 사용하고 이전 런타임 종속성을 삭제합니다. 더 이상 런타임 응용 프로그램에서 재사용 가능 변환 Lookup B를 사용하지 않습니다. 대신 데이터 개체 Data Object A 및 Data Object B를 사용합니다. 변환 Lookup B는 런타임 응용 프로그램에서 고아가 됩니다.

변경 내용의 범위 평가

다음 테이블은 패치 배포 동안 영향을 받는 개체 수를 보여 줍니다.

범위 항목	개체 수
선택	4
업데이트됨	수정됨: 3 영향받음: 1 합계: 4
추가됨	1

결과

직접 및 간접 종속성 상속은 직접 종속성만 상속에 유사한 영향을 미치지만 더 적은 개체가 영향을 받습니다. 매핑 Mapping B는 영향을 받는 대신 수정되므로 보다 예측 가능한 방식으로 업데이트됩니다.

패치 유형: 직접, 간접 및 원격

이 섹션에서는 직접, 간접 및 원격 종속성을 상속하는 응용 프로그램 패치에 대해 설명합니다.

상속된 개체 식별

다음 이미지는 직접, 간접 및 원격 종속성을 상속할 때 패치에 포함하는 개체를 보여 줍니다.

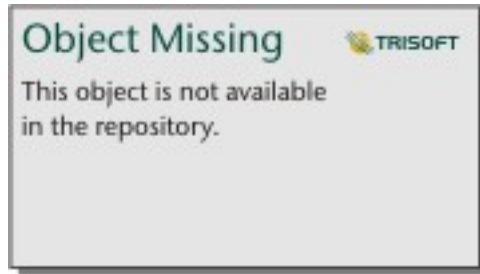


다음 테이블에는 매핑 Mapping A가 선택된 개체일 때 패치가 상속하는 종속성이 나열되어 있습니다.

종속성의 유형	종속성
직접	- Data Object A
간접	- Mapping B - Data Object B
원격	- Workflow B - Mapping C - Data Object C

개체 영향 평가

다음 이미지는 패치가 배포된 후 런타임 개체에 대한 영향을 보여 줍니다.



패치가 매핑 Mapping B 및 Mapping C를 상속하기 때문에 매핑은 런타임 응용 프로그램에서 해당 디자인 타임 종속성을 사용합니다. 재사용 가능 변환 Lookup B 및 데이터 개체 Data Object B는 런타임 응용 프로그램에서 고아가 됩니다.

변경 내용의 범위 평가

다음 테이블은 패치 배포 동안 영향을 받는 개체 수를 보여 줍니다.

범위 항목	개체 수
선택	7
업데이트됨	수정됨: 5 영향받음: 0 합계: 5
추가됨	2

결과

직접, 간접 및 원격 종속성을 상속하는 패치를 생성하는 경우 런타임 응용 프로그램에 대한 변경 내용이 투명합니다. 런타임 응용 프로그램의 업데이트 및 추가된 개체가 디자인 타임 응용 프로그램의 선택된 개체와 일치하며 런타임 응용 프로그램이 데이터를 디자인 타임 응용 프로그램의 개체와 동일한 방식으로 데이터를 변환하도록 기대할 수 있습니다.

트레이드 오프의 모든 개체에서 다운타임이 발생한다는 것이며 응용 프로그램을 재배포함으로써 상황이 나아질 수 있습니다. 응용 프로그램을 재배포하는 경우 런타임 응용 프로그램에서 재사용 가능 변환 Lookup B 및 데이터 개체 Data Object D의 고아 개체 제거와 같은 이점과 함께 동일한 개체를 업데이트합니다.

결론

이 예에서 직접 및 간접 종속성을 상속하는 패치 생성은 런타임 응용 프로그램에 최소의 영향을 미치므로 개체 가동 시간을 극대화합니다. 이를 통해 증분 응용 프로그램의 기능을 최대한 활용할 수 있습니다.

응용 프로그램이 개발 또는 QA 환경에서 배포된 경우 직접 및 간접 종속성을 상속하는 패치를 생성하도록 선택할 수 있습니다. 예를 들어 엔터프라이즈 데이터 처리를 시작하기 전에 샘플 데이터를 사용하여 선택된 매핑 Mapping A가 실행되었는지 테스트하고자 할 수 있습니다.

응용 프로그램이 UAT(사용자 수용 테스트) 또는 프로덕션 환경에 배포되었으며 응용 프로그램을 중지할 수 있는 경우 대신 응용 프로그램을 재배포하도록 선택할 수 있습니다. 응용 프로그램을 재배포하면 데이터가 요구 사항에 따라 변환되도록 보장하도록 런타임 응용 프로그램에서 가장 투명한 변경 내용이 생성됩니다.

제 12 장

CI/CD(지속적인 통합 및 지속적인 전달)

이 장에 포함된 항목:

- [CI/CD 개요, 169](#)
- [지속적인 통합, 170](#)
- [지속적인 전달, 171](#)
- [자동화된 통합 프로세스 샘플, 172](#)
- [자동화된 전달 프로세스 샘플, 174](#)
- [일반적인 통합 및 전달 작업 참조, 174](#)

CI/CD 개요

CI/CD(지속적인 통합 및 지속적인 전달)는 CI/CD 파이프라인에서 통합 및 전달 작업을 자동화하는 방법입니다. 데이터 통합 서비스 REST API 또는 `infacmd` 명령줄 프로그램을 사용하여 각 통합 및 전달 작업을 자동화할 수 있습니다.

CI/CD 파이프라인에는 개발자가 개체를 디자인하는 데 사용하는 통합 작업과 개체를 프로덕션 환경에 전달하는 전달 작업이 포함됩니다. REST API 또는 `infacmd`를 사용하여 다음과 같은 방식으로 통합과 전달을 자동화할 수 있습니다.

- 개발자가 개체에 대해 수행한 모든 변경을 배포하고 테스트합니다. 개발자는 개체가 테스트를 통과했는지 여부 그리고 개체에 필요한 변경 유형에 대한 즉각적인 피드백을 받습니다.
- 테스트를 통과한 개체를 프로덕션 환경에 전달합니다. 조직의 요구 사항에 따라, 프로덕션 환경에 개체를 전달하기 전에 QA 및 UAT와 같은 추가 요구 사항에 개체를 전달할 수 있습니다.

지속적인 통합과 지속적인 전달은 완전하게 기록되고 전체 팀에 표시되므로 팀원 모두가 수작업에서 벗어나 더 많은 시간을 확보할 수 있습니다.

REST API 또는 infacmd 사용

통합 및 전달 작업의 자동화를 위해 데이터 통합 서비스 REST API를 사용할지 아니면 infacmd 명령줄 프로그램을 사용할지를 결정하려면 조직의 요구 사항을 평가합니다.

REST API와 infacmd에는 다음과 같은 차이점이 있습니다.

- REST API에 비해 infacmd를 사용할 수 있는 경우가 더 많습니다. 예를 들어 응용 프로그램 백업에 infacmd 명령을 사용할 수 있습니다.
- infacmd를 사용하려면 Informatica 서비스 또는 infacmd 유틸리티를 설치해야 합니다. 반면 REST API에는 특정 설치 요구 사항이 없기 때문에 클라이언트 시스템에 Informatica 서비스가 설치되어 있지 않은 환경에서 REST API를 실행할 수 있습니다.

데이터 통합 서비스 REST API에 대한 자세한 내용은 *Informatica 응용 프로그램 서비스 가이드*를 참조하십시오.

infacmd 명령줄 프로그램에 대한 자세한 내용은 *Informatica 명령 참조*를 참조하십시오.

지속적인 통합

개체를 승인하여 추가 환경에 전달하기 전에 지속적인 통합을 통해 개발자가 개체에 대해 수행하는 모든 변경을 개발 환경에서 배포하고 테스트할 수 있습니다.

지속적인 통합을 설정하기 위해 다음 태스크를 자동화할 수 있습니다.

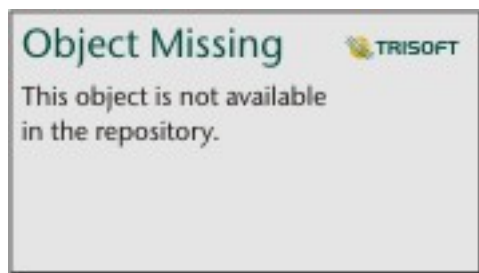
- 개체를 배포하여 온전성 테스트를 실행합니다.
- 버전 제어 작업을 실행하여 개체를 개발자에게 할당합니다.
- 개체 잠금에 따른 충돌을 검색하고 해결합니다.

개체 배포

지속적인 통합을 설정하여 개발자가 개체에 대해 수행한 모든 변경이 배포되고 테스트되었는지 확인합니다.

테스트를 위해 개체를 격리하려면 개체를 응용 프로그램 보관 파일에 배포한 다음 데이터 통합 서비스에 배포합니다. 개체가 데이터 통합 서비스에 배포되면 개발자는 테스트 결과를 방해하지 않고 Developer tool에서 개체에 계속 액세스하고 편집할 수 있습니다.

다음 이미지는 개체가 배포되는 방식을 보여 줍니다.



1. 자동화된 프로세스가 개체를 응용 프로그램 패치 보관 파일에 배포합니다.
2. 자동화된 프로세스가 보관 파일을 데이터 통합 서비스에 배포합니다.

개체가 배포되면 온전성 테스트를 실행하여 매핑이 성공적으로 실행되는지 확인할 수 있습니다. 추가 테스트를 수행하려는 경우 타사 도구를 사용하여 매핑 출력이 올바른지 확인할 수 있습니다. 테스트 결과를 기반으로, 추가 환경에 전달할 수 있도록 개체를 승인하거나 편집을 위해 개체를 개발자에게 할당할 수 있습니다.

개체 할당

팀 기반 개발 환경에서, 지속적인 통합을 설정하여 개발자에게 개체를 할당할 수 있습니다.

예를 들어 매핑이 온전성 테스트를 통과하지 못한 경우 해당 매핑을 마지막으로 수정한 개발자에게 매핑을 할당하여 편집하도록 할 수 있습니다. 매핑을 할당하려면 버전 제어 작업 수행 REST API 또는 `infacmd mrs reassignCheckedOutObject` 명령을 사용합니다.

특정 개발자에게 개체를 할당하거나 한 개발자에서 다른 개발자로 개체를 재할당하려면 버전 제어 작업을 수행할 수 있어야 합니다. 모델 리포지토리가 GIT, SVN 또는 Perforce와 같은 버전 제어 시스템과 통합되어 있는지 확인하십시오.

모델 리포지토리를 버전 제어 시스템과 통합하는 방법에 대한 자세한 내용은 *Informatica 응용 프로그램 서비스 가이드*를 참조하십시오.

개체 충돌 해결

개체를 변경해야 하는 경우 의도 기반 잠금을 사용하여 개체 사이의 충돌을 검색 및 해결하고 개발자 협업을 촉진하도록 지속적인 통합을 설정할 수 있습니다.

의도 기반 잠금은 다른 개발자가 개체에 대한 변경 내용을 저장하지 못하도록 방지합니다. 때때로 잠금은 동일한 개체에 액세스해야 하는 개발자 간에 충돌을 야기합니다.

예를 들어 한 개발자가 휴가를 떠나기 전에 개체를 잠갔다고 가정합니다. 이 잠금은 동일 개체에 액세스해야 하는 다른 개발자의 계획과 충돌합니다. 그러한 충돌을 해결하기 위해 타사 도구를 사용하여 직원의 작업 일정을 읽고, 잠재적 충돌을 검색하고, 잠긴 개체를 나열하는 `infacmd` 명령에 논리를 전달하고, 필요한 경우 개체의 잠금을 해제할 수 있습니다.

지속적인 전달

개체가 개발 환경에서 테스트를 통과하면 해당 개체는 프로덕션 환경에 지속적으로 전달됩니다. 조직의 요구 사항에 따라, 프로덕션 환경에 개체를 전달하기 전에 QA 및 UAT와 같은 추가 환경에 개체가 지속적으로 전달되도록 설정할 수 있습니다.

지속적인 전달을 단순화하기 위해 보관 파일을 사용하여 개발 환경에서 테스트되고 승인된 개체와 추가 환경에 전달되는 개체 간의 일관성을 생성할 수 있습니다. 일관성을 생성하려면 지속적인 전달을 설정하여 개체를 하나의 보관 파일에 배포하고, 보관 파일을 버전 제어 시스템에 저장하고, 보관 파일을 각 환경에 배포합니다.

다음 이미지는 하나의 응용 프로그램 패치 보관 파일을 어떤 방식으로 개발, QA, UAT 및 프로덕션 환경에 배포할 수 있는지 보여 줍니다.



1. 개체가 응용 프로그램 패치 보관 파일에 배포됩니다.
2. 보관 파일은 버전 제어 시스템에 저장됩니다.
3. 동일한 보관 파일이 개발, QA, UAT 및 프로덕션 환경에 배포됩니다.

자동화된 통합 프로세스 샘플

이 섹션은 개발 환경에서 개발자가 수행하는 매핑에 대한 모든 변경을 배포하고 테스트하기 위해 지속적인 통합을 설정하는 방법을 보여 줍니다.

지속적인 통합을 설정하여 다음과 같은 태스크를 수행합니다.

1. 배포를 위해 매핑을 준비합니다.
2. 매핑을 데이터 통합 서비스에 배포합니다.
3. 온전성 테스트를 실행합니다.
4. 편집을 위해 매핑을 준비합니다.

1단계. 배포를 위해 매핑 준비

테스트를 위해 매핑을 배포하기 전에 변경된 매핑을 식별하고 배포할 매핑을 표시합니다.

지속적인 통합을 설정하여 다음과 같은 통합 작업을 수행합니다.

1. 변경된 매핑을 식별합니다.

변경된 매핑을 식별하기 위해 각각의 디자인 타임 매핑을 데이터 통합 서비스에 배포된 해당하는 런타임 매핑과 지속적으로 비교합니다.

매핑을 비교하기 위해 쿼리를 쿼리된 두 매핑 비교 REST API 또는 `infacmd dis compareMapping` 명령에 전달합니다.

2. 태그를 할당하여 배포할 매핑을 표시합니다.

비교 보고서를 통해 디자인 타임 매핑과 해당하는 런타임 매핑 간의 차이점이 식별되면 디자인 타임 매핑에 태그를 할당하여 배포할 매핑을 표시합니다.

예를 들어 `DeploymentInProgress` 태그를 매핑에 할당합니다. 태그를 디자인 타임 개체에 태그 할당 REST API 또는 `infacmd dis tag` 명령에 전달할 수 있습니다.

2단계. 데이터 통합 서비스에 매핑 배포

매핑에 대한 배포 준비가 되었으면 변경된 매핑을 응용 프로그램 패치 보관 파일에 배포한 다음 데이터 통합 서비스에 배포합니다.

지속적인 통합을 설정하여 다음과 같은 통합 작업을 수행합니다.

1. 쿼리를 실행하여 배포하도록 표시된 매핑을 식별합니다.

배포가 필요한 매핑을 식별하려면 `DeploymentInProgress` 태그가 있는 디자인 타임 개체를 쿼리합니다. 쿼리를 디자인 타임 개체 가져오기 REST API 또는 `infacmd dis queryDesignTimeObjects` 명령에 전달합니다.

2. 매핑을 응용 프로그램 패치 보관 파일에 배포합니다.

쿼리에서 최소 하나의 매핑을 반환하면 응용 프로그램 패치 보관 파일에 매핑을 배포합니다.

쿼리를 보관 파일에 디자인 타임 개체 배포 REST API 또는 `infacmd dis deployObjectsToFile` 명령에 전달합니다. REST API 요청 또는 `infacmd` 명령이 보관 파일을 데이터 통합 서비스 시스템에 저장합니다.

3. 보관 파일을 데이터 통합 서비스에 배포합니다.

보관 파일을 개발 환경의 데이터 통합 서비스에 배포합니다. 보관 파일의 위치를 응용 프로그램 업데이트 REST API 또는 `infacmd tools patchApplication` 명령에 전달합니다.

3단계. 온전성 테스트 실행

데이터 통합 서비스에서 매핑이 업데이트되면 온전성 테스트를 실행하여 매핑이 성공적으로 실행되는지 확인합니다.

다음과 같은 통합 작업을 수행하려면 지속적인 통합을 설정합니다.

1. 배포된 매핑이 성공적으로 실행되는지 확인합니다.

배포된 매핑이 성공적으로 실행되는지 확인하려면 다음 태스크를 수행합니다.

1. 배포된 매핑을 실행하려면 배포된 매핑 실행 REST API 또는 `infacmd ms runMapping` 명령을 사용합니다.
2. 조직의 요구 사항에 따라 온전성 테스트를 실행합니다. 예를 들어 타사 도구에서 스크립트를 사용하여 통합 태스크를 자동화하는 경우 테스트 사례를 써서 매핑 대상에서 데이터를 읽고 매핑 출력이 올바른지 확인할 수 있습니다.

2. 태그를 할당하여 추가 변경이 필요한 매핑을 표시합니다.

추가 변경이 필요한 매핑을 표시하려면 `RequiresChanges` 태그를 디자인 타임 개체에 태그 할당 REST API 또는 `infacmd dis tag` 명령에 전달합니다.

변경이 필요한 매핑에서 `DeploymentInProgress` 태그를 제거할 수도 있습니다. `DeploymentInProgress` 태그를 디자인 타임 개체에서 태그 제거 REST API 또는 `infacmd dis untag` 명령에 전달합니다.

3. 태그를 할당하여 성공적으로 실행된 매핑을 승인합니다.

성공적으로 실행된 매핑을 승인하려면 `ReadyForDelivery` 태그를 리포지토리 개체 태그 바꾸기 REST API 또는 `infacmd dis replaceAllTag` 명령에 전달합니다. REST API 요청 또는 `infacmd` 명령을 수행하면 매핑에 할당된 모든 태그가 `ReadyForDelivery` 태그로 바뀝니다.

4단계. 편집을 위해 매핑 준비

업데이트된 매핑이 테스트에 실패하는 경우 편집할 매핑을 준비합니다.

지속적인 통합을 설정하여 다음과 같은 통합 작업을 수행합니다.

1. 편집을 위해 매핑을 적절한 개발자에게 할당합니다.

매핑을 적절한 개발자에게 할당하려면 다음 태스크를 수행합니다.

1. 디자인 타임 개체 가져오기 REST API 또는 `infacmd dis queryDesignTimeObjects` 명령을 사용하여 `RequiresChanges` 태그가 있는 디자인 타임 매핑을 쿼리합니다.
2. 사용 중인 타사 도구를 기반으로 쿼리 결과를 구문 분석하여 각 매핑을 마지막으로 수정한 개발자의 사용자 이름을 가져올 수 있습니다.
3. 편집을 위해 매핑을 개발자에게 할당하려면 버전 제어 작업 수행 REST API 또는 `infacmd mrs reassignCheckedOutObject` 명령을 사용합니다.

참고: 버전 제어 작업은 모델 리포지토리가 버전 제어 시스템과 통합된 경우에만 사용할 수 있습니다.

2. 개발자에게 테스트 결과에 대해 알립니다.

필요한 경우 타사 도구를 사용하여 변경이 필요한 매핑에 대한 내용을 개발자에게 알릴 수 있습니다.

예를 들어 디자인 타임 개체 가져오기 REST API 또는 `infacmd dis queryDesignTimeObjects` 명령을 사용하여 `RequiresChanges` 태그가 있는 디자인 타임 매핑을 쿼리할 수 있습니다.

그런 다음 타사 도구에서 스크립트를 실행하여 다음 태스크를 수행합니다.

1. 쿼리 결과를 읽고 응답을 구문 분석하여 매핑을 마지막으로 수정한 개발자의 사용자 이름을 가져옵니다.
2. 직원 데이터베이스에서 이 사용자 이름과 연결된 전자 메일 주소를 조회합니다.

3. 테스트에 실패한 매핑에 대한 정보와 매핑의 실패 이유를 전달하기 위해 전자 메일 메시지를 생성합니다.

자동화된 전달 프로세스 샘플

이 섹션은 개발 환경에서 매핑이 승인될 때 매핑을 프로덕션 환경에 배포하기 위해 지속적인 전달을 설정하는 방법을 보여 줍니다.

지속적인 전달을 설정하여 다음과 같은 전달 작업을 수행합니다.

1. 전달 승인된 매핑을 식별합니다.

전달 승인된 매핑을 식별하려면 다음 태스크를 수행합니다.

1. 개발 환경에 ReadyForDelivery 태그를 가진 매핑이 있는지 지속적으로 확인합니다.
2. ReadyForDelivery 태그에 대해 디자인 타임 매핑을 쿼리합니다.
3. 쿼리를 디자인 타임 개체 가져오기 REST API 또는 `infacmd dis queryDesignTimeObjects` 명령에 전달합니다.

2. 매핑을 응용 프로그램 패치 보관 파일에 배포합니다.

쿼리에서 최소 하나의 매핑을 반환하면 응용 프로그램 패치 보관 파일에 매핑을 배포합니다.

쿼리를 보관 파일에 디자인 타임 개체 배포 REST API 또는 `infacmd dis deployObjectsToFile` 명령에 전달합니다. REST API 요청 또는 `infacmd` 명령이 보관 파일을 데이터 통합 서비스 시스템에 저장합니다.

3. 보관 파일을 데이터 통합 서비스에 배포합니다.

보관 파일을 프로덕션 환경의 데이터 통합 서비스에 배포합니다. 보관 파일의 위치를 응용 프로그램 업데이트 REST API 또는 `infacmd tools patchApplication` 명령에 전달합니다.

일반적인 통합 및 전달 작업 참조

지속적인 통합과 지속적인 전달을 설정하는 데 사용할 수 있는 일반적인 통합 및 전달 작업을 참조하십시오.

REST API 요청에 대한 자세한 내용은 다음 위치 중 하나를 참조하십시오.

- 데이터 통합 서비스 프로세스 속성 또는 REST 작업 힙 서비스 프로세스 속성을 통해 액세스하는 REST API 문서.
- *Informatica 응용 프로그램 서비스 가이드*.

`infacmd` 명령에 대한 자세한 내용은 *Informatica 명령 참조*를 참조하십시오.

개체에 대한 정보 가져오기

다음 테이블에는 개체에 대한 정보를 가져오기 위해 자동화할 수 있는 작업이 나열되어 있습니다.

작업	REST API	infacmd
디자인 타임 개체 쿼리	디자인 타임 개체 가져오기	dis queryDesignTimeObjects
런타임 개체 쿼리	런타임 개체 가져오기	dis queryRunTimeObjects
응용 프로그램 나열	런타임 개체 가져오기	dis listApplications
응용 프로그램 개체 나열	런타임 개체 가져오기	dis listApplicationObjects
매개 변수 집합 나열	런타임 개체 가져오기	dis listParameterSets
응용 프로그램 패치 나열	적용할 수 없음	dis listPatchNames
매핑 비교	쿼리된 두 매핑 비교	dis compareMapping

개체 태그 지정

다음 테이블에는 개체에 태그를 지정하기 위해 자동화할 수 있는 작업이 나열되어 있습니다.

작업	REST API	infacmd
개체에 태그 지정	디자인 타임 개체에 태그 할당	dis tag
개체 태그 바꾸기	리포지토리 개체 태그 바꾸기	dis replaceAllTag
개체 태그 지정 해제	디자인 타임 개체에서 태그 제거	dis untag

충돌 검색 및 해결

다음 테이블에는 충돌 검색 및 해결을 위해 자동화할 수 있는 작업이 나열되어 있습니다.

작업	REST API	infacmd
개체 잠금 해제	적용할 수 없음	mrs unlockObject
잠긴 개체 나열	적용할 수 없음	mrs listLockedObjects

버전 제어 작업 수행

다음 테이블에는 버전 제어 작업을 수행하기 위해 자동화할 수 있는 작업이 나열되어 있습니다.

작업	REST API	infacmd
개체 체크 인	버전 제어 작업 수행	mrs checkInObject
개체 체크 아웃	버전 제어 작업 수행	적용할 수 없음
체크 아웃 실행 취소	버전 제어 작업 수행	mrs undoCheckout

작업	REST API	infacmd
체크 아웃된 개체 재할당	버전 제어 작업 수행	mrs reassignCheckedOutObject
체크 아웃된 개체 나열	디자인 타임 개체 가져오기	mrs listCheckedOutObjects

참고: 버전 제어 작업은 모델 리포지토리가 버전 제어 시스템과 통합된 경우에만 사용할 수 있습니다.

개체 배포

다음 테이블에는 개체를 배포하기 위해 자동화할 수 있는 작업이 나열되어 있습니다.

작업	REST API	infacmd
응용 프로그램 보관 파일에 개체 배포	적용할 수 없음*	tools deployApplication
응용 프로그램 패치 보관 파일에 개체 배포	보관 파일에 디자인 타임 개체 배포	dis deployObjectsToFile
데이터 통합 서비스에 응용 프로그램 보관 파일 배포	응용 프로그램 배포	dis deployApplication
데이터 통합 서비스에 응용 프로그램 패치 보관 파일 배포	응용 프로그램 업데이트	tools patchApplication
데이터 통합 서비스에 응용 프로그램 보관 파일 재배포	응용 프로그램 업데이트	dis updateApplication

* REST API를 사용하여 개체를 응용 프로그램 보관 파일에 배포할 수 없습니다. 응용 프로그램 패치 보관 파일을 사용하여 처음으로 응용 프로그램을 데이터 통합 서비스에 배포하거나 응용 프로그램을 재배포합니다.

배포된 응용 프로그램 관리

다음 테이블에는 배포된 응용 프로그램을 관리하기 위해 자동화할 수 있는 작업이 나열되어 있습니다.

작업	REST API	infacmd
응용 프로그램 시작	응용 프로그램 시작 또는 중지	dis startApplication
응용 프로그램 중지	응용 프로그램 시작 또는 중지	dis stopApplication
응용 프로그램 백업	적용할 수 없음	dis backupApplication
응용 프로그램 복원	적용할 수 없음	dis restoreApplication
응용 프로그램 이름 바꾸기	적용할 수 없음	dis renameApplication
응용 프로그램 배포 해제	응용 프로그램 배포 해제	dis undeployApplication
배포된 매핑 실행	배포된 매핑 실행	ms runMapping

REST API를 사용하여 응용 프로그램 배포 및 재배포

데이터 통합 서비스 REST API는 응용 프로그램 패치 보관 파일을 사용하여 개체를 배포합니다. 응용 프로그램 보관 파일을 사용하는 대신 응용 프로그램 패치 보관 파일을 사용하여 처음으로 개체를 데이터 통합 서비스에 배포하거나 응용 프로그램을 재배포합니다.

응용 프로그램 패치 보관 파일을 사용하여 다음 태스크를 수행합니다.

처음으로 응용 프로그램 배포

처음으로 응용 프로그램으로 개체를 데이터 통합 서비스에 배포하려면 다음 태스크를 완료합니다.

1. REST API 디자인 타임 개체를 보관 파일에 배포하십시오. 를 사용하여 응용 프로그램 패치 보관 파일을 생성합니다.
2. REST API 응용 프로그램 배포를 사용하여 패치를 데이터 통합 서비스에 배포합니다.
데이터 통합 서비스는 패치의 개체를 사용하여 증분 응용 프로그램을 생성합니다.

응용 프로그램 재배포

응용 프로그램을 재배포하려면 REST API Update an application을 사용하여 다음 구성 요소를 구성합니다.

1. 응용 프로그램 패치 보관 파일을 지정합니다.
2. 요청 페이로드 매개 변수 "redeployApplication":true를 설정합니다.
데이터 통합 서비스는 배포된 증분 응용 프로그램의 개체를 패치의 개체로 바꿉니다.

제 13 장

개체 가져오기 및 내보내기

이 장에 포함된 항목:

- [개체 가져오기 및 내보내기 개요, 178](#)
- [개체 가져오기 및 내보내기, 179](#)
- [개체 내보내기, 180](#)
- [개체 가져오기, 181](#)

개체 가져오기 및 내보내기 개요

프로젝트의 여러 개체를 .xml 파일로 내보낼 수 있습니다. 개체를 가져올 때 .xml 파일에서 개별 개체를 선택하거나 .xml 파일에서 모든 개체를 선택할 수 있습니다.

개체를 .xml 파일로 내보낸 후 .xml 파일에서 개체를 가져올 수 있습니다. 개체를 내보내면 Developer tool이 내보낸 개체의 메타데이터를 포함하는 .xml 파일을 생성합니다. 이 .xml 파일을 사용하여 개체를 프로젝트 또는 폴더로 가져옵니다. 또한 infacmd 명령을 통해 개체를 가져오고 내보낼 수 있습니다.

개체를 내보내고 가져와서 다음과 같은 태스크를 수행할 수 있습니다.

메타데이터를 프로덕션으로 배포

개발 리포지토리에서 매핑을 테스트한 후 매핑을 .xml 파일로 내보낸 다음 .xml 파일에서 프로덕션 리포지토리로 매핑을 가져올 수 있습니다.

메타데이터 보관

더 이상 필요하지 않은 개체를 리포지토리에서 제거하기 전에 .xml 파일로 내보낼 수 있습니다.

메타데이터 공유

제 3자와 메타데이터를 공유할 수 있습니다. 예를 들어 테스트 또는 분석을 위해 다른 사람에게 매핑을 보낼 수 있습니다.

리포지토리 간 메타데이터 복사

동일한 클라이언트에서 연결할 수 없는 리포지토리 간에 개체를 복사할 수 있습니다. 개체를 내보내고 .xml 파일을 대상 시스템에 전송합니다. 그런 다음 .xml 파일의 개체를 대상 리포지토리로 가져옵니다. 버전이 동일한 리포지토리 간에 개체를 내보내고 가져올 수 있습니다. 개체에 태그가 있을 경우 Developer 도구가 자동으로 리포지토리로 가져옵니다.

infacmd를 사용하여 내보내기 파일에서 읽기 가능 .xml 파일을 생성할 수 있습니다. 또한 개체를 리포지토리로 가져오기 전에 읽기 가능 .xml 파일에서 개체 이름을 편집하고 내보내기 .xml 파일을 업데이트할 수 있습니다.

이전 Informatica 릴리스에서 .xml 파일로 내보낸 개체를 현재 메타데이터 형식으로 업그레이드한 다음 업데이트된 개체를 현재 릴리스로 가져올 수 있습니다. 개체를 가져오기 전에 `mrs UpgradeExportedObjects` 명령을 사용하여 내보낸 개체를 현재 메타데이터 형식으로 업그레이드합니다.

개체 가져오기 및 내보내기

프로젝트 및 프로젝트의 개체를 가져오고 내보낼 수 있습니다. 또한 리포지토리의 응용 프로그램 보관 파일을 가져오고 내보낼 수 있습니다.

개체를 내보내면 **Developer** 도구가 종속 개체도 내보냅니다. 종속 개체는 다른 개체에서 사용되는 개체입니다. 예를 들어 매핑 입력으로 사용되는 실제 데이터 개체는 해당 매핑의 종속 개체입니다. 개체를 가져오면 **Developer** 도구가 모든 종속 개체를 가져옵니다.

프로젝트 또는 폴더의 개체를 내보내거나 가져오는 경우 모델 리포지토리 서비스가 개체 계층 구조를 유지합니다.

다음 표에는 내보낼 수 있는 개체 및 종속 개체가 나와 있습니다.

개체	종속성
응용 프로그램	- SQL 데이터 서비스, 매핑 또는 워크플로우와 해당 종속 개체
프로젝트	- 프로젝트는 다른 개체를 포함하지만 종속 개체를 포함하지 않습니다.
폴더	- 폴더는 다른 개체를 포함하지만 종속 개체를 포함하지 않습니다.
참조 테이블	- 참조 테이블은 종속 개체를 포함하지 않습니다.
콘텐츠 집합	- 콘텐츠 집합은 종속 개체를 포함하지 않습니다.
실제 데이터 개체(사용자 지정된 데이터 개체 제외)	- 실제 데이터 개체는 종속 개체를 포함하지 않습니다.
사용자 지정된 데이터 개체	- 실제 데이터 개체
논리적 데이터 개체 모델	- 논리적 데이터 개체 - 실제 데이터 개체 - 재사용 가능 변환 및 해당 종속 개체 - 맵렛 및 해당 종속 개체
변환	- 실제 데이터 개체 - 참조 테이블 - 콘텐츠 집합
맵렛	- 논리적 데이터 개체 - 실제 데이터 개체 - 재사용 가능 변환 및 해당 종속 개체 - 맵렛 및 해당 종속 개체

개체	종속성
매핑	<ul style="list-style-type: none"> - 논리적 데이터 개체 - 실제 데이터 개체 - 재사용 가능 변환 및 해당 종속 개체 - 맵렛 및 해당 종속 개체
SQL 데이터 서비스	<ul style="list-style-type: none"> - 논리적 데이터 개체 - 실제 데이터 개체 - 재사용 가능 변환 및 해당 종속 개체 - 맵렛 및 해당 종속 개체
프로필	<ul style="list-style-type: none"> - 논리적 데이터 개체 - 실제 데이터 개체
성과 기록표	<ul style="list-style-type: none"> - 프로필 및 해당 종속 개체
웹 서비스	<ul style="list-style-type: none"> - 작업 매핑
워크플로우	<ul style="list-style-type: none"> - 매핑 및 해당 종속 개체

개체 내보내기

개체를 내보내면 **Developer tool**이 개체의 메타데이터를 포함하는 XML 파일을 작성합니다.

내보낼 개체를 선택할 수 있습니다. 또한 모든 종속 개체를 내보내도록 선택해야 합니다. **Developer tool**은 개체 및 해당 종속 개체를 내보냅니다. **Developer tool**은 개체의 마지막으로 저장된 버전을 내보냅니다. **Developer tool**은 CRCVALUE(순환 중복성 확인 값) 코드를 XML 파일의 요소로 포함합니다. CRCVALUE 코드가 포함된 요소에서 특성을 수정하는 경우 개체를 가져올 수 없습니다. 특성을 수정하려면 **infacmd xrf** 명령을 사용합니다.

infacmd tools ExportObjects 명령을 사용하여 개체를 내보낼 수도 있습니다.

개체 내보내기

다른 리포지토리의 모델 리포지토리 개체를 사용하려면 개체를 XML 메타데이터 파일로 내보냅니다.

1. **파일 > 내보내기**를 클릭합니다.
내보내기 마법사가 열립니다.
2. **Informatica > 개체 메타데이터 파일 내보내기**를 선택합니다.
다음을 클릭합니다.
3. **찾아보기**를 클릭합니다. 내보낼 개체가 포함된 리포지토리 프로젝트를 선택합니다.
다음을 클릭합니다.
4. 내보낼 하나 이상의 개체를 선택합니다. 내보내기 프로세스를 시작하기 전에 리포지토리 개체를 강조 표시한 경우 마법사가 자동으로 해당 개체를 선택합니다.
5. XML 메타데이터 파일의 파일 이름과 위치를 입력합니다. **Developer** 도구가 사용자가 선택한 모든 개체를 단일 파일로 내보냅니다.

다음을 클릭합니다.

6. 마법사가 메타데이터 개체가 사용하는 종속 개체를 표시합니다.

다음을 클릭하여 종속 개체를 수락합니다.

7. 선택하는 개체에 참조 데이터 개체가 있을 경우 **콘텐츠 내보내기** 옵션을 선택하고 내보내기 설정을 확인합니다.
 - 내보내는 참조 데이터 파일의 이름과 위치를 확인합니다. **Developer** 도구 서비스는 참조 데이터 파일을 단일 ZIP 파일로 내보냅니다. 기본적으로 마법사는 ZIP 파일과 XML 메타데이터 파일을 동일한 디렉터리로 내보냅니다.
 - 참조 데이터에서 사용하는 코드 페이지를 확인합니다. 기본 코드 페이지는 UTF-8입니다. 참조 테이블 데이터를 내보내는 경우 기본 코드 페이지를 그대로 사용합니다.
 - 내보낼 확률 모델 데이터를 확인합니다. 기본적으로 마법사는 모든 모델 데이터를 내보냅니다. 선택하는 개체에 확률 모델이 없을 경우 내보내기 프로세스에서 옵션을 무시합니다.
8. **마침**을 클릭하여 선택한 개체를 내보냅니다.

Developer 도구가 개체 메타데이터를 XML 파일로 내보내고 모든 종속 참조 데이터 파일을 ZIP 파일로 내보냅니다.

개체 가져오기

내보내기 파일에서 프로젝트 또는 프로젝트 내의 개체를 가져올 수 있습니다. 개체 및 모든 종속 개체를 프로젝트 또는 폴더로 가져올 수 있습니다.

프로젝트 또는 개별 개체를 가져올 수 있습니다. 프로젝트의 모든 개체를 재사용하려는 경우 프로젝트를 가져옵니다. 프로젝트 전체의 개체를 재사용하려는 경우 개별 개체를 가져옵니다.

개체를 가져오면 **Developer tool**이 모든 종속 개체를 나열합니다. 개체를 가져오려면 각 종속 개체를 대상에 추가해야 합니다.

개체를 가져올 때 내보내기 파일에 있는 개체의 이름이 대상 프로젝트 또는 폴더에 있는 개체의 이름과 같을 수 있습니다. 이름 지정 충돌에 대한 해결 방법을 선택할 수 있습니다.

`infacmd tools ImportObjects` 명령을 사용하여 개체를 가져올 수도 있습니다.

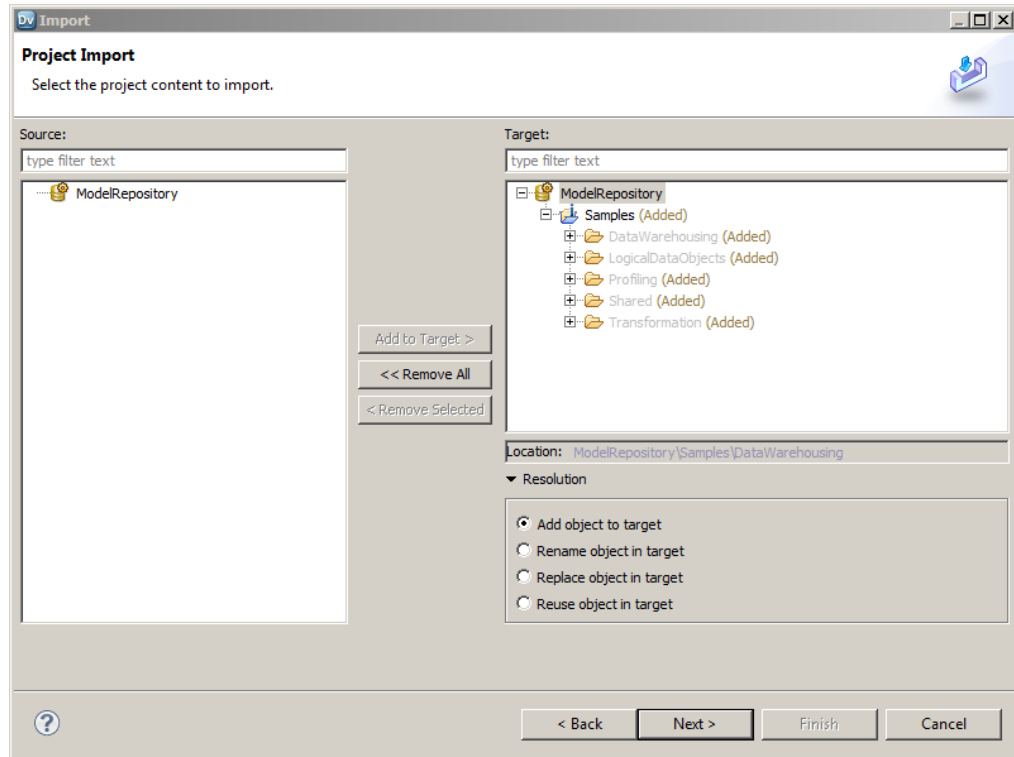
이전 버전에서 작성한 내보내기 파일에서 개체를 가져올 수는 없습니다.

프로젝트 가져오기

XML 파일에서 대상 리포지토리로 프로젝트를 가져올 수 있습니다. 또한 프로젝트의 콘텐츠를 대상 리포지토리의 프로젝트로 가져올 수 있습니다.

1. **파일 > 가져오기**를 클릭합니다.
2. **Informatica > 개체 메타데이터 파일 가져오기(기본)**을 선택합니다.
3. **다음**을 클릭합니다.
4. **찾아보기**를 클릭하고 가져올 내보내기 파일을 선택합니다.
5. **다음**을 클릭합니다.
6. 프로젝트를 선택하거나 소스 창에서 "<project name>프로젝트 콘텐츠"를 선택합니다.
 - 소스 창에서 프로젝트를 선택하는 경우 프로젝트를 가져올 대상 창에서 모델 리포지토리 서비스를 선택합니다.

- 소스 창에서 프로젝트 콘텐츠를 선택하는 경우 대상 창에서 프로젝트 콘텐츠를 가져올 프로젝트를 선택합니다.
7. **대상에 추가**를 클릭하여 프로젝트를 대상에 추가합니다.
- 팁:** 또한 소스 창의 프로젝트를 대상 창의 리포지토리로 끌어서 놓을 수 있습니다. 또는 소스 창의 프로젝트 콘텐츠를 대상 창의 프로젝트로 끌어서 놓을 수 있습니다.
8. **해결**을 클릭하여 중복 개체의 처리 방법을 지정합니다.
- 가져온 개체의 이름을 바꾸거나 기존 개체를 가져온 개체로 바꾸거나 기존 개체를 재사용할 수 있습니다. 기본적으로 **Developer** 도구는 모든 중복 개체의 이름을 바꿉니다.

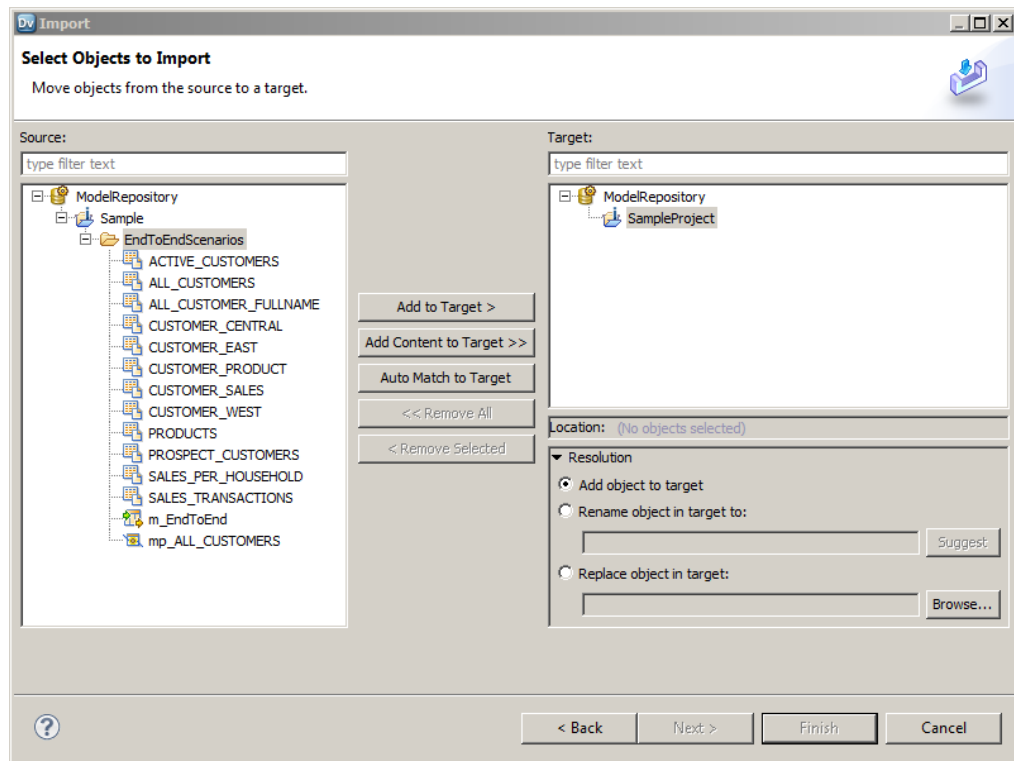


9. **다음**을 클릭합니다.
- Developer 도구가 가져오려는 참조 체이بل 데이터를 나열합니다. 추가 참조 테이블 설정을 지정합니다.
10. **다음**을 클릭합니다.
- Developer 도구가 가져올 개체를 요약합니다. **소스 및 대상 개체 연결**을 클릭하면 개체 중 하나를 선택할 때 소스 및 대상 표시 창에서 개체를 연결할 수 있습니다. 예를 들어 이 옵션을 선택한 후 소스 창에서 개체를 선택하면 Developer 도구가 대상 창에서 동일한 개체를 선택합니다.
11. 추가 가져오기 설정 창에서 가져오기 파일의 연결을 대상 도메인 연결에 매핑합니다. 또한 개체에 대해 기존 태그를 덮어쓸지 여부도 선택할 수 있습니다.
12. **마침**을 클릭합니다.
- 중복 프로젝트의 이름을 바꾸도록 선택했을 경우 모델 리포지토리 서비스에서 개체 이름에 숫자를 추가합니다. 프로젝트를 가져온 후 프로젝트 이름을 바꿀 수 있습니다.

개체 가져오기

XML 파일 또는 응용 프로그램 보관 파일에서 개체를 가져올 수 있습니다. 개체 및 모든 종속 개체를 프로젝트로 가져옵니다.

1. **파일 > 가져오기**를 클릭합니다.
2. **Informatica > 개체 메타데이터 파일 가져오기(고급)**를 선택합니다.
3. **다음**을 클릭합니다.
4. **찾아보기**를 클릭하고 가져올 내보내기 파일을 선택합니다.
5. **다음**을 클릭합니다.
6. 소스 창에서 가져올 개체를 선택합니다.
7. 대상 창에서 개체를 가져올 프로젝트를 선택합니다.
8. **대상에 추가**를 클릭하여 개체를 대상에 추가합니다.



대상에 자동 일치를 클릭하면 Developer tool이 대상 선택 항목에서 이름, 유형 및 상위 계층을 기준으로 현재 소스 선택 항목의 하위 항목을 개별적으로 일치시키려고 하고 일치하는 개체를 추가합니다.

폴더 또는 프로젝트 아래 모든 개체를 가져오려면 대상 폴더 또는 프로젝트를 선택하고 **대상에 콘텐츠 추가**를 클릭합니다.

팁: 또한 소스 창에서 대상 창의 필요한 프로젝트로 개체를 끌어서 놓을 수 있습니다. 소스 및 대상에서 개체 계층 구조를 유지하려면 Ctrl 키를 누른 상태에서 끌어서 놓습니다.

9. 클릭하여 중복 개체를 처리할 방법을 지정합니다.

가져온 개체의 이름을 바꾸거나 기존 개체를 가져온 개체로 바꾸거나 기존 개체를 재사용할 수 있습니다. 기본적으로 Developer tool은 모든 중복 개체의 이름을 바꿉니다.

10. **다음**을 클릭합니다.

Developer tool이 가져오기 파일의 종속 개체를 나열합니다.

11. 종속 개체를 대상 폴더 또는 프로젝트에 추가합니다.
12. 다음을 클릭합니다.
Developer tool이 가져오려는 참조 체이블 데이터를 나열합니다. 추가 참조 테이블 설정을 지정합니다.
13. 다음을 클릭합니다.
Developer tool이 가져올 개체를 요약합니다. **소스 및 대상 개체 연결**을 클릭하면 개체 중 하나를 선택할 때 소스 및 대상 표시 창에서 개체를 연결할 수 있습니다. 예를 들어 이 옵션을 선택한 후 소스 창에서 개체를 선택하면 Developer tool이 대상 창에서 동일한 개체를 선택합니다.
14. 추가 가져오기 설정 창에서 가져오기 파일의 연결을 대상 도메인 연결에 매핑합니다. 또한 개체에 대해 기존 태그를 덮어쓸지 여부도 선택할 수 있습니다.
15. 마침을 클릭합니다.
중복 프로젝트의 이름을 바꾸기로 선택할 경우 **가져오기** 마법사는 가져온 프로젝트의 이름을 "<원래 이름>_<사본 번호>"로 지정합니다. 프로젝트를 가져온 후 프로젝트 이름을 바꿀 수 있습니다.

이전 Informatica 릴리스에서 개체 가져오기

이전 Informatica 릴리스에서 내보낸 개체를 현재 메타데이터 형식으로 업그레이드한 다음 업그레이드된 개체를 현재 Informatica 릴리스로 가져올 수 있습니다. 내보낸 개체를 현재 릴리스로 가져오기 전에 해당 개체를 현재 메타데이터 형식으로 업그레이드해야 합니다.

여러 개의 Informatica 릴리스가 설치된 엔터프라이즈 환경에서 이전 Informatica 릴리스의 개체를 현재 Informatica 릴리스로 가져올 수 있습니다. 예를 들어 이전 Informatica 릴리스의 개발 도메인에서 개체를 내보낸 다음 현재 Informatica 릴리스의 테스트 도메인으로 개체를 가져올 수 있습니다.

Informatica 10.1 또는 10.1.1 모델 리포지토리의 개체를 개체 메타데이터가 포함되는 .xml 파일로 내보냅니다. `mrs UpgradeExportedObjects` 명령을 사용하여 내보낸 개체를 현재 메타데이터 형식으로 업그레이드합니다. 이 명령을 실행할 때 내보낸 개체가 포함되는 .xml 파일의 경로와 파일 이름을 지정해야 합니다.

`mrs UpgradeExportedObjects` 개체 명령을 실행하기 전에 .xml 파일을 수정하지 마십시오. 명령을 실행하기 전에 파일을 수정하면 파일의 개체가 성공적으로 업그레이드되지 않을 수 있습니다.

`mrs UpgradeExportedObjects` 명령은 개체를 업그레이드한 다음 업그레이드된 개체가 포함되는 .xml 파일을 생성합니다. .xml 파일의 파일 이름과 경로를 지정해야 합니다. .xml 파일을 현재 Informatica 릴리스의 리포지토리로 가져올 수 있습니다.

`mrs UpgradeExportedObjects` 명령을 사용하려면 Informatica 10.2 모델 리포지토리 서비스에 액세스해야 합니다. 명령을 실행할 때 도메인 내에서 실행되는 Informatica 10.2 모델 리포지토리 서비스의 서비스 이름을 제공해야 합니다.

`mrs UpgradeExportedObjects` 명령에 대한 자세한 내용은 *Informatica 명령 참조*를 참조하십시오.

부록 A

데이터 유형 참조

이 부록에 포함된 항목:

- [데이터 유형 참조 개요, 185](#)
- [변환 데이터 유형, 186](#)
- [복합 파일 및 변환 데이터 유형, 197](#)
- [플랫 파일 및 변환 데이터 유형, 204](#)
- [DB2 for LUW 및 변환 데이터 유형, 205](#)
- [DB2 for i5/OS, DB2 for z/OS 및 변환 데이터 유형, 206](#)
- [JDBC 및 변환 데이터 유형, 207](#)
- [Microsoft SQL Server 및 변환 데이터 유형, 209](#)
- [비관계형 및 변환 데이터 유형, 211](#)
- [ODBC 및 변환 데이터 유형, 213](#)
- [Oracle 및 변환 데이터 유형, 215](#)
- [SAP HANA 및 변환 데이터 유형, 217](#)
- [XML 및 변환 데이터 유형, 219](#)
- [데이터 변환, 221](#)

데이터 유형 참조 개요

매핑을 작성할 때 데이터 통합 서비스가 소스의 데이터를 읽고, 해당 데이터를 변환하여, 대상에 데이터를 쓰는 지침 집합을 작성합니다. 데이터 통합 서비스는 매핑의 첫 번째 변환에서 시작하여 매핑의 데이터 흐름 및 매핑의 각 포트에 할당된 데이터 유형을 기반으로 데이터를 변환합니다.

Developer tool에는 다음 두 가지 유형의 데이터 유형이 표시됩니다.

원시 데이터 유형

원시 데이터 유형은 실제 데이터 개체로 사용되는 관계형 테이블 또는 플랫 파일과 관련됩니다. 원시 데이터 유형은 실제 데이터 개체의 열 속성에 표시됩니다.

변환 데이터 유형

변환 데이터 유형은 변환에 표시되는 데이터 유형의 집합입니다. ANSI SQL-92 일반 데이터 유형에 기반하는 내부 데이터 유형으로, 데이터 통합 서비스는 이 데이터 유형을 사용하여 여러 플랫폼에서 데이터를 이동합니다. 변환 데이터 유형은 매핑의 모든 변환에 표시됩니다.

변환 데이터 유형에는 다음 데이터 유형이 포함됩니다.

- 기본 데이터 유형. 단일 열 위치에 있는 단일 데이터 값을 나타냅니다.
- 복합 데이터 유형. 단일 열 위치에 있는 여러 데이터 값을 나타냅니다. 복합 파일의 계층 데이터를 처리하려면 **Spark** 엔진에서 실행되는 매핑에 복합 데이터 유형을 사용합니다.

데이터 통합 서비스는 소스 데이터를 읽을 때 원시 데이터 유형을 비교 가능한 변환 데이터 유형으로 변환한 후 데이터를 변환합니다. 데이터 통합 서비스가 대상에 쓸 때는 변환 데이터 유형을 비교 가능한 원시 데이터 유형으로 변환합니다.

다중 바이트 문자 집합을 지정하는 경우 데이터 유형은 최대 3바이트의 문자를 저장할 수 있는 추가 공간을 데이터베이스에 할당합니다.

변환 데이터 유형

다음 테이블에는 변환 데이터 유형이 설명되어 있습니다.

데이터 유형	크기(바이트)	설명
Array	제한 없는 수의 문자입니다.	복합 데이터 유형. 복합 소스 및 대상이 포함된 배열을 사용할 수 있습니다.
Bigint	8바이트	-9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807 전체 자릿수 19, 소수 자릿수 0 정수 값.
이진	전체 자릿수	1~104,857,600바이트 플랫 파일 소스에 이진 데이터를 사용할 수 없습니다.
날짜/시간	16바이트	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일 전체 자릿수 29, 소수 자릿수 9 (나노초까지의 전체 자릿수) 결합된 날짜/시간 값.
10진수	8바이트(높은 정밀도가 비활성화되거나 전체 자릿수가 38보다 큰 경우) 16바이트(전체 자릿수가 18보다 작거나 같고 높은 정밀도가 활성화된 경우) 20바이트(전체 자릿수가 18보다 크고 28보다 작거나 같은 경우) 24바이트(전체 자릿수가 28보다 크고 38보다 작거나 같은 경우)	선언된 전체 자릿수와 소수 자릿수가 있는 10진수 값. 소수 자릿수는 전체 자릿수보다 작거나 같아야 합니다. 최대 38자리의 전체 자릿수를 지원하는 변환에서 전체 자릿수는 1 ~ 38자리이고 소수 자릿수는 0 ~ 38입니다. 최대 28자리의 전체 자릿수를 지원하는 변환에서 전체 자릿수는 1 ~ 28자리이고 소수 자릿수는 0 ~ 28입니다. 전체 자릿수를 최대 자릿수보다 많이 지정하는 경우 데이터 통합 서비스가 10진수 값을 많은 전체 자릿수 모드에서 두 배로 변환합니다.
배정밀도	8바이트	배정밀도 부동 소수점 숫자 값. 전체 자릿수와 소수 자릿수를 편집할 수 있습니다. 소수 자릿수는 전체 자릿수보다 작거나 같아야 합니다.

데이터 유형	크기(바이트)	설명
정수	4바이트	-2,147,483,648 ~ 2,147,483,647 전체 자릿수 10, 소수 자릿수 0 정수 값.
Map	제한 없는 수의 문자입니다.	복합 데이터 유형. 복합 소스 및 대상이 포함된 맵을 사용할 수 있습니다.
문자열	유니코드 모드: (전체 자릿수 + 1) * 2 ASCII 모드: 전체 자릿수 + 1	1 ~ 104,857,600자 고정 길이 또는 변경되는 길이 문자열입니다.
Struct	제한 없는 수의 문자입니다.	복합 데이터 유형 복합 소스 및 대상이 포함된 구조체를 사용할 수 있습니다.
텍스트	유니코드 모드: (전체 자릿수 + 1) * 2 ASCII 모드: 전체 자릿수 + 1	1 ~ 104,857,600자 고정 길이 또는 변경되는 길이 문자열입니다.
timestampWithTZ	40바이트	A.D. 1947년 8월 1일 ~ A.D. 2040년 12월 31일 -12:00 ~ +14:00 전체 자릿수 36, 소수 자릿수 9 (나노초까지의 전체 자릿수) 시간대가 포함된 타임스탬프 데이터 유형은 다음과 같은 시간대 영역을 지원하지 않습니다. - AFRICA_CAIRO - AFRICA_MONROVIA - EGYPT - AMERICA_MONTREAL

Integer 데이터 유형

정수 데이터를 소스에서 대상으로 전달하고 정수 데이터에 대해 변환을 수행할 수 있습니다. 변환 언어는 Bigint 및 Integer 데이터 유형을 지원합니다.

변환 Integer 데이터 유형은 정확한 값을 나타냅니다.

계산에서 정수 값 사용

계산에서 정수 값을 사용할 경우 데이터 통합 서비스는 경우에 따라 계산을 수행하기 전에 정수 값을 부동 소수점 숫자로 변환합니다. 예를 들어 MOD(12.00, 5)를 평가하기 위해 데이터 통합 서비스는 나누기 연산을 수행하기 전에 정수 값 "5"를 부동 소수점 숫자로 변환합니다. 데이터 통합 서비스는 사용자가 많은 전체 자릿수를 활성화하는지 여부에 따라 정수 값을 배정밀도 값 또는 10진수 값으로 변환합니다.

데이터 통합 서비스는 다음과 같은 산술 연산에서 정수 값을 변환합니다.

산술 연산	많은 전체 자릿수 비활성화됨	많은 전체 자릿수 활성화됨
소수점을 넣을 수 없는 함수 및 계산 예: 정수 더하기, 빼기 및 곱하기와 CUME, MOVINGSUM, SUM 등의 함수	변환 없음 ¹	10진수
소수점을 넣을 수 있는 비과학 함수 및 계산 예: 정수 나누기와 AVG, MEDIAN, PERCENTILE 등의 함수	배정밀도	10진수
모든 과학 함수와 EXP, LN, LOG, POWER 및 SQRT 함수	배정밀도	배정밀도

¹ 계산에서 범위를 벗어난 결과를 생성할 경우 통합 서비스가 행 오류를 씁니다.

변환 배정밀도 데이터 유형은 최대 15자리의 전체 자릿수를 지원하고 Bigint 데이터 유형은 최대 19자리의 전체 자릿수를 지원합니다. 따라서 전체 자릿수가 15자리가 넘는 Bigint 값을 생성하는 계산에서는 전체 자릿수 손실이 발생할 수 있습니다.

예를 들어 식 변환에 다음과 같은 계산이 있을 수 있습니다.

POWER(BIGINTVAL, EXPVAL)

계산을 수행하기 전에 데이터 통합 서비스는 POWER 함수에 대한 입력을 배정밀도 값으로 변환합니다.

BIGINTVAL 포트에 Bigint 값 9223372036854775807이 있으면 데이터 통합 서비스는 이 값을 9.22337203685478e+18로 변환하므로 전체 자릿수의 마지막 4자리가 손실됩니다. EXPVAL 포트에 값 1.0이 있고 결과 포트가 Bigint일 경우 이 계산은 결과가 9223372036854780000이고 최대 bigint 값을 초과하므로 행 오류를 생성합니다.

10진수 값을 생성할 수 있는 계산에서 정수 데이터 유형을 사용하고 많은 전체 자릿수를 활성화할 경우 데이터 통합 서비스는 정수 값을 10진수 값으로 변환합니다.

최대 전체 자릿수가 28인 10진수 데이터 유형을 지원하는 변환의 경우, 높은 전체 자릿수 모드에서 결과가 전체 자릿수가 28자리를 넘는 값을 생성하지 않는 한 계산에서 전체 자릿수 손실이 발생하지 않습니다. 이 경우 데이터 통합 서비스는 결과를 배정밀도로 저장합니다. 포트 전체 자릿수가 28자리 이하인 경우, 높은 전체 자릿수 모드에서 결과가 28자리보다 큰 값을 생성하면 데이터 통합 서비스가 해당 행을 거부합니다.

최대 전체 자릿수가 38인 10진수 데이터 유형을 지원하는 변환의 경우, 높은 전체 자릿수 모드에서 결과가 전체 자릿수가 38자리를 넘는 값을 생성하지 않는 한 계산에서 전체 자릿수 손실이 발생하지 않습니다. 이 경우 데이터 통합 서비스는 결과를 배정밀도로 저장합니다. 포트 전체 자릿수가 38자리 이하인 경우, 높은 전체 자릿수 모드에서 결과가 38자리보다 큰 값을 생성하면 데이터 통합 서비스가 해당 행을 거부합니다.

식의 정수 상수

통합 서비스는 계산에서 정수 결과를 생성해도 식의 상수를 부동 소수점 값으로 해석합니다. 예를 들어 많은 전체 자릿수가 활성화되지 않았을 경우 식 INTVALUE + 1000에서 통합 서비스는 정수 값 "1000"을 배정밀도 값으로 변환합니다. 많은 전체 자릿수가 활성화되었을 경우에는 값 "1000"을 10진수 값으로 변환합니다. 값 1000을 정수 값으로 처리하려면 상수를 유지하도록 정수 데이터 유형으로 변수 포트를 생성하고 식을 수정하여 두 개의 포트를 추가합니다.

NaN 값

NaN(숫자가 아님)은 특히 부동 소수점 계산에서 잘못된 입력 피연산자에 대한 연산의 결과로 대개 반환되는 값입니다. 예를 들어 연산에서 0으로 0을 나누려고 하는 경우 NaN 결과를 반환합니다.

운영 체제와 프로그래밍 언어가 NaN을 다르게 나타낼 수 있습니다. 예를 들어 다음 목록은 NaN의 올바른 문자열 표현을 보여 줍니다.

```
nan
NaN
NaN%
NAN
NaNQ
NaNS
qNaN
sNaN
1.#SNAN
1.#QNAN
```

통합 서비스는 Win64EMT 플랫폼에서 QNAN 값을 1.#QNAN으로 변환합니다. 1.#QNAN은 NaN의 올바른 표현입니다.

플랫 파일에 정수 값 쓰기

고정 너비 플랫 파일에 정수 값을 쓰는 경우 파일 기록기는 데이터가 범위 내에 있는지 확인하지 않습니다. 예를 들어 대상 열의 필드 너비가 13 이상일 경우 파일 기록기는 결과 3,000,000,000을 대상 정수 열에 씁니다. 파일 기록기는 결과가 정수 값의 올바른 범위를 벗어나기 때문에 행을 거부하지 않습니다.

Binary 데이터 유형

매핑에 이진 데이터가 있을 경우 통합 서비스가 소스에서 대상으로 데이터를 이동하기에 충분한 메모리를 할당할 수 있도록 변환 Binary 데이터 유형의 전체 자릿수를 설정합니다.

플랫 파일 소스에는 Binary 데이터 유형을 사용할 수 없습니다.

Date/Time 데이터 유형

Date/Time 데이터 유형은 양력 체계의 A.D. 1년부터 A.D. 9999년까지의 연도를 처리합니다. A.D. 9999년을 넘어가는 연도는 오류를 일으킵니다.

Date/Time 데이터 유형은 전체 자릿수가 나노초 단위까지인 날짜를 지원합니다. 이 데이터 유형의 전체 자릿수는 29이고 소수 자릿수는 9입니다. 일부 원시 데이터 유형의 전체 자릿수는 이보다 작습니다. 날짜/시간 값이 포함된 소스를 가져오는 경우 가져오기 프로세스가 소스 열에서 올바른 전체 자릿수를 가져옵니다. 예를 들어 Microsoft SQL Server의 Datetime 데이터 유형은 전체 자릿수가 23이고 소수 자릿수가 3입니다. 날짜/시간 값이 포함된 Microsoft SQL Server 소스를 가져오면 매핑 소스에 있는 날짜/시간 열의 전체 자릿수는 23이고 소수 자릿수는 3입니다.

통합 서비스는 소스의 날짜/시간 값을 매핑 소스에 지정된 전체 자릿수로 읽습니다. 통합 서비스는 날짜/시간 값을 변환할 때 최대 29자리의 전체 자릿수를 지원합니다. 예를 들어 전체 자릿수가 밀리초까지인 날짜/시간 값을 가져오는 경우 식 변환에서 ADD_TO_DATE 함수를 사용하여 나노초를 날짜에 추가할 수 있습니다.

더 작은 전체 자릿수를 지원하는 대상 열에 날짜/시간 값을 쓰는 경우 통합 서비스는 대상 열의 전체 자릿수로 값을 자릅니다. 더 큰 전체 자릿수를 지원하는 대상 열에 날짜/시간 값을 쓰는 경우 통합 서비스는 날짜/시간 값의 지원되지 않는 부분에 0을 삽입합니다.

시간대가 포함된 타임스탬프

시간대가 포함된 타임스탬프는 시간대 오프셋 TIME_ZONE_HOUR: TIME_ZONE_MINUTE이 포함된 타임스탬프 데이터 유형의 변형이며, 일광 절약 시간 또는 시간대 영역 이름을 포함하거나 포함하지 않을 수 있습니다. 시간대 오프셋은 현지 시간대와 UTC(협정 세계시) 사이의 차이(시간 및 분)입니다.

예를 들어 '16-JUN-08 07.01.25.376000 PM -06:00'에서 -06:00이 시간대 오프셋입니다.

또 다른 예로 '05-JUN-2008 07:01:25.376000 PM America/Los_Angeles'에서는 America/Los_Angeles가 시간대 영역 이름입니다.

시간대가 포함된 타임스탬프 데이터 유형을 Developer tool로 가져오는 경우, 연결된 변환 데이터 유형은 timestampWithTZ입니다.

timestampWithTZ의 전체 자릿수는 36이고 소수 자릿수는 9입니다. 시간대가 포함된 타임스탬프 범위 값 범위는 -12:00 < UTC < +14:00입니다.

데이터 통합 서비스는 Oracle 및 플랫폼 파일 데이터 개체의 시간대가 포함된 타임스탬프 데이터 유형을 처리할 수 있습니다. 시간대가 포함된 타임스탬프 데이터 유형은 1947년 8월부터 2040년 12월 사이의 데이터에만 적용할 수 있습니다.

Developer tool에서 다음과 같은 시간대 영역은 시간대가 포함된 타임스탬프 데이터 유형에 지원되지 않습니다.

- AFRICA_CAIRO
- AFRICA_MONROVIA
- EGYPT
- AMERICA_MONTREAL

시간대가 포함된 타임스탬프 데이터 유형은 시간대 데이터베이스 버전 2015b에서 IANA 표준을 사용합니다. 시간대가 포함된 타임스탬프 데이터 유형을 사용할 때 데이터 손상을 방지하려면 Oracle 서버 및 클라이언트에 Oracle DST 패치 24가 있어야 합니다. 데이터 개체 캐싱을 활성화할 경우, 데이터 개체 캐싱에 사용되는 데이터베이스 서버에 Oracle DST 패치 24가 있어야 합니다.

이 패치가 없는 경우에는 Oracle 웹 사이트에서 Oracle DST 패치 24를 다운로드하십시오. Oracle 서버에서 Oracle DST 패치 24를 사용하는지 확인하려면 Oracle 서버의 명령 프롬프트에서 다음 명령을 실행하십시오.

```
SELECT VERSION FROM v$timezone_file;
```

버전이 24로 표시되는지 확인하면 됩니다. Oracle 서버에서 다음 명령을 실행하여 버전을 확인할 수도 있습니다.

```
SELECT TZ_VERSION FROM registry$database;
```

TZ_VERSION이 24로 표시되는지 확인하면 됩니다.

현지 시간대가 포함된 타임스탬프

현지 시간대가 포함된 타임스탬프는 데이터베이스 시간대에 맞게 타임스탬프 데이터가 정규화되는 타임스탬프 데이터 유형의 변형입니다. 시간대 변위는 열 데이터의 일부가 아닙니다. 데이터 통합 서비스가 데이터를 읽으면 Oracle에서 데이터 통합 서비스의 시간대에 맞게 데이터를 반환합니다.

현지 시간대가 포함된 타임스탬프 데이터 유형을 Developer tool로 가져오는 경우, 연결된 변환 데이터 유형은 날짜/시간입니다.

예: '04-APR-10 10.27.451 AM'

현지 시간대가 포함된 타임스탬프 (9)는 전체 자릿수가 29이고 소수 자릿수는 9입니다. 이는 날짜/시간(29,9) 변환 데이터 유형에 매핑됩니다.

데이터 통합 서비스가 현지 시간이 포함된 타임스탬프를 읽거나 쓸 때의 기본 세션 시간대를 설정하려면 ORA_SDTZ 환경 변수를 지정하십시오. ORA_SDTZ 환경 변수는 다음 값 중 하나로 설정할 수 있습니다.

- 운영 체제 현지 시간대('OS_TZ')
- 데이터베이스 시간대('DB_TZ')
- UTC의 절대 오프셋(예: '-05:00')
- 시간대 영역 이름(예: 'America/Los_Angeles')

Oracle에 대해 지원되는 시간대

다음 설치 위치에서는 AFRICA_CAIRO, AFRICA_MONROVIA, EGYPT 및 AMERICA_MONTREAL을 제외하고 Oracle에 대해 지원되는 모든 시간대의 목록이 포함된 시간대 파일을 확인할 수 있습니다.

<Informatica 설치 디렉터리>/services/shared/timezones/timezones.txt

Oracle에서 지원하는 추가적인 시간대에 기반하여 나중에 시간대를 추가하려면 새로운 시간대가 포함된 새 파일을 동일한 설치 위치에 저장하십시오.

Decimal 및 Double 데이터 유형

10진수 및 배정밀도 데이터를 소스에서 대상으로 전달하고 10진수 및 배정밀도 데이터에 대해 변환을 수행할 수 있습니다.

변환 언어는 다음과 같은 데이터 유형을 지원합니다.

10진수

최대 38자리의 전체 자릿수를 지원하는 변환에서 전체 자릿수는 1 ~ 38자리이고 소수 자릿수는 0 ~ 38입니다. 최대 28자리의 전체 자릿수를 지원하는 변환에서 전체 자릿수는 1 ~ 28자리이고 소수 자릿수는 0 ~ 28입니다. 소수 자릿수가 전체 자릿수보다 크거나 전체 자릿수가 음수인 10진수 값은 사용할 수 없습니다. 변환은 10진수 데이터 유형에 할당하는 모든 범위를 표시하지만 데이터 통합 서비스가 지원하는 전체 자릿수는 변환에 따라 최대 38자리 또는 28자리입니다.

높은 전체 자릿수를 활성화하고, 변환에 따라 포트의 전체 자릿수가 38자리 또는 28자리보다 크면 데이터 통합 서비스가 결과를 배정밀도로 저장합니다.

배정밀도

배정밀도 부동 소수점 숫자 값입니다.

전체 자릿수와 소수 자릿수를 편집할 수 있습니다. 소수 자릿수는 전체 자릿수보다 작거나 같아야 합니다.

계산의 10진수 및 배정밀도 값

다음 테이블에는 데이터 통합 서비스가 많은 전체 자릿수 구성에 따라 10진수 값을 처리하는 방식이 나와 있습니다.

포트 데이터 유형	전체 자릿수	높은 정밀도 비활성화	높은 정밀도 활성화
10진수	0 ~ 15	10진수	10진수
10진수	전체 자릿수가 최대 38 자리인 10진수 데이터 유형을 지원하는 변환에 대한 15 ~ 38자리입니다. 전체 자릿수가 최대 28 자리인 10진수 데이터 유형을 지원하는 변환에 대한 15 ~ 28자리입니다.	배정밀도	10진수
10진수	전체 자릿수가 최대 38 자리인 10진수 데이터 유형을 지원하는 변환에 대해 38자리를 초과합니다. 전체 자릿수가 최대 28 자리인 10진수 데이터 유형을 지원하는 변환에 대해 28자리를 초과합니다.	배정밀도	배정밀도

많은 전체 자릿수를 활성화하면 데이터 통합 서비스가 식 함수의 숫자 상수를 10진수로 변환합니다. 많은 전체 자릿수를 활성화하지 않으면 데이터 통합 서비스가 숫자 상수를 배정밀도로 변환합니다.

변환에 따라 28자리 또는 38자리가 넘는 숫자 값에 대해 최대 전체 자릿수를 보장할 수 있습니다. 변환 함수를 사용하여 변환 또는 계산을 수행하기 전에 큰 숫자를 자르거나 반올림합니다.

전체 자릿수가 최대 38자리인 10진수 데이터 유형을 지원하는 변환인 경우 전체 자릿수를 최대 38자리까지 보장하려면 10진수 데이터 유형을 사용하고 많은 전체 자릿수를 활성화합니다.

조회 또는 조인 조건 등의 같음 조건에서 사용하는 데이터에는 배정밀도 데이터 유형을 사용하지 마십시오.

10진수 및 배정밀도 값에 대한 전체 자릿수 손실

결과에서 전체 자릿수가 허용되는 최대 자릿수를 넘는 값을 생성하는 경우 계산에서 10진수 및 배정밀도 데이터 유형 모두 전체 자릿수 손실이 발생할 수 있습니다.

전체 자릿수가 허용되는 최대 자릿수보다 큰 경우 데이터 통합 서비스는 결과를 배정밀도로 저장합니다.

중간 결과의 전체 자릿수 또는 소수 자릿수가 변경되는 경우 중간 계산 중에 전체 자릿수 손실이 발생할 수 있습니다. 중간 결과의 전체 자릿수 및 소수 자릿수는 최종 결과의 전체 자릿수 또는 소수 자릿수에 영향을 미칠 수 있습니다.

많은 전체 자릿수가 비활성화된 경우 전체 자릿수 손실

많은 전체 자릿수를 비활성화하면 데이터 통합 서비스는 10진수 값을 배정밀도로 변환합니다. 10진수 값의 전체 자릿수가 15자리를 초과하면 전체 자릿수 손실이 발생합니다. 예를 들어 숫자 40012030304957666903을 전달하는 10진수(20,0)이 포함된 매핑이 있다고 가정합니다. 많은 전체 자릿수를 비활성화하면 데이터 통합 서비스가 10진수 값을 배정밀도 값으로 변환하고 $4.00120303049577 \times 10^{19}$ 를 전달합니다.

중간 계산이 있는 식을 변환에 사용하면 전체 자릿수가 추가로 손실될 수 있습니다. 많은 전체 자릿수를 비활성화하면 데이터 통합 서비스가 중간 결과를 배정밀도로 변환하고 작은 전체 자릿수를 사용하여 식에서 다음 계산

을 수행합니다. 따라서 데이터 통합 서비스에서 인수의 순서에 따라 식의 출력이 다르게 생성될 수 있습니다. 예를 들어 많은 전체 자릿수를 비활성화한 경우 $\tan(90 \times \pi / 180)$ 의 출력은 $\tan(90/180 \times \pi)$ 의 출력과 다릅니다. 데이터 통합 서비스는 이러한 식의 중간 계산을 다음과 같이 수행합니다.

$\tan(90 \times \pi / 180) = \tan(1.5707963267950003) = -9.649380295141232E12$

$\tan(90/180 \times \pi) = \tan(1.570796326795) = -9.670099380792184E12$

배정밀도 값의 반올림 방법

데이터베이스가 배정밀도 데이터 유형 계산을 처리하는 컴퓨터 시스템 및 시스템 런타임 라이브러리에서의 차이로 인해 결과가 예상과 다를 수 있습니다. 배정밀도 데이터 유형은 IEEE 794 표준을 따릅니다. 데이터베이스 클라이언트 라이브러리 변경, 데이터베이스의 버전 차이 또는 시스템 런타임 라이브러리의 변경은 수학적으로 동일한 값의 이진 표현에 영향을 줍니다. 또한 많은 시스템 런타임 라이브러리가 짝수로 반올림 또는 대칭적 산술 방법을 구현합니다. 짝수로 반올림 방법은 숫자가 다음의 높은 숫자 또는 낮은 숫자 사이에 있을 경우 짝수 최하위 비트의 가장 가까운 값으로 반올림하는 것을 나타냅니다. 예를 들어 짝수로 반올림 방법을 사용하면 0.125는 0.12로 반올림됩니다. 대칭적 산술 방법은 마지막 숫자가 5 이상일 경우 숫자를 다음의 높은 숫자로 반올림합니다. 예를 들어 대칭적 산술 방법을 사용하면 0.125는 0.13으로, 0.124는 0.12로 반올림됩니다.

플랫폼 차이에 영향을 더 적게 받는 계산 결과를 제공하기 위해 통합 서비스는 배정밀도 데이터 유형 값 중 15자리의 유효 자릿수를 저장합니다. 예를 들어 Windows에서 계산이 숫자 1234567890.1234567890을 반환하고 UNIX에서 동일한 계산이 1234567890.1234569999를 반환할 경우 통합 서비스는 이 숫자를 1234567890.1234600000으로 변환합니다.

String 데이터 유형

변환 데이터 유형에는 다음 String 데이터 유형이 포함됩니다.

- 문자열
- 텍스트

String 및 Text 데이터 유형은 최대 104,857,600자의 동일한 전체 자릿수를 지원하지만 통합 서비스는 String을 사용하여 문자열 데이터를 소스에서 대상으로 이동하고 Text를 사용하여 텍스트 데이터를 소스에서 대상으로 이동합니다. 일부 데이터베이스는 텍스트 데이터를 문자열 데이터와 다르게 저장하기 때문에 통합 서비스는 두 유형의 문자 데이터 간에 구분해야 합니다. 일반적으로 Char 및 Varchar 같은 작은 문자열 데이터 유형은 변환에서 String으로 표시되는 반면 Text, Long 및 Long Varchar 같은 큰 텍스트 데이터 유형은 Text로 표시됩니다.

문자열 및 텍스트를 변환 내에서 서로 바꿔 사용할 수 있습니다. 그러나 조회 변환에서는 대상 데이터 유형이 일치해야 합니다. 데이터베이스 드라이버에서 데이터를 정확하게 전달하려면 문자열 데이터 유형과 변환 데이터 유형이 일치해야 합니다. 예를 들어 조회 테이블의 Varchar는 조회 변환의 문자열과 일치해야 합니다.

복합 데이터 유형

복합 데이터 유형을 사용하면 변환의 단일 행 또는 열 위치에서 여러 데이터 값을 나타낼 수 있습니다. Spark 엔진에서 실행되는 매핑을 통해 계층 데이터를 직접 읽고, 처리하고, 쓰려면 복합 데이터 유형을 사용합니다.

계층 데이터를 처리하려면 매핑의 복합 포트에 복합 데이터 유형을 할당합니다. 계층 데이터의 복합 포트를 처리하는 포트를 호출합니다. Developer tool에서 포트 유형 구성의 복합 포트 데이터를 처리하는 데 사용할 복합 데이터 유형을 지정합니다.

Spark 엔진에서 실행되는 매핑의 변환에서 다음과 같은 복합 데이터 유형을 사용할 수 있습니다.

Array

배열은 요소의 정렬된 컬렉션입니다. 요소는 정수 또는 문자열 같은 기본 데이터 유형이거나 배열, 구조체 또는 맵 같은 복합 데이터 유형일 수 있습니다. 배열의 모든 요소는 동일한 데이터 유형이어야 합니다.

Map

맵에는 키-값 쌍의 정렬되지 않은 컬렉션이 포함됩니다. 값 부분은 기본 데이터 유형이거나 복합 데이터 유형일 수 있습니다. 값에는 단일 데이터 유형만 사용할 수 있습니다.

Struct

구조체는 데이터 유형이 서로 다른 요소의 컬렉션입니다. **Struct** 데이터 유형은 개념적으로 테이블 행과 유사합니다. 이 데이터 유형에는 데이터 유형이 미리 정의된 고정된 수의 명명된 필드가 포함됩니다.

복합 데이터 유형에는 기본 데이터 유형 또는 복합 데이터 유형이 포함될 수 있습니다. 복합 데이터 유형에 포함되는 문자 수에는 제한이 없습니다.

복합 포트에는 중첩된 데이터 유형을 사용할 수 있습니다. 중첩된 데이터 유형은 복합 데이터 유형 안에 다른 복합 데이터 유형이 포함된 데이터 유형입니다. 예를 들어 구조체의 배열을 생성하거나 다른 구조체의 배열을 포함하는 구조체를 생성할 수 있습니다. 중첩된 데이터 유형에는 최대 10개 수준의 중첩이 포함될 수 있습니다.

Array 데이터 유형

Array 데이터 유형은 요소의 정렬된 컬렉션을 나타냅니다. 배열 데이터를 전달, 생성 또는 처리하려면 포트에 **Array** 데이터 유형을 할당합니다.

배열은 0부터 시작하는 인덱스 목록입니다. 배열 인덱스는 배열 요소의 위치를 나타냅니다. 예를 들어 배열 인덱스 0은 배열의 첫 번째 요소를 나타냅니다. 변환 언어에는 배열 요소 및 함수에 액세스하여 배열 데이터를 생성하고 처리하는 연산자가 포함됩니다.

배열은 1차원이거나 다중 차원일 수 있습니다. 1차원 배열은 선형 배열입니다. 다중 차원 배열은 배열로 구성된 배열입니다. 배열의 변환 데이터 유형에는 최대 5개의 차원이 포함될 수 있습니다.

형식

`array <data_type> []`

다음 테이블에는 이 데이터 유형의 인수가 설명되어 있습니다.

인수	설명
array	배열 열 또는 포트의 이름입니다.
data_type	배열에 있는 요소의 데이터 유형입니다. 요소는 기본 데이터 유형이거나 복합 데이터 유형일 수 있습니다. 배열의 모든 요소는 동일한 데이터 유형이어야 합니다.
[]	배열의 차원이며 아래 첨자로 표시됩니다. 아래 첨자가 하나([])이면 1차원 배열을 나타냅니다. 아래 첨자가 둘([][])이면 2차원 배열을 나타냅니다. 각 차원의 요소는 동일한 데이터 유형입니다.

배열의 요소에는 이름이 없습니다. 배열의 요소 수는 행마다 다를 수 있습니다.

배열 예제

1차원 배열

다음 배열 열은 고객 전화 번호를 포함하는 문자열 요소의 1차원 배열을 나타냅니다.

```
custphone string[]
```

다음 예제는 `custphone` 열의 데이터 값을 보여 줍니다.

custphone

[205-128-6478,722-515-2889]

[107-081-0961,718-051-8116]

[107-031-0961,NULL]

2차원 배열

다음 배열 열은 고객 회사 및 개인 전자 메일 주소를 포함하는 문자열 요소의 2차원 배열을 나타냅니다.

`email_work_pers string[][]`

다음 예제는 `email_work_pers` 열의 데이터 값을 보여 줍니다.

email_work_pers

[john_baer@xyz.com,jbaer@xyz.com][john.baer@fgh.com,jbaer@ijk.com]

[bobbi_apperley@xyz.com,bapperl@xyz.com][apperlbob@fgh.com,bobbi@ijk.com]

[linda_bender@xyz.com,lbender@xyz.com][l.bender@fgh.com,NULL]

Map 데이터 유형

Map 데이터 유형은 키-값 쌍 요소의 정렬되지 않은 컬렉션을 나타냅니다. 맵 요소는 한 항목을 다른 항목에 매핑하는 키와 값의 쌍입니다. 구조체 데이터를 전달, 생성 또는 처리하려면 **Map** 데이터 유형을 포트에 할당합니다.

키는 기본 데이터 유형이어야 합니다. 값은 기본 또는 복합 데이터 유형일 수 있습니다. 복합 데이터 유형의 값을 포함하는 **Map** 데이터 유형은 중첩된 맵입니다. 중첩된 맵에는 최대 3개 수준의 **Map** 데이터 유형이 중첩될 수 있습니다.

참고: Spark 엔진은 맵 데이터에서 중복 키를 허용합니다.

변환 언어에는 맵 요소에 액세스하기 위한 아래 첨자 연산자가 포함됩니다. 맵 데이터를 생성하고 처리하는 함수도 포함됩니다.

형식

`map <primitive_type -> data_type>`

다음 테이블에는 이 데이터 유형의 인수가 설명되어 있습니다.

인수	설명
map	맵 열 또는 포트의 이름입니다.
primitive_type	맵 요소에 있는 키의 데이터 유형입니다. 키는 기본 데이터 유형이어야 합니다.
data_type	맵 요소에 있는 값의 데이터 유형입니다. 값은 기본 또는 복합 데이터 유형일 수 있습니다.

맵 예제

다음 맵 열은 정수 키와 문자열 값을 사용하여 고객 ID를 고객 이름에 매핑하는 맵 데이터를 나타냅니다.

```
custid_name <integer -> string>
```

다음 예제는 `custid_name` 열의 데이터 값을 보여 줍니다.

custid_name

```
<26745 -> 'John Baer'>
```

```
<56743 -> 'Bobbi Apperley'>
```

```
<32879 -> 'Linda Bender'>
```

Struct 데이터 유형

Struct 데이터 유형은 데이터 유형이 서로 다른 요소의 컬렉션을 나타냅니다. **Struct** 데이터 유형에는 데이터의 구조를 정의하는 연관된 스키마가 포함됩니다. 구조체 데이터를 전달, 생성 또는 처리하려면 **Struct** 데이터 유형을 포트에 할당합니다.

Struct 데이터 유형의 스키마는 구조체 데이터의 요소 이름과 요소 수를 결정합니다. 또한 스키마는 구조체 데이터의 요소 순서도 결정합니다. **Informatica**는 복합 데이터 유형 정의를 사용하여 구조체 데이터의 스키마를 나타냅니다.

변환 언어에는 구조체 요소에 액세스하는 연산자가 포함됩니다. 또한 구조체 데이터를 생성, 처리 및 구문 분석하고 데이터의 스키마를 수정하는 함수도 포함됩니다.

형식

```
struct {element_name1:value1 [, element_name2:value2, ...]}
```

구조체의 스키마는 다음과 같은 형식입니다.

```
schema {element_name1:data_type1 [, element_name2:data_type2, ...]}
```

다음 테이블에는 이 데이터 유형의 인수가 설명되어 있습니다.

인수	설명
struct	구조체 열 또는 포트의 이름입니다.
schema	데이터 구조의 정의입니다. 스키마는 구조체 요소의 이름과 데이터 유형을 결정하는 이름-유형 쌍입니다.
element_name	구조체 요소의 이름입니다.
value	구조체 요소의 값입니다.
data_type	요소 값의 데이터 유형입니다. 요소 값은 기본 또는 복합 데이터 유형일 수 있습니다. 구조체의 각 요소에는 다른 데이터 유형이 포함될 수 있습니다.

구조체 예제

다음 스키마는 고객 주소를 저장하는 구조체 데이터에 대한 것입니다.

```
address  
{st_number:integer,st_name:string,city:string,state:string,zip:string}
```

다음 예제는 `cust_address` 열의 구조체 데이터 값을 보여 줍니다.

`cust_address`

```
{st_number:154,st_name:Addison Ave,city:Redwood City,state:CA,zip:94065}
```

```
{st_number:204,st_name:Ellis St,city:Mountain View,state:CA,zip:94043}
```

```
{st_number:357,st_name:First St,city:Sunnyvale,state:CA,zip:94085}
```

복합 파일 및 변환 데이터 유형

복합 데이터 유형을 매핑에 사용하여 복합 파일의 계층 데이터를 처리할 수 있습니다.

Spark 및 Databricks Spark 엔진에서 실행되는 매핑의 다음과 같은 복합 파일에서 복합 데이터 유형을 사용할 수 있습니다.

- Avro
- JSON(JavaScript Object Notation)
- ORC(Optimized Row Columnar)
- Parquet

Avro 데이터 유형 및 변환 데이터 유형

Avro 데이터 유형은 데이터 통합 서비스가 플랫폼 간에 데이터를 이동하는 데 사용하는 변환 데이터 유형에 매핑됩니다.

다음 테이블에서는 데이터 통합 서비스에서 지원하는 Avro 데이터 유형과 해당하는 변환 데이터 유형을 비교합니다.

Avro 데이터 유형	변환 데이터 유형	범위
배열	배열	제한 없는 수의 문자입니다.
부울	정수	TRUE(1) 또는 FALSE(0)입니다.
바이트	이진	전체 자릿수 4000
날짜	날짜/시간	0001년 1월 1일 ~ 9999년 12월 31일
10진수	10진수	선언된 전체 자릿수와 소수 자릿수가 있는 10진수 값. 소수 자릿수는 전체 자릿수보다 작거나 같아야 합니다. 최대 38자리의 전체 자릿수를 지원하는 변환에서 전체 자릿수는 1 ~ 38자리이고 소수 자릿수는 0 ~ 38입니다. 최대 28자리의 전체 자릿수를 지원하는 변환에서 전체 자릿수는 1 ~ 28자리이고 소수 자릿수는 0 ~ 28입니다. 전체 자릿수를 최대 자릿수보다 많이 지정하는 경우 데이터 통합 서비스가 10진수 값을 많은 전체 자릿수 모드에서 두 배로 변환합니다.

Avro 데이터 유형	변환 데이터 유형	범위
배정밀도	배정밀도	전체 자릿수 15
고정됨	이진	1~104,857,600바이트.
부동 소수점 수	배정밀도	전체 자릿수 15.
Int	정수	-2,147,483,648 ~ 2,147,483,647 전체 자릿수 10 및 소수 자릿수 0
긴 정수	Bigint	-9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807. 전체 자릿수 19 및 소수 자릿수 0
매핑	매핑	제한 없는 수의 문자입니다.
레코드	구조	제한 없는 수의 문자입니다.
문자열	문자열	1 ~ 104,857,600자.
시간	날짜/시간	하루 중의 시간입니다. 전체 자릿수는 밀리초까지입니다.
타임스탬프	날짜/시간	0001년 1월 1일 00:00:00 ~ 9999년 12월 31일 23:59:59.997. 전체 자릿수는 밀리초까지입니다.
합집합	["primitive_type complex_type", "null"] 또는 ["null", "primitive_type complex_type"]의 합집합에서 해당하는 데이터 유형입니다.	기본 또는 복합 데이터 유형에 종속됩니다.

Avro 합집합 데이터 유형

합집합은 필드에 둘 이상의 데이터 유형이 포함될 수 있음을 나타냅니다. 예를 들어 필드가 문자열이거나 null일 수 있음을 나타내는 합집합이 있습니다. 합집합은 해당 데이터 유형을 포함하는 JSON 배열로 표현됩니다. Developer tool은 ["primitive_type|complex_type", "null"] 또는 ["null", "primitive_type|complex_type"]의 합집합만 해석합니다. Avro 데이터 유형은 해당하는 변환 데이터 유형으로 변환됩니다.

Avro 타임스탬프 데이터 유형 지원

다음 테이블에는 Avro 파일 형식에 대한 타임스탬프 데이터 유형 지원이 나열되어 있습니다.

타임스탬프 데이터 유형	원시	Spark
Timestamp_micros	예	예
Timestamp_millis	예	아니요
Time_millis	예	아니요
Time_micros	예	아니요

지원되지 않는 Avro 데이터 유형

Developer tool은 다음과 같은 Avro 데이터 유형을 지원하지 않습니다.

- 열거
- Null
- Timestamp_tz

JSON 데이터 유형 및 변환 데이터 유형

JSON 데이터 유형은 데이터 통합 서비스가 플랫폼 간에 데이터를 이동하는 데 사용하는 변환 데이터 유형에 매핑됩니다.

다음 테이블에서는 데이터 통합 서비스에서 지원하는 JSON 데이터 유형과 해당하는 변환 데이터 유형을 비교합니다.

JSON	변환	범위
Array	Array	제한 없는 수의 문자입니다.
배정밀도	배정밀도	전체 자릿수 15자리.
정수	정수	-2,147,483,648 ~ 2,147,483,647. 전체 자릿수는 10자리, 소수 자릿수는 0자리입니다.
Object	Struct	제한 없는 수의 문자입니다.
문자열	문자열	1 ~ 104,857,600자.

지원되지 않는 JSON 데이터 유형

Developer tool은 다음과 같은 JSON 데이터 유형을 지원하지 않습니다.

- 날짜
- 10진수
- 타임스탬프
- 열거
- 합집합

ORC 데이터 유형 및 변환 데이터 유형

ORC 파일 데이터 유형은 데이터 통합 서비스가 플랫폼 간에 데이터를 이동하는 데 사용하는 변환 데이터 유형에 매핑됩니다.

다음 테이블에는 데이터 통합 서비스에서 지원하는 ORC 파일 데이터 유형과 해당하는 변환 데이터 유형이 나열되어 있습니다.

ORC 파일 데이터 유형	변환 데이터 유형	범위 및 설명
BigInt	BigInt	-9223372036854775808 ~ 9,223,372,036,854,775,807.
부울	정수	TRUE(1) 또는 FALSE(0)입니다.
문자	문자열	1 ~ 104,857,600자.
날짜	날짜/시간	0001년 1월 1일 ~ 9999년 12월 31일
배정밀도	배정밀도	전체 자릿수는 15자리입니다.
부동 소수점 수	배정밀도	정밀도는 15자리입니다.
정수	정수	-2,147,483,648 ~ 2,147,483,647.
SmallInt	정수	-32,768 ~ 32,767.
문자열	문자열	1 ~ 104,857,600자.
타임스탬프	날짜/시간	0001년 1월 1일 00:00:00 ~ 9999년 12월 31일 23:59:59.997. 전체 자릿수는 밀리초까지입니다.
TinyInt	정수	-128 ~ 127.
Varchar	문자열	1 ~ 104,857,600자.

Spark 또는 Databricks Spark 엔진에서 매핑을 실행하여 ORC 파일을 대상에 쓸 때 데이터 통합 서비스에서는 Char 및 Varchar 데이터 유형의 데이터를 문자열로 씁니다.

참고: ORC 데이터 유형을 사용하여 Spark 엔진에서만 실행되는 매핑에서 복합 파일 개체를 읽고 쓸 수 있습니다.

지원되지 않는 ORC 데이터 유형

Developer tool은 다음과 같은 ORC 데이터 유형을 지원하지 않습니다.

- 매핑
- 목록
- 구조
- 합집합

Parquet 데이터 유형 및 변환 데이터 유형

Parquet 데이터 유형은 데이터 통합 서비스가 플랫폼 간에 데이터를 이동하는 데 사용하는 변환 데이터 유형에 매핑됩니다.

다음 테이블에서는 데이터 통합 서비스에서 지원하는 Parquet 데이터 유형과 해당하는 변환 데이터 유형을 비교합니다.

Parquet	변환	범위
이진	이진	1~104,857,600바이트
Binary (UTF8)	문자열	1 ~ 104,857,600자
부울	정수	-2,147,483,648 ~ 2,147,483,647 전체 자릿수 10, 소수 자릿수 0
날짜	날짜/시간	0001년 1월 1일 ~ 9999년 12월 31일
10진수	10진수	선언된 전체 자릿수와 소수 자릿수가 있는 10진수 값. 소수 자릿수는 전체 자릿수보다 작거나 같아야 합니다. 최대 38자리의 전체 자릿수를 지원하는 변환에서 전체 자릿수는 1 ~ 38자리이고 소수 자릿수는 0 ~ 38입니다. 최대 28자리의 전체 자릿수를 지원하는 변환에서 전체 자릿수는 1 ~ 28자리이고 소수 자릿수는 0 ~ 28입니다. 전체 자릿수를 최대 자릿수보다 많이 지정하는 경우 데이터 통합 서비스가 10진수 값을 많은 전체 자릿수 모드에서 두 배로 변환합니다.
배정밀도	배정밀도	전체 자릿수 15자리.
부동 소수점 수	배정밀도	전체 자릿수 15자리.
Int32	정수	-2,147,483,648 ~ 2,147,483,647 전체 자릿수 10, 소수 자릿수 0
Int64	Bigint	-9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807 전체 자릿수 19, 소수 자릿수 0
매핑	매핑	제한 없는 수의 문자입니다.
Struct	Struct	제한 없는 수의 문자입니다.
시간	날짜/시간	하루 중의 시간입니다. 전체 자릿수는 밀리초까지입니다.

Parquet	변환	범위
타임스탬프	날짜/시간	0001년 1월 1일 00:00:00 ~ 9999년 12월 31일 23:59:59.997. 전체 자릿수는 밀리초까지입니다.
group (LIST)	Array	제한 없는 수의 문자입니다.

Parquet 파일 읽기 또는 쓰기를 지정하는 Parquet 스키마는 소문자여야 합니다. Parquet은 대/소문자 구분 스키마를 지원하지 않습니다.

Parquet 타임스탬프 데이터 유형 지원

다음 테이블에는 Parquet 파일 형식에 대한 타임스탬프 데이터 유형 지원이 나열되어 있습니다.

타임스탬프 데이터 유형	원시	Spark
Timestamp_micros	예	아니요
Timestamp_millis	예	아니요
Time_millis	예	아니요
Time_micros	예	아니요
int96	예	예

지원되지 않는 Parquet 데이터 유형

Developer tool은 다음과 같은 Parquet 데이터 유형을 지원하지 않습니다.

- Timestamp_nanos
- Time_nanos
- Timestamp_tz

Rules and Guidelines for Data Types

Consider the following rules and guidelines for data types:

- Avro data types support:
 - Date, Decimal, and Timestamp data types are applicable when you run a mapping in the native environment or on the Spark engine in Cloudera CDH 6.1 distribution.
 - Time data type is applicable when you run a mapping in the native environment in Cloudera CDH 6.1 distribution.
- JSON data types support:
 - For PowerExchange for Microsoft Azure Data Lake Storage Gen2, you can read and write complex file objects in JSON format in mappings that run in the native environment, Spark engine, and Databricks Spark engine.
 - For other file-based adapters, you can read and write complex file objects in JSON format in mappings that run on the Spark engine only.

- Parquet data types support:
 - Date, Time, and Timestamp data types till microseconds are applicable when you run a mapping in the native environment, Blaze, and Spark engine in the Cloudera CDH 6.1, Hortonworks HDP 3.1, Azure HDInsight HDI 4.0, and Cloudera CDP 7.1 distributions.
 - Date, Time_Millis and Timestamp_Millis data types are applicable when you run a mapping in the native environment or Spark engine in the Amazon Elastic MapReduce (EMR) 5.23, MapR 6.1, and Hortonworks HDP 2.6 distributions.
 - Decimal data types are applicable when you run a mapping in the native environment and Spark engine in Cloudera CDH 6.1, Cloudera CDH 6.3, Hortonworks HDP 2.6, Hortonworks HDP 3.1, Amazon EMR 5.20, Amazon EMR 5.23, MapR 6.1, Google Dataproc 1.4, and Azure HDInsight HDI 4.0 distributions.
 - Date, Time, Timestamp, and Decimal data types are applicable when you run a mapping on the Databricks Spark engine.
 - When you run a mapping and use Date data type that does not have a time value, the Data Integration Service adds the time value, based on the time zone, to the date in the target.
For example, Date data type used in the source:
1980-01-09
Value generated in the target:
1980-01-09 00:00:00
 - When you run a mapping in the native environment and use Time data type in the source, the Data Integration Service writes incorrect date value to the target.
For example, Time data type used in the source:
1980-01-09 06:56:01.365235000
Incorrect Date value is generated in the target:
1899-12-31 06:56:01.365235000
 - When you run a mapping in the native environment and use Date data type in the source, the Data Integration Service writes incorrect time value to the target.
For example, Date data type used in the source:
1980-01-09 00:00:00
Incorrect Time value generated in the target:
1980-01-09 05:30:00
- To run a mapping that reads and writes Date, Time, Timestamp, and Decimal data types, update the -DINFA_HADOOP_DIST_DIR option to the developerCore.ini file. The developerCore.ini file is located in the following directory:
<Client installation directory>\clients\DeveloperClient\
Add the following path to the developerCore.ini file:
-DINFA_HADOOP_DIST_DIR=hadoop\CDH_6.1
- To use precision up to 38 digits for Decimal data type in the native environment, set the EnableSDKDecimal38 custom property to true for the Data Integration Service. The EnableSDKDecimal38 custom property is applicable to all file-based PowerExchange adapters except PowerExchange for HDFS.

플랫 파일 및 변환 데이터 유형

플랫 파일 데이터 유형은 데이터 통합 서비스가 플랫폼 간에 데이터를 이동하는 데 사용하는 변환 데이터 유형에 매핑됩니다.

다음 테이블에서는 플랫 파일 데이터 유형과 변환 데이터 유형을 비교합니다.

플랫 파일 데이터 유형	변환 데이터 유형	범위
Bigint	Bigint	전체 자릿수 19, 소수 자릿수 0
날짜/시간	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일(전체 자릿수는 나노초)
배정밀도	배정밀도	전체 자릿수 15자리
Int	정수	-2,147,483,648 ~ 2,147,483,647
Nstring	문자열	1 ~ 104,857,600자
숫자	10진수	<p>최대 38자리의 전체 자릿수를 지원하는 변환에서 전체 자릿수는 1 ~ 38자리이고 소수 자릿수는 0 ~ 38입니다.</p> <p>최대 28자리의 전체 자릿수를 지원하는 변환에서 전체 자릿수는 1 ~ 28자리이고 소수 자릿수는 0 ~ 28입니다.</p> <p>전체 자릿수를 최대 자릿수보다 많게 지정하는 경우 데이터 통합 서비스가 10진수 값을 많은 전체 자릿수 모드에서 두 배로 변환합니다.</p> <p>전체 자릿수가 15자리보다 많은 경우 데이터 통합 서비스는 10진수 값을 낮은 전체 자릿수 모드의 배정밀도로 변환합니다.</p>
문자열	문자열	1 ~ 104,857,600자
TimestampWithTZ	TimestampWithTZ	<p>A.D. 1947년 8월 1일 ~ A.D. 2040년 12월 31일 -12:00 ~ +14:00</p> <p>전체 자릿수 36, 소수 자릿수 9 (나노초까지의 전체 자릿수)</p> <p>시간대가 포함된 타임스탬프 데이터 유형은 다음과 같은 시간대 영역을 지원하지 않습니다.</p> <ul style="list-style-type: none"> - AFRICA_CAIRO - AFRICA_MONROVIA - EGYPT - AMERICA_MONTREAL <p>참고: TimestampWithTZ는 원시 환경에만 적용할 수 있습니다.</p>

데이터 통합 서비스는 플랫 파일의 숫자 열에서 숫자가 아닌 데이터를 읽을 경우 행을 삭제하고 로그에 메시지를 기록합니다. 또한 데이터 통합 서비스는 플랫 파일의 날짜/시간 열에서 날짜/시간이 아닌 데이터를 읽을 경우에도 행을 삭제하고 로그에 메시지를 기록합니다.

DB2 for LUW 및 변환 데이터 유형

DB2 for LUW 데이터 유형은 데이터 통합 서비스가 플랫폼 간에 데이터를 이동하는 데 사용하는 변환 데이터 유형에 매핑됩니다.

다음 테이블에서는 DB2 for LUW 데이터 유형과 변환 데이터 유형을 비교합니다.

데이터 유형	범위	변환	범위
Bigint	-9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807	Bigint	-9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807 전체 자릿수 19, 소수 자릿수 0
Blob	1 ~ 2,147,483,647바이트	이진	1~104,857,600바이트
문자	1 ~ 254자	문자열	1 ~ 104,857,600자
Char for bit data	1 ~ 254바이트	이진	1~104,857,600바이트
Clob	1 ~ 2,447,483,647바이트	텍스트	1 ~ 104,857,600자
날짜	A.D. 0001 ~ 9999년 전체 자릿수 19, 소수 자릿수 0(일까지의 전체 자릿수)	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일 (나노초까지의 전체 자릿수)
Dbclob	1 ~ 1073741823바이트	Dbclob	1~104,857,600바이트
10진수	전체 자릿수 1 ~ 31, 소수 자릿수 0 ~ 31	10진수	전체 자릿수 1 ~ 31, 소수 자릿수 0 ~ 31
부동 소수점 수	전체 자릿수 1 ~ 15	배정밀도	전체 자릿수 15
그래픽	1 ~ 127바이트	그래픽	1 ~ 127바이트
정수	-2,147,483,648 ~ 2,147,483,647	정수	-2,147,483,648 ~ 2,147,483,647 전체 자릿수 10, 소수 자릿수 0
긴 Varchar	32700바이트	긴 Varchar	32700바이트
긴 Vargraphic	163350바이트	긴 Vargraphic	163350바이트
숫자	전체 자릿수 1 ~ 31, 소수 자릿수 0 ~ 31	10진수	1 ~ 28 전체 자릿수, 배율 0 ~ 28
Smallint	-32,768 ~ 32,767	정수	-2,147,483,648 ~ 2,147,483,647 전체 자릿수 10, 소수 자릿수 0
Time ¹	24시간 기간 전체 자릿수 19, 소수 자릿수 0 (초까지의 전체 자릿수)	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일 (나노초까지의 전체 자릿수)
타임스탬프	26바이트 전체 자릿수 26, 소수 자릿수 6 (마이크로초까지의 전체 자릿수)	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일 (나노초까지의 전체 자릿수)

데이터 유형	범위	변환	범위
Varchar	최대 4,000자	문자열	1 ~ 104,857,600자
Varchar for bit data	최대 4,000바이트	이진	1~104,857,600바이트
Vargraphic	1 ~ 16336바이트	Vargraphic	1 ~ 16336바이트
1. 데이터 통합 서비스가 DB2 Time 데이터 유형을 변환 Date/Time 데이터 유형으로 변환하면 날짜 값이 시간 값에 추가됩니다.			

DB2 for i5/OS, DB2 for z/OS 및 변환 데이터 유형

DB2 for i5/OS 및 DB2 for z/OS 데이터 유형은 IBM DB2 데이터 유형이 변환 데이터 유형과 매핑되는 것과 동일한 방식으로 변환 데이터 유형에 매핑됩니다. 데이터 통합 서비스는 변환 데이터 유형을 사용하여 플랫폼 간에 데이터를 이동합니다.

다음 표에서는 DB2 for i5/OS 및 DB2 for z/OS 데이터 유형과 변환 데이터 유형을 비교합니다.

데이터 유형	범위	변환	범위
Bigint	-9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807	Bigint	-9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807 전체 자릿수 19, 소수 자릿수 0
문자	1 ~ 254자	문자열	1 ~ 104,857,600자
Char for bit data	1 ~ 254바이트	이진	1~104,857,600바이트
날짜	A.D. 0001 ~ 9999년 전체 자릿수 19, 소수 자릿수 0(일까지의 전체 자릿수)	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일 (나노초까지의 전체 자릿수)
10진수	전체 자릿수 1 ~ 31, 소수 자릿수 0 ~ 31	10진수	전체 자릿수 1 ~ 31, 소수 자릿수 0 ~ 31
부동 소수점 수	전체 자릿수 1 ~ 15	배정밀도	전체 자릿수 15
정수	-2,147,483,648 ~ 2,147,483,647	정수	-2,147,483,648 ~ 2,147,483,647 전체 자릿수 10, 소수 자릿수 0
Smallint	-32,768 ~ 32,767	정수	-2,147,483,648 ~ 2,147,483,647 전체 자릿수 10, 소수 자릿수 0
Time ¹	24시간 기간 전체 자릿수 19, 소수 자릿수 0 (초까지의 전체 자릿수)	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일 (나노초까지의 전체 자릿수)

데이터 유형	범위	변환	범위
Timestamp ²	26바이트 전체 자릿수 26, 소수 자릿수 6 (마이크로초까지의 전체 자릿수)	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일 (나노초까지의 전체 자릿수)
Varchar	최대 4,000자	문자열	1 ~ 104,857,600자
Varchar for bit data	최대 4,000바이트	이진	1~104,857,600바이트
<p>1. 데이터 통합 서비스가 DB2 Time 데이터 유형을 변환 Date/Time 데이터 유형으로 변환하면 날짜 값이 시간 값에 추가됩니다.</p> <p>2. DB2 for z/OS 버전 10의 확장된 전체 자릿수 타임스탬프는 다음과 같이 변환 데이터 유형에 매핑됩니다.</p> <ul style="list-style-type: none"> - 소수 자릿수가 6이면 전체 자릿수는 26이고 변환 데이터 유형은 날짜/시간입니다. - 소수 자릿수가 0이면 전체 자릿수는 19이고 변환 데이터 유형은 문자열입니다. - 소수 자릿수가 1 ~ 5 또는 7 ~ 12이면 전체 자릿수는 20 + 소수 자릿수이고 변환 데이터 유형은 문자열입니다. 			

지원되지 않는 DB2 for i5/OS 및 DB2 for z/OS 데이터 유형

Developer 도구는 특정한 DB2 for i5/OS 데이터 유형과 DB2 for z/OS 데이터 유형을 지원하지 않습니다.

Developer 도구는 DB2 for i5/OS 및 DB2 for z/OS LOB(대형 개체) 데이터 유형을 지원하지 않습니다. LOB 열은 관계형 테이블 개체에서 지원되지 않음으로 나타나며 기본 유형은 **varchar**이고 전체 자릿수와 소수 자릿수가 0입니다. 이러한 열은 매핑의 출력 또는 사용자 지정된 데이터 개체로 예상되지 않습니다.

JDBC 및 변환 데이터 유형

데이터 통합 서비스는 JDBC 소스에서 데이터를 읽을 경우 기본 데이터 유형을 해당 JDBC 데이터 유형으로 변환한 후 변환 데이터 유형으로 변환합니다. 데이터 통합 서비스는 변환 데이터 유형을 사용하여 플랫폼 간에 데이터를 이동합니다.

다음 표에서는 JDBC 데이터 유형과 변환 데이터 유형을 비교합니다.

JDBC 데이터 유형	변환	범위
Bigint	Bigint	-9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807 정밀도 19, 배율 0
이진*	이진	1~104,857,600바이트
비트	정수	-2,147,483,648 ~ 2,147,483,647 정밀도 10, 배율 0
Blob*	이진	1~104,857,600바이트
부울	정수	-2,147,483,648 ~ 2,147,483,647 정밀도 10, 배율 0

JDBC 데이터 유형	변환	범위
Char*	문자열	1 ~ 104,857,600자
Clob*	텍스트	1 ~ 104,857,600자
날짜	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일 (나노초까지의 전체 자릿수)
10진수	10진수	전체 자릿수 1 ~ 28, 소수 자릿수 0 ~ 28
배정밀도	배정밀도	전체 자릿수 15
부동 소수점 수	배정밀도	전체 자릿수 15
정수	정수	-2,147,483,648 ~ 2,147,483,647 정밀도 10, 배율 0
긴 VarBinary*	이진	1~104,857,600바이트
긴 Varchar*	텍스트	1 ~ 104,857,600자
숫자	10진수	전체 자릿수 1 ~ 28, 소수 자릿수 0 ~ 28
실수	배정밀도	전체 자릿수 15
Smallint	정수	-2,147,483,648 ~ 2,147,483,647 정밀도 10, 배율 0
시간	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일 (나노초까지의 전체 자릿수)
타임스탬프	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일 (나노초까지의 전체 자릿수)
Tinyint	정수	-2,147,483,648 ~ 2,147,483,647 정밀도 10, 배율 0
Varchar*	문자열	1 ~ 104,857,600자
Varbinary*	이진	1~104,857,600바이트
*포트의 데이터 크기가 100MB보다 클 경우 Developer 도구는 기본적으로 포트 전체 자릿수를 4000으로 설정합니다. 큰 크기의 데이터를 처리하려면 포트 전체 자릿수를 늘리십시오.		

Microsoft SQL Server 및 변환 데이터 유형

Microsoft SQL Server 데이터 유형은 데이터 통합 서비스가 플랫폼 간에 데이터를 이동하는 데 사용하는 변환 데이터 유형에 매핑됩니다.

다음 테이블에서는 Microsoft SQL Server 데이터 유형과 변환 데이터 유형을 비교합니다.

Microsoft SQL Server	범위	변환	범위
이진	1 ~ 8,000바이트	이진	1~104,857,600바이트
bigint	-9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807	bigint	-9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807 전체 자릿수 19, 소수 자릿수 0
비트	1비트	문자열	1 ~ 104,857,600자
문자	1 ~ 8,000자	문자열	1 ~ 104,857,600자
날짜	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일
날짜/시간	A.D. 1753년 1월 1일 ~ A.D. 9999년 12월 31일 전체 자릿수 23, 소수 자릿수 3 (3.33밀리초까지의 전체 자릿수)	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일 (나노초까지의 전체 자릿수)
Datetime2	A.D. 0001년 1월 1일 00:00:00 ~ A.D. 9999년 12월 31일 23:59:59.9999999	타임스탬프	전체 자릿수 22 ~ 27
10진수	전체 자릿수 1 ~ 38, 소수 자릿수 0 ~ 38	10진수	최대 38자리의 전체 자릿수를 지원하는 변환에서 전체 자릿수는 1 ~ 38자리이고 소수 자릿수는 0 ~ 38입니다. 전체 자릿수를 최대 자릿수보다 많게 지정하는 경우 데이터 통합 서비스가 10진수 값을 많은 전체 자릿수 모드에서 두 배로 변환합니다.
부동 소수점 수	-1.79E+308 ~ 1.79E+308	배정밀도	전체 자릿수 15
이미지	1 ~ 2,147,483,647바이트	이진	1~104,857,600바이트
Int	-2,147,483,648 ~ 2,147,483,647	정수	-2,147,483,648 ~ 2,147,483,647 전체 자릿수 10, 소수 자릿수 0
통화	-922,337,203,685,477.5807 ~ 922,337,203,685,477.5807	10진수	1 ~ 28 전체 자릿수, 배율 0 ~ 28
nchar	1~4000자	문자열	1 ~ 104,857,600자

Microsoft SQL Server	범위	변환	범위
ntext	1~1,073,741,823바이트	텍스트	1 ~ 104,857,600자
숫자	전체 자릿수 1 ~ 38, 소수 자릿수 0 ~ 38	10진수	<p>최대 38자리의 전체 자릿수를 지원하는 변환에서 전체 자릿수는 1 ~ 38자리이고 소수 자릿수는 0 ~ 38입니다.</p> <p>최대 28자리의 전체 자릿수를 지원하는 변환에서 전체 자릿수는 1 ~ 28자리이고 소수 자릿수는 0 ~ 28입니다.</p> <p>전체 자릿수를 최대 자릿수보다 많게 지정하는 경우 데이터 통합 서비스가 10진수 값을 많은 전체 자릿수 모드에서 두 배로 변환합니다.</p>
nvarchar	1~4000자	문자열	1 ~ 104,857,600자
실수	-3.40E+38 ~ 3.40E+38	배정밀도	전체 자릿수 15
Smalldatetime	1900년 1월 1일 ~ 2079년 6월 6일 전체 자릿수 19, 소수 자릿수 0 (분까지의 전체 자릿수)	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일(전체 자릿수는 나노초)
Smallint	-32,768 ~ 32,768	정수	-2,147,483,648 ~ 2,147,483,647 전체 자릿수 10, 소수 자릿수 0
Smallmoney	-214,748.3648 ~ 214,748.3647	10진수	1 ~ 28 전체 자릿수, 배율 0 ~ 28
Sysname	1 ~ 128자	문자열	1 ~ 104,857,600자
텍스트	1 ~ 2,147,483,647자	텍스트	1 ~ 104,857,600자
타임스탬프	8바이트	이진	1~104,857,600바이트
Tinyint	0 ~ 255	정수	-2,147,483,648 ~ 2,147,483,647 전체 자릿수 10, 소수 자릿수 0
Uniqueidentifier	전체 자릿수 38, 소수 자릿수 0	문자열	<p>Uniqueidentifier 데이터를 성공적으로 이동 또는 변경하려면 데이터가 다음 형식인지 확인합니다.</p> <p>XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX</p> <p>여기서 x는 0-9 또는 a-f 범위의 16진수입니다.</p>
Varbinary	1 ~ 8,000바이트	이진	1~104,857,600바이트
Varchar	1 ~ 8,000자	문자열	1 ~ 104,857,600자

Uniqueidentifier 데이터 유형

Uniqueidentifier는 GUID(Globally Unique Identifier)를 저장하는 데 사용되는 Microsoft SQL Server 데이터 유형입니다. 16바이트의 데이터를 저장할 수 있습니다.

Developer 도구는 Uniqueidentifier 데이터 유형을 문자열로 처리합니다. Uniqueidentifier 데이터를 이동 또는 변경하려면 Uniqueidentifier 열을 문자열 열에 연결합니다. Uniqueidentifier 데이터를 성공적으로 이동 또는 변경하려면 데이터가 다음 형식인지 확인합니다.

```
XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX
```

여기서 x는 0-9 또는 a-f 범위의 16진수입니다.

예를 들어 6F9619FF-8B86-D011-B42D-00C04FC964FF는 올바른 Uniqueidentifier 값입니다.

Developer 도구는 16바이트의 Uniqueidentifier 데이터를 36자로 저장할 수 있습니다. 하지만 Uniqueidentifier 데이터는 두 개의 중괄호 내에서 표현될 수 있으므로 Developer 도구는 이러한 중괄호를 수용하기 위해 전체 자릿수에 두 개의 추가 문자를 할당합니다. Uniqueidentifier 열을 문자열 열에 연결할 경우 Uniqueidentifier 데이터를 성공적으로 이동 또는 변경하기 위해 문자열 열의 전체 자릿수를 38로 설정합니다.

비관계형 및 변환 데이터 유형

비관계형 데이터 유형은 데이터 통합 서비스가 플랫폼 간에 데이터를 이동하는 데 사용하는 변환 데이터 유형에 매핑됩니다.

비관계형 데이터 유형은 다음과 같은 유형의 연결에 적용됩니다.

- Adabas
- IMS
- 순차
- VSAM

다음 표에서는 관계형 데이터 유형과 변환 데이터 유형을 비교합니다.

비관계형	전체 자릿수	변환	범위
BIN	10	이진	1~104,857,600바이트
CHAR	10	문자열	1 ~ 104,857,600자 고정 길이 또는 변경되는 길이 문자열입니다.
DATE	10	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일 결합된 날짜/시간 값(나노초까지의 전체 자릿수)
DOUBLE	18	배정밀도	전체 자릿수 15자리 배정밀도 부동 소수점 숫자 값.
FLOAT	7	배정밀도	전체 자릿수 15자리 배정밀도 부동 소수점 숫자 값.

비관계형	전체 자릿수	변환	범위
NUM8	3	정수	전체 자릿수 10, 소수 자릿수 0 정수 값.
NUM8U	3	정수	전체 자릿수 10, 소수 자릿수 0 정수 값.
NUM16	5	정수	전체 자릿수 10, 소수 자릿수 0 정수 값.
NUM16U	5	정수	전체 자릿수 10, 소수 자릿수 0 정수 값.
NUM32	10	정수	전체 자릿수 10, 소수 자릿수 0 정수 값.
NUM32U	10	배정밀도	전체 자릿수 15자리 배정밀도 부동 소수점 숫자 값.
NUM64	19	10진수	정밀도 1 ~ 28자릿수, 배율 0 ~ 28 선언된 전체 자릿수와 소수 자릿수가 있는 10진수 값. 소수 자릿수는 전체 자릿수보다 작거나 같아야 합니다. 소수 자릿수가 음수이거나 선언된 전체 자릿수가 28보다 큰 값을 전달하면 데이터 통합 서비스가 값을 배정밀도로 변환합니다.
NUM64U	19	10진수	정밀도 1 ~ 28자릿수, 배율 0 ~ 28 선언된 전체 자릿수와 소수 자릿수가 있는 10진수 값. 소수 자릿수는 전체 자릿수보다 작거나 같아야 합니다. 소수 자릿수가 음수이거나 선언된 전체 자릿수가 28보다 큰 값을 전달하면 데이터 통합 서비스가 값을 배정밀도로 변환합니다.
NUMCHAR	100	문자열	1 ~ 104,857,600자 고정 길이 또는 변경되는 길이 문자열입니다.
PACKED	15	10진수	정밀도 1 ~ 28자릿수, 배율 0 ~ 28 선언된 전체 자릿수와 소수 자릿수가 있는 10진수 값. 소수 자릿수는 전체 자릿수보다 작거나 같아야 합니다. 소수 자릿수가 음수이거나 선언된 전체 자릿수가 28보다 큰 값을 전달하면 데이터 통합 서비스가 값을 배정밀도로 변환합니다.
TIME	5	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일 결합된 날짜/시간 값(나노초까지의 전체 자릿수)
TIMESTAMP	5	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일 결합된 날짜/시간 값(나노초까지의 전체 자릿수)

비관계형	전체 자릿수	변환	범위
UPACKED	15	10진수	정밀도 1 ~ 28자릿수, 배율 0 ~ 28 선언된 전체 자릿수와 소수 자릿수가 있는 10진수 값. 소수 자릿수는 전체 자릿수보다 작거나 같아야 합니다. 소수 자릿수가 음수이거나 선언된 전체 자릿수가 28보다 큰 값을 전달하면 데이터 통합 서비스가 값을 배정밀도로 변환합니다.
UZONED	15	10진수	정밀도 1 ~ 28자릿수, 배율 0 ~ 28 선언된 전체 자릿수와 소수 자릿수가 있는 10진수 값. 소수 자릿수는 전체 자릿수보다 작거나 같아야 합니다. 소수 자릿수가 음수이거나 선언된 전체 자릿수가 28보다 큰 값을 전달하면 데이터 통합 서비스가 값을 배정밀도로 변환합니다.
VARBIN	10	이진	1~104,857,600바이트
VARCHAR	10	문자열	1 ~ 104,857,600자 고정 길이 또는 변경되는 길이 문자열입니다.
ZONED	15	10진수	정밀도 1 ~ 28자릿수, 배율 0 ~ 28 선언된 전체 자릿수와 소수 자릿수가 있는 10진수 값. 소수 자릿수는 전체 자릿수보다 작거나 같아야 합니다. 소수 자릿수가 음수이거나 선언된 전체 자릿수가 28보다 큰 값을 전달하면 데이터 통합 서비스가 값을 배정밀도로 변환합니다.

ODBC 및 변환 데이터 유형

ODBC 데이터 유형은 데이터 통합 서비스가 플랫폼 간에 데이터를 이동하는 데 사용하는 변환 데이터 유형에 매핑됩니다.

다음 표에서는 Microsoft Access 또는 Excel 등의 ODBC 데이터 유형과 변환 데이터 유형을 비교합니다.

데이터 유형	변환	범위
Bigint	Bigint	-9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807 전체 자릿수 19, 소수 자릿수 0
이진	이진	1~104,857,600바이트
비트	문자열	1 ~ 104,857,600자
문자	문자열	1 ~ 104,857,600자
날짜	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일 (나노초까지의 전체 자릿수)

데이터 유형	변환	범위
10진수	10진수	최대 38자리의 전체 자릿수를 지원하는 변환에서 전체 자릿수는 1 ~ 38자리이고 소수 자릿수는 0 ~ 38입니다. 전체 자릿수를 최대 자릿수보다 많이 지정하는 경우 데이터 통합 서비스가 10진수 값을 많은 전체 자릿수 모드에서 두 배로 변환합니다.
배정밀도	배정밀도	전체 자릿수 15
부동 소수점 수	배정밀도	전체 자릿수 15
정수	정수	-2,147,483,648 ~ 2,147,483,647 전체 자릿수 10, 소수 자릿수 0
긴 Varbinary	이진	1~104,857,600바이트
Nchar	문자열	1 ~ 104,857,600자
Nvarchar	문자열	1 ~ 104,857,600자
Ntext	텍스트	1 ~ 104,857,600자
숫자	10진수	최대 38자리의 전체 자릿수를 지원하는 변환에서 전체 자릿수는 1 ~ 38자리이고 소수 자릿수는 0 ~ 38입니다. 전체 자릿수를 최대 자릿수보다 많이 지정하는 경우 데이터 통합 서비스가 10진수 값을 많은 전체 자릿수 모드에서 두 배로 변환합니다.
실수	배정밀도	전체 자릿수 15
Smallint	정수	-2,147,483,648 ~ 2,147,483,647 전체 자릿수 10, 소수 자릿수 0
텍스트	텍스트	1 ~ 104,857,600자
시간	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일 (나노초까지의 전체 자릿수)
타임스탬프	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일 (나노초까지의 전체 자릿수)
Tinyint	정수	-2,147,483,648 ~ 2,147,483,647 전체 자릿수 10, 소수 자릿수 0
Varbinary	이진	1~104,857,600바이트
Varchar	문자열	1 ~ 104,857,600자

Oracle 및 변환 데이터 유형

Oracle 데이터 유형은 데이터 통합 서비스가 플랫폼 간에 데이터를 이동하는 데 사용하는 변환 데이터 유형에 매핑됩니다.

다음 테이블에서는 Oracle 데이터 유형과 변환 데이터 유형을 비교합니다.

Oracle	범위	변환	범위
Blob	최대 4GB	이진	1~104,857,600바이트
문자(L)	1 ~ 2,000바이트	문자열	1 ~ 104,857,600자
Clob	최대 4GB	텍스트	1 ~ 104,857,600자
날짜	B.C. 4712년 1월 1일 ~ A.D. 4712년 12월 31일 전체 자릿수 19, 소수 자릿수 0	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일 (나노초까지의 전체 자릿수)
부동 소수점 수	전체 자릿수 1 ~ 15, 소수 자릿수 0	배정밀도	전체 자릿수 15, 소수 자릿수 0
긴 정수	최대 2GB	텍스트	1 ~ 104,857,600자 매핑에 긴 정수 데이터를 포함할 경우 통합 서비스는 이를 변환 문자열 데이터 유형으로 변환하고 104,857,600자로 자릅니다.
긴 정수(원시)	최대 2GB	이진	1~104,857,600바이트
Nchar	1 ~ 2,000바이트	문자열	1 ~ 104,857,600자
Nclob	최대 4GB	텍스트	1 ~ 104,857,600자
숫자	전체 자릿수 1 ~ 38	배정밀도	전체 자릿수 15
숫자(P,S)	전체 자릿수 1 ~ 38, 소수 자릿수 0 ~ 38	10진수	최대 38자리의 전체 자릿수를 지원하는 변환에서 전체 자릿수는 1 ~ 38자리이고 소수 자릿수는 0 ~ 38입니다. 전체 자릿수를 최대 자릿수보다 많게 지정하는 경우 데이터 통합 서비스가 10진수 값을 많은 전체 자릿수 모드에서 두 배로 변환합니다.

Oracle	범위	변환	범위
Nvarchar2	1 ~ 4,000바이트	문자열	1 ~ 104,857,600자
원시	1 ~ 2,000바이트	이진	1~104,857,600바이트
타임스탬프	B.C. 4712년 1월 1일 ~ A.D. 9999년 12월 31일 전체 자릿수 19 ~ 29, 소수 자릿수 0 ~ 9 (나노초까지의 전체 자릿수)	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일 (나노초까지의 전체 자릿수)
시간대가 포함된 타임스탬프	B.C. 4712년 1월 1일 ~ A.D. 9999년 12월 31일 -12:00 ~ +14:00 전체 자릿수 26 ~ 36, 소수 자릿수 0 ~ 9 (나노초까지의 전체 자릿수)	timestampWithTZ	A.D. 1947년 8월 1일 ~ A.D. 2040년 12월 31일 -12:00 ~ +14:00 전체 자릿수 36, 소수 자릿수 9 (나노초까지의 전체 자릿수) 시간대가 포함된 타임스탬프 데이터 유형은 다음과 같은 시간대 영역을 지원하지 않습니다. - AFRICA_CAIRO - AFRICA_MONROVIA - EGYPT - AMERICA_MONTR EAL
현지 시간대가 포함된 타임스탬프	B.C. 4712년 1월 1일 ~ A.D. 9999년 12월 31일 전체 자릿수 19 ~ 29, 소수 자릿수 0 ~ 9 (나노초까지의 전체 자릿수)	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일 (나노초까지의 전체 자릿수)
Varchar	1 ~ 4,000바이트	문자열	1 ~ 104,857,600자
Varchar2	1 ~ 4,000바이트	문자열	1 ~ 104,857,600자
XMLType	최대 4GB	텍스트	1 ~ 104,857,600자

숫자(P,S) 데이터 유형

Developer tool은 소수 자릿수가 음수인 Oracle 숫자(P,S) 값을 지원합니다.

그러나 소수 자릿수가 전체 자릿수 38보다 크거나 전체 자릿수가 음수인 숫자(P,S) 값은 지원하지 않습니다.

소수 자릿수가 음수인 Oracle 숫자가 포함된 테이블을 가져오면 Developer tool이 10진수 데이터 유형으로 표시합니다. 하지만 데이터 통합 서비스는 이 값을 배정밀도로 변환합니다.

Char, Varchar, Clob 데이터 유형

데이터 통합 서비스는 유니코드 데이터 이동 모드를 사용할 경우 사용자가 **Oracle** 데이터베이스의 열에 대해 설정한 길이 의미에 따라 **Char**, **Varchar**, 및 **Clob** 열의 전체 자릿수를 읽습니다.

바이트 의미를 사용하여 열 길이를 결정하는 경우 데이터 통합 서비스는 전체 자릿수를 바이트 수로 읽습니다. **char** 의미를 사용하는 경우 데이터 통합 서비스는 전체 자릿수를 문자 수로 읽습니다.

지원되지 않는 Oracle 데이터 유형

Developer tool은 특정한 Oracle 데이터 유형을 지원하지 않습니다.

Developer tool은 다음과 같은 Oracle 데이터 유형을 지원하지 않습니다.

- Bfile
- 간격 일을 초로 전환
- 간격 년을 월로 전환
- Mslabel
- 원시 Mslabel
- Rowid

SAP HANA 및 변환 데이터 유형

SAP HANA 데이터 유형은 데이터 통합 서비스가 플랫폼 간에 데이터를 이동하는 데 사용하는 변환 데이터 유형에 매핑됩니다.

다음 표에서는 SAP HANA 데이터 유형과 변환 데이터 유형을 비교합니다.

SAP HANA 데이터 유형	범위	변환 데이터 유형	범위
영숫자	전체 자릿수 1 ~ 127	Nstring	1 ~ 104,857,600자
Bigint	-9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807	Bigint	-9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807 전체 자릿수 19, 소수 자릿수 0
이진	이진 데이터 바이트를 저장하는 데 사용합니다.	이진	1~104,857,600바이트
Blob	최대 2GB	이진	1~104,857,600바이트
Clob	최대 2GB	텍스트	1 ~ 104,857,600자
날짜	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일 전체 자릿수 10, 소수 자릿수 0	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일 (나노초까지의 전체 자릿수)

SAP HANA 데이터 유형	범위	변환 데이터 유형	범위
10진수(전체 자릿수, 소수 자릿수) 또는 10진수(p, s)	전체 자릿수 1 ~ 34	10진수	최대 38자리의 전체 자릿수를 지원하는 변환에서 전체 자릿수는 1 ~ 38자리이고 소수 자릿수는 0 ~ 38입니다. 최대 28자리의 전체 자릿수를 지원하는 변환에서 전체 자릿수는 1 ~ 28자리이고 소수 자릿수는 0 ~ 28입니다. 전체 자릿수를 최대 자릿수보다 많이 지정하는 경우 데이터 통합 서비스가 10진수 값을 많은 전체 자릿수 모드에서 두 배로 변환합니다.
배정밀도	단정밀도 64비트 부동 소수점 숫자를 지정합니다.	배정밀도	전체 자릿수 15
부동 소수점 수	전체 자릿수 1 ~ 53	배정밀도	전체 자릿수 15
정수	-2,147,483,648 ~ 2,147,483,647	정수	-2,147,483,648 ~ 2,147,483,647 전체 자릿수 10, 소수 자릿수 0
NClob	최대 2GB	Ntext	1 ~ 104,857,600자
Nvarchar	전체 자릿수 1 ~ 5000	Nstring	1 ~ 104,857,600자
실수	단정밀도 32비트 부동 소수점 숫자를 지정합니다.	실수	전체 자릿수 7, 소수 자릿수 0
Seconddate	0001-01-01 00:00:01 ~ 9999-12-31 24:00:00	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일 (나노초까지의 전체 자릿수)
Shorttext	텍스트 검색 및 문자열 검색 기능을 지원하는 가변 길이 문자열을 지정합니다.	Nstring	1 ~ 104,857,600자
Smalldecimal	전체 자릿수 1 ~ 16	10진수	1 ~ 28 전체 자릿수, 배율 0 ~ 28
Smallint	-32,768 ~ 32,767	작은 정수	정밀도 5, 배율 0
텍스트	텍스트 검색 기능을 지원하는 가변 길이 문자열을 지정합니다.	텍스트	1 ~ 104,857,600자
시간	24시간 기간	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일 (나노초까지의 전체 자릿수)
타임스탬프	0001-01-01 00:00:00.0000000 ~ 9999-12-31 23:59:59.9999999	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일 (나노초까지의 전체 자릿수)
Tinyint	0 ~ 255	작은 정수	정밀도 5, 배율 0

SAP HANA 데이터 유형	범위	변환 데이터 유형	범위
Varchar	전체 자릿수 1 ~ 5000	문자열	1 ~ 104,857,600자
Varbinary	1 ~ 5000바이트	이진	1~104,857,600바이트

XML 및 변환 데이터 유형

XML 데이터 유형은 데이터 통합 서비스가 플랫폼에서 데이터를 이동하기 위해 사용하는 변환 데이터 유형에 매핑됩니다.

데이터 통합 서비스는 W3C 권장 사항(2001년 5월 2일)에 지정된 모든 XML 데이터 유형을 지원합니다. 그러나 데이터 통합 서비스에서 전체 XML 값 범위를 지원하지 않을 수 있습니다. XML 데이터 유형에 대한 자세한 정보는 다음 위치에서 XML 데이터 유형에 대한 W3C 사양을 참조하십시오. <http://www.w3.org/TR/xmlschema-2>.

다음 테이블에서는 XML 데이터 유형을 변환 데이터 유형과 비교합니다.

데이터 유형	변환	범위
anyURI	문자열	1 ~ 104,857,600자
base64Binary	2진	1~104,857,600바이트
부울	문자열	1 ~ 104,857,600자
바이트	정수	-2,147,483,648 ~ 2,147,483,647
날짜	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일(전체 자릿수는 나노초)
날짜/시간	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일(전체 자릿수는 나노초)
10진수	10진수	1 ~ 28 전체 자릿수, 배율 0 ~ 28
배정밀도	배정밀도	전체 자릿수 15자리
기간	문자열	1 ~ 104,857,600자
ENTITIES	문자열	1 ~ 104,857,600자
ENTITY	문자열	1 ~ 104,857,600자
부동 소수점 수	배정밀도	전체 자릿수 15자리
gDay	문자열	1 ~ 104,857,600자
gMonth	문자열	1 ~ 104,857,600자
gMonthDay	문자열	1 ~ 104,857,600자
gYear	문자열	1 ~ 104,857,600자

데이터 유형	변환	범위
gYearMonth	문자열	1 ~ 104,857,600자
hexBinary	2진	1~104,857,600바이트
ID	문자열	1 ~ 104,857,600자
IDREF	문자열	1 ~ 104,857,600자
IDREFS	문자열	1 ~ 104,857,600자
int	정수	-2,147,483,648 ~ 2,147,483,647
정수	정수	-2,147,483,648 ~ 2,147,483,647
언어	문자열	1 ~ 104,857,600자
긴 정수	Bigint	-9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807
이름	문자열	1 ~ 104,857,600자
NCName	문자열	1 ~ 104,857,600자
negativeInteger	정수	-2,147,483,648 ~ 2,147,483,647
NMTOKEN	문자열	1 ~ 104,857,600자
NMTOKENS	문자열	1 ~ 104,857,600자
nonNegativeInteger	정수	-2,147,483,648 ~ 2,147,483,647
nonPositiveInteger	정수	-2,147,483,648 ~ 2,147,483,647
normalizedString	문자열	1 ~ 104,857,600자
NOTATION	문자열	1 ~ 104,857,600자
positiveInteger	정수	-2,147,483,648 ~ 2,147,483,647
QName	문자열	1 ~ 104,857,600자
짧은 정수	정수	-2,147,483,648 ~ 2,147,483,647
문자열	문자열	1 ~ 104,857,600자
시간	날짜/시간	A.D. 0001년 1월 1일 ~ A.D. 9999년 12월 31일(전체 자릿수는 나노초)
토큰	문자열	1 ~ 104,857,600자
unsignedByte	정수	-2,147,483,648 ~ 2,147,483,647
unsignedInt	정수	-2,147,483,648 ~ 2,147,483,647

데이터 유형	변환	범위
unsignedLong	Bigint	-9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807
unsignedShort	정수	-2,147,483,648 ~ 2,147,483,647

데이터 변환

한 데이터 유형에서 다른 데이터 유형으로 데이터를 변환할 수 있습니다.

한 데이터 유형에서 다른 데이터 유형으로 데이터를 변환하려면 다음 방법 중 하나를 사용하십시오.

- 포트 간에 데이터를 전달합니다(포트 간 데이터 변환).
- 변환 함수를 사용하여 데이터를 변환합니다.
- 변환 산술 연산자를 사용하여 데이터를 변환합니다.

포트 간 데이터 변환

데이터 통합 서비스는 포트의 데이터 유형에 따라 데이터를 변환합니다. 데이터 통합 서비스는 데이터가 포트를 통과할 때마다 해당 포트에 할당된 데이터 유형을 식별하고 필요한 경우 지원되는 데이터 유형에 따라 데이터를 변환합니다.

데이터 유형이 동일한 포트 간에 데이터 전달

데이터 유형이 동일한 포트 간에 데이터를 전달하도록 매핑을 작성할 수 있습니다. 데이터 유형이 동일한 포트 간에 데이터를 전달할 때 데이터 통합 서비스는 데이터 또는 데이터 유형을 변경하지 않고 데이터를 전달합니다.

동일한 숫자 데이터 유형의 포트 간에 데이터를 전달하고 데이터가 변환 간에 전송되는 경우 데이터 통합 서비스가 항상 데이터가 전달되는 포트의 소수 자릿수 및 전체 자릿수로 데이터를 변환하는 것은 아닙니다.

전체 자릿수, 소수 자릿수, 전체 자릿수 모드가 서로 다른 포트 간에 다음과 같은 방법으로 10진수 데이터를 전송할 수 있습니다.

작은 전체 자릿수 모드에서 10진수 데이터 전달

작은 전체 자릿수 모드에서 10진수 포트에 데이터를 전달하는 경우 데이터 통합 서비스는 모든 10진수 포트를 전체 자릿수가 15, 소수 자릿수가 0인 고정밀도로 변환합니다. 예를 들어 작은 전체 자릿수 모드에서 10진수(14, 3)에서 10진수(9, 5)로 값을 전달하는 경우 데이터 통합 서비스는 작은 전체 자릿수 모드에서 값을 내부적으로 저장하고 데이터를 자르지 않습니다.

높은 정밀도 모드에서 소수 자릿수를 줄여 10진수 데이터 전달

높은 정밀도 모드에서 10진수 포트에 데이터를 전달할 때 10진수 포트 간에 소수 자릿수를 줄이면 소수 자릿수 잘라내기가 발생합니다. 예를 들어 10진수(18, 5)에서 10진수(18, 2)로 값을 전달한다고 가정합니다. 18.01234를 전달하는 경우 데이터 통합 서비스가 데이터의 소수 자릿수를 잘라 출력 데이터는 18.01이 됩니다.

높은 정밀도 모드에서 전체 자릿수를 줄여 10진수 데이터 전달

높은 정밀도 모드에서 전체 자릿수를 줄여 10진수 포트에 데이터를 전달할 수 있습니다. 예를 들어 높은 정밀도 모드에서 10진수(19, 5)에서 10진수(17, 2)로 값을 전달하는 경우 출력 필드에 17자리를 초과하는 값이 포함되어 있으면 데이터 통합 서비스가 해당 행을 거부합니다.

데이터 유형이 서로 다른 포트 간에 데이터 전달

데이터 유형이 서로 다른 포트 간에 데이터를 전달할 때 데이터 통합 서비스는 변환 언어의 변환 함수를 사용하여 데이터의 유형을 다른 유형으로 변환합니다.

예를 들어 문자열 포트를 정수 포트에 연결한다고 가정합니다. 데이터 통합 서비스는 매핑을 실행할 때 **TO_INTEGER** 함수를 사용하여 입력 데이터를 문자열에서 정수 데이터 유형으로 변환합니다.

데이터 통합 서비스에서 포트 간 변환을 수행하는 경우 전달하는 데이터는 변환 데이터 유형에 대해 유효해야 합니다. 데이터 통합 서비스에서 변환할 수 없는 값을 사용하면 변환 행 오류가 발생합니다. 예를 들어 "9,000,000,000,000,000.777" 값을 포함하는 문자열 포트를 **Bigint** 포트에 연결합니다. 데이터 통합 서비스는 이 문자열을 **Bigint** 값으로 변환하지 못하고 오류를 반환합니다.

데이터 통합 서비스의 포트 간 변환은 변환 간 그리고 파이프라인의 마지막 변환과 대상 간에 수행됩니다.

다음 테이블에는 데이터 통합 서비스가 수행하는 포트 간 변환이 설명되어 있습니다.

데이터 유형	Bigint	정수	10진수	배정밀도	문자열, 텍스트	날짜/시간	이진	시간대가 포함된 타임스탬프
Bigint	아니요	예	예	예	예	아니요	아니요	아니요
정수	예	아니요	예	예	예	아니요	아니요	아니요
10진수	예	예	아니요	예	예	아니요	아니요	아니요
배정밀도	예	예	예	아니요	예	아니요	아니요	아니요
문자열, 텍스트	예	예	예	예	예	예	아니요	예
날짜/시간	아니요	아니요	아니요	아니요	예	예	아니요	아니요
이진	아니요	아니요	아니요	아니요	아니요	아니요	예	아니요
시간대가 포함된 타임스탬프	아니요	아니요	아니요	아니요	예	아니요	아니요	예

부록 B

키보드 바로 가기

이 부록에 포함된 항목:

- [고급 매핑 옵션에 대한 키보드 바로 가기, 223](#)
- [개체 키보드 바로 가기, 224](#)
- [포트 키보드 바로 가기, 225](#)
- [변환 색상표 키보드 바로 가기, 226](#)
- [작업 영역 키보드 바로 가기, 226](#)

고급 매핑 옵션에 대한 키보드 바로 가기

키보드 바로 가기를 사용하여 고급 옵션을 사용하여 매핑 실행 대화 상자로 이동할 수 있습니다.

키보드 바로 가기로 다음과 같은 태스크를 수행할 수 있습니다.

매핑 구성 선택.

Shift+a를 누릅니다.

사용자 지정 매핑 구성 지정.

Shift+c를 누릅니다.

기본 데이터 통합 서비스 사용.

Shift+u를 누릅니다.

데이터 통합 서비스.

Shift+i를 누릅니다. 추가 옵션을 찾아보려면 **Shift+b**를 누릅니다.

추적 수준 재정의.

Shift+t를 누릅니다.

최적화 수준.

Shift+z를 누릅니다.

매핑에서 기본값 적용.

Shift+d를 누릅니다.

매개 변수 집합 적용.

Shift+s를 누릅니다. 추가 옵션을 찾아보려면 **Shift+o**를 누릅니다.

매개 변수 파일 적용.

Shift+f를 누릅니다. 추가 옵션을 찾아보려면 **Shift+o**를 누릅니다.

개체 키보드 바로 가기

키보드 바로 가기를 사용하여 편집기에 있는 개체를 대상으로 작업할 수 있습니다.

편집기에서 개체를 선택하면 개체의 모양을 변경하고 개체를 이동할 수 있습니다. 개체가 선택되면 녹색 테두리가 개체 주위에 나타납니다. 또한 개체에 포커스가 있음을 나타내는 점선도 선택된 개체 주위에 나타납니다. 특정 개체에 포커스가 있는 상태에서 다른 개체를 선택할 수 있습니다. 편집기에서 여러 개체를 선택하려는 경우 선택된 개체에서 다른 개체로 포커스를 이동합니다.

참고: 편집기에서 개체의 탐색 순서는 위에서 아래와 왼쪽에서 오른쪽입니다.

키보드 바로 가기로 다음과 같은 작업을 수행할 수 있습니다.

개체를 선택합니다.

편집기가 선택되어 있으면 **Tab** 키를 누릅니다. 개체에도 포커스가 있습니다. **Tab** 키를 다시 눌러 다음 개체를 선택합니다.

이전 개체를 선택합니다.

Shift+Tab을 누릅니다.

특정 개체를 찾습니다.

Ctrl+O를 누릅니다.

다음 개체에 포커스를 둡니다.

개체에 포커스가 있을 경우 **Ctrl+Tab**을 눌러 다음 개체로 포커스를 이동합니다. 이전 개체는 선택된 상태로 유지됩니다. **Ctrl+Tab**을 다시 눌러 다음 개체에 포커스를 둡니다.

여러 개체를 개별적으로 선택합니다.

개체에 포커스가 있을 경우 **Ctrl+Tab**을 눌러 선택하려는 다음 개체로 포커스를 이동합니다. 그런 다음 **Ctrl+스페이스바**를 눌러 포커스가 있는 개체를 선택합니다. 이러한 단계를 반복하여 여러 개체를 개별적으로 선택합니다.

선택한 개체를 아이콘으로 표시합니다.

Ctrl+I를 누릅니다.

선택한 개체를 복원합니다.

Ctrl+R를 누릅니다.

선택한 개체의 크기를 조정합니다.

Ctrl+Shift를 누른 상태에서 위로, 아래로, 왼쪽 및 오른쪽 화살표 키를 사용합니다.

선택한 개체를 이동합니다.

화살표 키를 사용하여 원하는 위치로 한 번에 한 픽셀씩 하나 이상의 개체를 이동합니다.

개체에서 편집기로 이동합니다.

한 개 또는 여러 개의 개체가 선택되거나 포커스가 있을 때 **Esc** 키를 누릅니다.

포트 키보드 바로 가기

키보드 바로 가기를 사용하여 포트를 대상으로 작업할 수 있습니다.

포트를 선택하면 포트를 편집하고 다른 포트에 연결할 수 있습니다. 포트가 선택되면 전체 포트가 녹색으로 표시됩니다. 또한 선택된 포트의 한 포트 값 주위에 해당 포트에 포커스가 있음을 알리는 점선도 표시됩니다. 특정 포트에 포커스가 있는 상태에서 다른 포트를 선택할 수 있습니다. 여러 포트를 개별적으로 선택하려는 경우 한 포트에서 다른 포트로 포커스를 이동합니다.

키보드 바로 가기로 다음과 같은 태스크를 수행할 수 있습니다.

포트를 선택합니다.

개체가 선택되고 포커스가 있을 때 **Ctrl+G**를 누릅니다. 위로 및 아래로 화살표 키를 사용하여 다른 포트를 선택합니다.

여러 포트를 선택합니다.

Shift+위로 또는 **Shift+아래로**를 누릅니다.

여러 포트를 개별적으로 선택합니다.

Ctrl 키를 누른 상태에서 위로 및 아래로 화살표 키를 사용하여 선택하려는 포트에 포커스를 둡니다. 그런 다음 **Ctrl+스페이스바**를 눌러 포트를 선택합니다. 이러한 단계를 반복하여 여러 포트를 개별적으로 선택합니다.

다음 포트 값에 포커스를 두고 값을 편집합니다.

Tab 키를 누릅니다.

이전 포트 값에 포커스를 두고 값을 편집합니다.

Shift+Tab을 누릅니다.

포트에서 개체로 이동합니다.

Esc 키를 누릅니다.

두 개체의 포트를 연결합니다.

첫 번째 개체에서 연결하려는 포트를 선택한 후 **Ctrl+L**을 누릅니다. 선택한 포트가 강조 표시됩니다. 두 번째 개체로 이동하여 연결하려는 포트를 선택합니다. **Ctrl+L**을 눌러 포트를 연결합니다.

연결 모드를 끕니다.

Ctrl+Alt+L을 눌러 첫 번째 개체에서 선택한 포트를 선택 취소합니다.

링크를 선택합니다.

선택한 포트에서 **Ctrl+G**를 눌러 이 포트에서 나가는 링크를 선택합니다.

다음 링크를 선택합니다.

Tab 키를 누릅니다.

이전 링크를 선택합니다.

Shift+Tab을 누릅니다.

변환 색상표 키보드 바로 가기

키보드 바로 가기를 사용하여 **변환** 색상표를 탐색할 수 있습니다.

키보드 바로 가기로 다음과 같은 작업을 수행할 수 있습니다.

편집기에서 변환 색상표로 이동합니다.

편집기가 선택되어 있으면 **Ctrl+Shift+P**를 누릅니다.

변환 색상표를 탐색합니다.

Tab 키를 누르거나 **위로** 및 **아래로** 화살표 키를 사용합니다.

변환 색상표에서 편집기로 이동합니다.

Esc 키를 누릅니다.

작업 영역 키보드 바로 가기

키보드 바로 가기를 사용하여 작업 영역에서 편집기 및 보기를 탐색할 수 있습니다.

편집기를 선택하면 편집기 내에서 탐색할 수 있습니다. 편집기가 선택되면 편집기 주위에 녹색 테두리가 나타납니다.

편집기와 보기에 포커스를 둘 수 있습니다. 편집기 또는 보기에 포커스가 있을 경우 주위에 파란색 테두리가 나타납니다. 포커스가 있는 보기의 탭 주위에 점선이 표시됩니다. 보기에 포커스가 있을 경우 보기를 선택하거나 다른 보기에 포커스를 둘 수도 있습니다.

키보드 바로 가기로 다음과 같은 작업을 수행할 수 있습니다.

보기에 포커스를 둡니다.

편집기가 선택되어 있으면 **Shift+Tab**을 누릅니다. 그런 다음 **왼쪽** 및 **오른쪽** 화살표 키를 사용하여 동일한 편집기 내 다른 보기에 포커스를 둡니다. 또는 작업 영역의 다른 보기에 포커스가 있을 때까지 **Shift+Tab**을 누릅니다.

보기에서 첫 번째 영역을 선택합니다.

편집기에서 보기에 포커스가 있을 때 **Ctrl+Tab**을 누릅니다. 다음 위젯을 선택하려면 **Tab** 단추를 세 번 누릅니다.

보기에서 이전 영역을 선택합니다.

Ctrl+Shift+Tab을 누릅니다.

매핑 속성 보기에서 설명을 선택하고 읽습니다.

Ctrl+A를 누릅니다.

편집기를 최소화합니다.

보기에 포커스가 있으면 **Shift+Tab**을 눌러 **최소화** 컨트롤을 선택한 후 스페이스바를 누릅니다.

편집기를 최대화합니다.

보기에 포커스가 있으면 **Shift+Tab**을 눌러 **최소화** 컨트롤을 선택합니다. **오른쪽** 화살표 키를 사용하여 **최대화** 컨트롤을 선택한 후 스페이스바를 누릅니다.

작업 영역의 특정 영역을 선택합니다.

편집기가 선택되어 있으면 **Ctrl+Tab**을 눌러 작업 영역의 다른 영역(예: 보기)을 선택합니다. **Ctrl+Tab**을 다시 눌러 다음 영역을 선택합니다.

작업 영역의 이전 영역을 선택합니다.

Ctrl+Shift+Tab을 누릅니다.

대화 상자의 오류 메시지로 이동합니다.

Ctrl+M을 누릅니다.

부록 C

연결 속성

이 부록에 포함된 항목:

- [연결 속성 개요, 229](#)
- [Adabas 연결 속성, 230](#)
- [Amazon Redshift 연결 속성, 232](#)
- [Amazon S3 연결 속성, 234](#)
- [블록체인 연결 속성, 236](#)
- [Cassandra 연결 속성, 237](#)
- [DataSift 연결 속성, 238](#)
- [Databricks 연결 속성, 239](#)
- [Facebook 연결 속성, 241](#)
- [Greenplum 연결 속성, 242](#)
- [Google Analytics 연결 속성, 243](#)
- [Google BigQuery 연결 속성, 244](#)
- [Google Cloud Spanner 연결 속성, 245](#)
- [Google Cloud Storage 연결 속성, 246](#)
- [Hadoop 연결 속성, 247](#)
- [HBase 연결 속성, 253](#)
- [HDFS 연결 속성, 253](#)
- [MapR-DB용 HBase 연결 속성, 255](#)
- [Hive 연결 속성, 255](#)
- [HTTP 연결 속성, 259](#)
- [IBM DB2 연결 속성, 260](#)
- [IBM DB2 for i5/OS 연결 속성, 263](#)
- [IBM DB2 for z/OS 연결 속성, 266](#)
- [IMS 연결 속성, 269](#)
- [JDBC 연결 속성, 271](#)
- [JDBC V2 연결 속성, 273](#)
- [JD Edwards EnterpriseOne 연결 속성, 275](#)
- [LDAP 연결 속성, 276](#)
- [LinkedIn 연결 속성, 277](#)
- [Microsoft Azure Blob Storage 연결 속성, 278](#)

- [Microsoft Azure Cosmos DB SQL API 연결 속성, 279](#)
- [Microsoft Azure Data Lake Storage Gen1 연결 속성, 280](#)
- [Microsoft Azure Data Lake Storage Gen2 연결 속성, 281](#)
- [Microsoft Azure SQL 데이터 웨어하우스 연결 속성, 282](#)
- [MS SQL Server 연결 속성, 283](#)
- [Netezza 연결 속성, 287](#)
- [OData 연결 속성, 287](#)
- [ODBC 연결 속성, 288](#)
- [Oracle 연결 속성, 290](#)
- [Salesforce 연결 속성, 292](#)
- [Salesforce Marketing Cloud 연결 속성, 293](#)
- [SAP 연결 속성, 294](#)
- [순차적 연결 속성, 297](#)
- [Snowflake 연결 속성, 299](#)
- [Teradata Parallel Transporter 연결 속성, 300](#)
- [Tableau 연결 속성, 302](#)
- [Tableau V3 연결 속성, 303](#)
- [Twitter 연결 속성, 304](#)
- [Twitter 스트리밍 연결 속성, 305](#)
- [VSAM 연결 속성, 306](#)
- [웹 콘텐츠-Kapow Katalyst 연결 속성, 308](#)
- [웹 서비스 연결 속성, 309](#)
- [데이터베이스 연결의 식별자 속성, 311](#)

연결 속성 개요

연결 속성을 통해 Informatica 클라이언트가 데이터 소스에 연결할 수 있습니다.

이 장에는 Informatica 클라이언트를 통해 작성하고 관리할 수 있는 각 연결에 대한 연결 속성이 나와 있습니다.

Adabas 연결 속성

Adabas 연결을 사용하여 Adabas 데이터베이스에 액세스합니다. Adabas 연결은 메인프레임 데이터베이스 유형 연결입니다. Developer tool에서 Adabas 연결을 작성합니다. Administrator 도구 또는 Developer tool에서 Adabas 연결을 관리할 수 있습니다.

다음 테이블에는 Adabas 연결 속성이 설명되어 있습니다.

옵션	설명
위치	Adabas에 연결하는 PowerExchange Listener의 위치에 대한 노드 이름입니다. 노드 이름은 PowerExchange dbmover.cfg 구성 파일에서 NODE 문의 첫 번째 매개 변수에 정의되어 있습니다.
사용자 이름	데이터베이스 사용자 이름입니다. 지원되는 Linux 또는 UNIX 시스템의 데이터베이스의 경우, PowerExchange LDAP 사용자 인증을 활성화했으면 사용자 이름이 엔터프라이즈 사용자 이름입니다. 자세한 내용은 <i>PowerExchange 참조 설명서</i> 를 참조하십시오.
암호	데이터베이스 사용자 이름에 대한 암호 또는 유효한 PowerExchange 암호입니다. PowerExchange 암호는 9자에서 128자 사이일 수 있으며 다음 문자를 포함할 수 있습니다. <ul style="list-style-type: none"> - 대문자 및 소문자 - 숫자 0~9 - 공백 - 다음 특수 문자: ' - ; # \ , . / ! % & * () _ + { } : @ < > ? 참고: 첫 문자는 아포스트로피입니다. 암호는 작은따옴표('), 큰따옴표(") 또는 통화 기호를 포함할 수 없습니다. 암호를 사용하려면 PowerExchange Listener가 DBMOVER 멤버에서 SECURITY=(1,N) 이상의 보안 설정으로 실행되고 있는지 확인하십시오. 자세한 내용은 <i>PowerExchange Reference Manual</i> 에서 "SECURITY Statement"를 참조하십시오. IBM IRRPHREX 종료에서 허용 가능한 문자는 PowerExchange 암호에서 허용 가능한 문자에 영향을 미치지 않습니다. 참고: 유효한 RACF 암호는 최대 100자일 수 있습니다. PowerExchange는 유효성 검사를 위해 암호를 RACF로 전달할 때 100자를 넘는 암호를 자릅니다.
코드 페이지	필수. 데이터 소스에서 읽거나 데이터 소스에 기록할 때 사용하는 코드 페이지의 이름입니다. 대개 이 값은 ISO 코드 페이지 이름입니다(예: ISO-8859-6).
통과 보안 활성화됨	연결에 대한 통과 보안을 활성화합니다. 연결에 대한 통과 보안을 활성화한 경우 도메인은 연결 개체에 정의된 자격 증명 대신 클라이언트 사용자 이름과 암호를 사용하여 해당하는 데이터베이스에 로그인합니다.
암호화 유형	데이터 통합 서비스가 사용하는 암호화 유형입니다. 다음 옵션 중 하나를 선택합니다. <ul style="list-style-type: none"> - 없음 - AES 기본값은 없음입니다. 참고: Informatica에서는 암호화 유형 및 수준 연결 속성을 구성하는 대신 SSL(Secure Sockets Layer) 인증을 사용하는 것을 권장합니다. SSL 인증은 더 엄격한 보안을 제공하며 다양한 Informatica 제품에서 사용됩니다. PowerExchange 네트워크에서 SSL 인증을 구현하는 것에 대한 자세한 내용은 <i>PowerExchange 참조 설명서</i> 를 참조하십시오.

옵션	설명
[암호화] 수준	<p>AES를 암호화 유형으로 선택하는 경우 다음 옵션 중 하나를 선택하여 데이터 통합 서비스가 사용하는 암호화 수준을 나타냅니다.</p> <ul style="list-style-type: none"> - 1. 128비트 암호화 키를 사용합니다. - 2. 192비트 암호화 키를 사용합니다. - 3. 256비트 암호화 키를 사용합니다. <p>AES를 암호화 유형으로 선택하지 않는 경우 이 옵션은 무시됩니다.</p> <p>기본값은 1입니다.</p>
간격 크기	<p>선택 사항입니다. 소스 시스템이 PowerExchange Listener로 전달할 수 있는 데이터의 양입니다. 외부 응용 프로그램, 데이터베이스 또는 데이터 통합 서비스 노드가 병목 지점인 경우 간격 크기를 설정하십시오. 낮은 값을 사용할수록 성능이 빨라집니다.</p> <p>최소값과 기본값은 0입니다. 0 값이 최상의 성능을 제공합니다.</p>
행으로 해석	<p>선택 사항입니다. 간격 크기를 행 수로 표현하려면 이 옵션을 선택합니다. 간격 크기를 킬로바이트로 표현하려면 이 옵션을 선택 취소합니다. 기본적으로 이 옵션은 선택되지 않으며 간격 크기는 킬로바이트 단위입니다.</p>
압축	<p>선택 사항입니다. 소스 데이터 압축을 활성화하려면 이 옵션을 선택합니다. 데이터를 압축하면 Informatica 응용 프로그램이 네트워크를 통해 전송하는 데이터의 양을 줄일 수 있습니다. 기본적으로 이 옵션은 선택되지 않으며 압축이 비활성화됩니다.</p>
오프로드 처리	<p>선택 사항입니다. 일부 대량 데이터 처리를 소스 시스템에서 데이터 통합 서비스 시스템으로 오프로드할지 여부를 제어합니다. 다음 옵션 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - AUTO. 데이터 통합 서비스가 오프로드 처리를 사용할지 여부를 결정합니다. - 예. 오프로드 처리를 사용합니다. - 아니요. 오프로드 처리를 사용하지 않습니다. <p>기본값은 AUTO입니다.</p>
작업자 스레드	<p>선택 사항입니다. 오프로드 처리가 활성화된 경우 데이터 통합 서비스가 대량 데이터를 처리하기 위해 사용하는 스레드 수입니다. 최적의 성능을 위해서는 이 값이 데이터 통합 서비스 시스템에서 사용할 수 있는 프로세서 수를 초과하지 않아야 합니다. 올바른 값은 1~64입니다. 기본값은 다중 스레딩을 비활성화하는 0입니다.</p>
배열 크기	<p>선택 사항입니다. 작업자 스레드를 위한 저장소 배열의 레코드 수입니다. 이 옵션은 작업자 스레드 옵션을 0보다 큰 값으로 설정한 경우 적용됩니다. 유효한 값은 1~5000입니다. 기본값은 25입니다.</p>
쓰기 모드	<p>선택 사항입니다. 데이터 통합 서비스가 PowerExchange Listener로 데이터를 전송하는 모드입니다. 다음 쓰기 모드 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - CONFIRMWRITEON. PowerExchange Listener로 데이터를 전송하고 추가 데이터를 전송하기 전에 응답을 대기합니다. 오류 복구가 중요한 경우 이 옵션을 선택합니다. 하지만 이 옵션을 선택하면 성능이 저하될 수 있습니다. - CONFIRMWRITEOFF. 응답을 기다리지 않고 PowerExchange Listener로 데이터를 전송합니다. 오류 발생 시 대상 테이블을 다시 로드할 수 있는 경우 이 옵션을 사용합니다. - ASYNCHRONOUSWITHFAULTTOLERANCE. 응답을 기다리지 않고 PowerExchange Listener로 데이터를 전송합니다. 이 옵션도 오류 검색을 활성화합니다. 이 옵션은 CONFIRMWRITEOFF의 속도와 CONFIRMWRITEON의 데이터 무결성을 결합합니다. <p>기본값은 CONFIRMWRITEON입니다.</p>

Amazon Redshift 연결 속성

Amazon Redshift 연결을 설정할 때 연결 속성을 구성해야 합니다.

다음 테이블에는 Amazon Redshift 연결 속성이 설명되어 있습니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 있습니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~ ` ! \$ % ^ & * () - + = { [] } \ ; ' " < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결의 설명입니다. 설명은 4,000자를 초과할 수 없습니다.
위치	연결을 작성하려는 도메인입니다.
유형	연결 유형입니다. 데이터베이스에서 Amazon Redshift를 선택합니다.

세부 정보 탭에 Amazon Redshift 연결에 대한 연결 특성이 포함되어 있습니다. 다음 테이블에는 연결 특성이 설명되어 있습니다.

속성	설명
사용자 이름	Amazon Redshift 계정의 사용자 이름입니다.
암호	Amazon Redshift 계정의 암호입니다.
액세스 키 ID	Amazon S3 버킷 액세스 키 ID입니다. 참고: AWS IAM(Identity and Access Management) 인증을 사용하지 않는 경우 필요합니다.
암호 액세스 키	Amazon S3 버킷 암호 액세스 키 ID입니다. 참고: AWS IAM(Identity and Access Management) 인증을 사용하지 않는 경우 필요합니다.
마스터 대칭 키	선택 사항입니다. 클라이언트 측 암호화를 활성화하면 256비트 AES 암호화 키가 Base64 형식으로 제공 됩니다. 타사 도구를 사용하여 키를 생성할 수 있습니다. 값을 지정하는 경우 고급 대상 속성에서 암호화 유형을 클라이언트 측 암호화로 지정해야 합니다.
JDBC URL	Amazon Redshift 연결 URL입니다.

속성	설명
클러스터 지역	<p>선택 사항입니다. 액세스하려는 버킷이 상주하는 AWS 클러스터 지역입니다.</p> <p>JDBC URL 연결 속성에 클러스터 지역 이름이 없는 사용자 지정 JDBC URL을 제공하도록 선택하는 경우 클러스터 지역을 선택합니다.</p> <p>클러스터 지역 및 JDBC URL 연결 속성 모두에 클러스터 지역을 지정하는 경우 데이터 통합 서비스는 JDBC URL 연결 속성에 지정한 클러스터 지역을 무시합니다.</p> <p>JDBC URL 연결 속성에 지정한 클러스터 지역 이름을 사용하려면 이 속성에서 클러스터 지역으로 없음을 선택합니다.</p> <p>다음 클러스터 지역 중 하나를 선택합니다.</p> <p>다음 지역 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - 아시아 태평양(몸바이) - 아시아 태평양(서울) - 아시아 태평양(싱가포르) - 아시아 태평양(시드니) - 아시아 태평양(도쿄) - AWS GovCloud(US) - 캐나다(중부) - 중국(베이징) - 중국(닝샤) - EU(아일랜드) - EU(프랑크푸르트) - EU(런던) - EU(파리) - 남아메리카(상파울루) - 미국 동부(오하이오) - 미국 동부(버지니아 북부) - 미국 서부(캘리포니아 북부) - 미국 서부(오리건) <p>기본값은 없음입니다.</p> <p>PowerExchange for Amazon Redshift에서 사용하는 AWS SDK가 지원하는 클러스터 지역에서만 데이터를 읽거나 쓸 수 있습니다.</p>
고객 마스터 키 ID	<p>선택 사항입니다. 교차 계정 액세스를 위한 사용자 지정 키의 AWS KMS(AWS Key Management Service) 또는 ARN(Amazon 리소스 이름)에서 생성된 고객 마스터 키 ID를 지정합니다. Amazon S3 버킷이 상주하는 지역에 해당하는 고객 마스터 키를 생성해야 합니다. 다음 값 중 하나를 지정할 수 있습니다.</p> <p>고객이 생성한 고객 마스터 키</p> <p>클라이언트 측 또는 서버 측 암호화를 활성화합니다.</p> <p>기본 고객 마스터 키</p> <p>클라이언트 측 또는 서버 측 암호화를 활성화합니다. 계정의 관리자 사용자만 기본 고객 마스터 키 ID를 사용하여 클라이언트 측 암호화를 활성화할 수 있습니다.</p>

Amazon S3 연결 속성

Amazon S3 연결을 설정할 때 연결 속성을 구성해야 합니다.

다음 테이블에는 Amazon S3 연결 속성이 설명되어 있습니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 있습니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~ ` ! \$ % ^ & * () - + = { [] } \ ; " ' < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	선택 사항입니다. 연결의 설명입니다. 설명은 4,000자를 초과할 수 없습니다.
위치	연결을 작성하려는 도메인입니다.
유형	Amazon S3 연결 유형입니다.
액세스 키	Amazon S3 버킷 액세스를 위한 액세스 키입니다. 다음 인증 방법을 기반으로 액세스 키 값을 제공합니다. <ul style="list-style-type: none"> - 기본 인증: 실제 액세스 키 값을 제공합니다. - IAM 인증: 액세스 키 값을 제공하지 마십시오. - 역할 가정을 통한 임시 보안 자격 증명: Amazon S3 버킷에 대한 액세스 권한이 없는 IAM 사용자의 액세스 키를 제공합니다.
암호 키	Amazon S3 버킷 액세스를 위한 암호 액세스 키입니다. <p>암호 키는 액세스 키와 연관되어 있으며 계정을 고유하게 식별합니다. 다음 인증 방법을 기반으로 액세스 키 값을 제공합니다.</p> <ul style="list-style-type: none"> - 기본 인증: 실제 액세스 암호 값을 제공합니다. - IAM 인증: 액세스 암호 값을 제공하지 마십시오. - 역할 가정을 통한 임시 보안 자격 증명: Amazon S3 버킷에 대한 액세스 권한이 없는 IAM 사용자의 액세스 암호를 제공합니다.
IAM 역할 ARN	사용자가 동적으로 생성된 임시 보안 자격 증명을 사용하기 위해 가정한 IAM 역할의 ARN입니다. <p>임시 보안 자격 증명을 사용하여 AWS 리소스에 액세스하려는 경우 이 속성의 값을 입력합니다.</p> <p>IAM 인증과 함께 임시 보안 자격 증명을 사용하려는 경우 액세스 키 및 암호 키 연결 속성을 제공하지 마십시오. IAM 인증 없이 임시 보안 자격 증명을 사용하려는 경우 액세스 키 및 암호 키 연결 속성의 값을 입력해야 합니다.</p> <p>IAM 역할의 ARN을 얻는 방법에 대한 자세한 내용은 AWS 설명서를 참조하십시오.</p>
폴더 경로	Amazon S3 개체의 전체 경로입니다. 경로에는 버킷 이름과 폴더 이름이 포함되어야 합니다. <p>폴더 경로 끝에 슬래시를 사용 하지 마십시오. 예를 들어 <버킷 이름>/<내 폴더 이름>입니다.</p>
마스터 대칭 키	선택 사항입니다. 클라이언트 측 암호화를 활성화하면 256비트 AES 암호화 키가 Base64 형식으로 제공됩니다. 타사 도구를 사용하여 마스터 대칭 키를 생성할 수 있습니다.

속성	설명
지역 이름	<p>액세스하려는 버킷이 상주하는 AWS 지역을 선택합니다.</p> <p>다음 지역 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - 아시아 태평양(뭄바이) - 아시아 태평양(서울) - 아시아 태평양(싱가포르) - 아시아 태평양(시드니) - 아시아 태평양(도쿄) - AWS GovCloud(US) - 캐나다(중부) - 중국(베이징) - 중국(홍콩) - 중국(닝샤) - EU(아일랜드) - EU(프랑크푸르트) - EU(런던) - EU(파리) - 남아메리카(상파울루) - 미국 동부(오하이오) - 미국 동부(버지니아 북부) - 미국 서부(캘리포니아 북부) - 미국 서부(오리건) <p>기본값은 미국 동부(버지니아 북부)입니다.</p>
고객 마스터 키 ID	<p>선택 사항입니다. 교차 계정 액세스를 위한 사용자 지정 키의 AWS KMS(AWS Key Management Service) 또는 ARN(Amazon 리소스 이름)에서 생성된 고객 마스터 키 ID 또는 별칭 이름을 지정합니다. Amazon S3 버킷이 상주하는 지역과 동일한 지역에 대한 고객 마스터 키를 생성해야 합니다.</p> <p>다음 값 중 하나를 지정할 수 있습니다.</p> <p>고객이 생성한 고객 마스터 키</p> <p>클라이언트 측 또는 서버 측 암호화를 활성화합니다.</p> <p>기본 고객 마스터 키</p> <p>클라이언트 측 또는 서버 측 암호화를 활성화합니다. 계정의 관리자 사용자만 기본 고객 마스터 키 ID를 사용하여 클라이언트 측 암호화를 활성화할 수 있습니다.</p>
페더레이션된 SSO IdP	<p>적용할 수 없습니다. 페더레이션된 SSO 옵션을 선택할 수 없습니다.</p>

블록체인 연결 속성

블록체인 연결을 설정하는 경우 연결 속성을 구성해야 합니다.

다음 테이블에는 블록체인 연결에 대한 일반 연결 속성이 설명되어 있습니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 있습니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결의 설명입니다. 연결을 식별하는 데 사용할 수 있는 문자열을 입력합니다. 설명은 4,000자를 초과할 수 없습니다.
Swagger 파일 경로	블록체인과 통신하기 위한 REST API가 포함된 Swagger 파일 경로의 절대 경로입니다. Swagger 파일은 데이터 통합 서비스 시스템에 저장된 JSON 파일이어야 합니다. Swagger 파일의 형식이 다른 경우(예: YAML) 파일을 JSON 형식으로 변환합니다.
인증 유형*	런타임 엔진이 REST 서버에 연결하는 데 사용하는 인증 방법입니다. 없음, 기본, 다이제스트 또는 OAuth를 사용할 수 있습니다.
인증 사용자 ID*	REST 서버에 인증하기 위한 사용자 이름입니다.
인증 암호*	사용자 이름이 REST 서버에 인증하기 위한 암호입니다.
OAuth 소비자 키*	OAuth 인증 유형에 필요합니다. REST 서버와 연결된 클라이언트 키입니다.
OAuth 소비자 암호*	OAuth 인증 유형에 필요합니다. REST 서버에 연결하기 위한 클라이언트 암호입니다.
OAuth 토큰*	OAuth 인증 유형에 필요합니다. REST 서버에 연결하기 위한 액세스 토큰입니다.
OAuth 토큰 암호*	OAuth 인증 유형에 필요합니다. OAuth 토큰과 연결된 암호입니다.
프록시 유형*	프록시의 유형입니다. 프록시 없음, 플랫폼 프록시 또는 사용자 지정을 사용할 수 있습니다.
프록시 세부 정보*	<호스트>:<포트> 형식을 사용하는 프록시 구성입니다.
트러스트 저장소 파일 경로*	SSL 인증서가 포함된 트러스트 저장소 파일의 절대 경로입니다.
트러스트 저장소 암호*	트러스트 저장소 파일의 암호입니다.
키 저장소 파일 경로*	REST 서버와의 양방향 보안 연결을 설정하는 데 필요한 키 및 인증서가 포함된 키 저장소 파일의 절대 경로입니다.
키 저장소 암호*	키 저장소 파일의 암호입니다.

속성	설명
고급 속성	<p>블록체인의 자산에 액세스하기 위한 고급 속성의 목록입니다. 세미콜론으로 구분된 이름-값 쌍을 사용하여 고급 속성을 지정하십시오.</p> <p>다음과 같은 고급 속성을 사용할 수 있습니다.</p> <ul style="list-style-type: none"> - baseUrl. Swagger 파일에 기본 URL이 포함되어 있지 않은 경우 필요합니다. 블록체인의 자산에 액세스하는 데 사용되는 기본 URL입니다. - X-API-KEY. API 키를 사용하여 REST 서버에 인증하는 경우 필요합니다. <p>연결에서 구성하는 고급 속성은 블록체인 데이터 개체의 해당 고급 속성 값을 재정의합니다. 예를 들어 연결과 데이터 개체 모두 기본 URL을 지정하는 경우 연결의 값은 데이터 개체의 값을 재정의합니다.</p>
쿠키	<p>REST API가 구현되는 방식에 따라 필요합니다. REST 서버로 전달되는 쿠키 정보를 지정하기 위한 쿠키 속성의 목록입니다. 세미콜론으로 구분된 이름-값 쌍을 사용하여 속성을 지정하십시오.</p> <p>연결에서 구성하는 쿠키 속성은 블록체인 데이터 개체의 해당 쿠키 속성 값을 재정의합니다.</p>
<p>* 속성이 무시됩니다. 이 기능을 사용하려면 속성을 고급 속성으로 구성하고 Swagger 파일의 속성 이름을 기반으로 이름-값 쌍을 제공합니다.</p> <p>예를 들어 기본 권한 부여를 사용하려면 다음 이름-값 쌍을 구성합니다.</p> <p>Authorization=Basic <credentials></p>	

Cassandra 연결 속성

Cassandra 연결을 설정하는 경우 연결 속성을 구성해야 합니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 테이블에는 Cassandra 연결 속성이 설명되어 있습니다.

속성	설명
이름	<p>연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 있습니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다.</p> <p>~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /</p>
ID	<p>데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다.</p> <p>ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 생성한 후 이 속성을 변경할 수 없습니다.</p> <p>기본값은 연결 이름입니다.</p>
설명	선택 사항입니다. 연결의 설명입니다. 설명은 4,000자를 초과할 수 없습니다.
위치	연결을 작성하려는 도메인입니다.
유형	연결 유형입니다. Cassandra를 선택합니다.
호스트 이름	Cassandra 서버의 호스트 이름 또는 IP 주소입니다.

속성	설명
포트	Cassandra 서버 포트 번호입니다. 기본값은 9042입니다.
사용자 이름	Cassandra 서버에 액세스하는 데 필요한 사용자 이름입니다.
암호	Cassandra 서버 액세스를 위한 사용자 이름에 해당하는 암호입니다.
기본 키스페이스	기본적으로 사용하는 Cassandra 키스페이스의 이름입니다.
SQL 식별자 문자	<p>데이터베이스에서 SQL 또는 CQL 쿼리의 구분자로 분리된 식별자를 묶기 위해 사용하는 문자 유형입니다. 사용 가능한 문자는 데이터베이스 유형에 따라 다릅니다.</p> <p>데이터베이스에서 일반 식별자를 사용할 경우 없음을 선택합니다. 데이터 통합 서비스에서 SQL 또는 CQL 쿼리를 생성할 때 식별자 주위에 구분자로 분리된 문자를 배치하지 않습니다.</p> <p>데이터베이스에서 구분자로 분리된 식별자를 사용할 경우 문자를 선택합니다. 데이터 통합 서비스가 SQL 또는 CQL 쿼리를 생성할 때 구분자로 분리된 식별자를 이 문자로 묶습니다.</p>
추가 연결 속성	<p>하나 이상의 JDBC 연결 매개 변수를 다음 형식으로 입력하십시오.</p> <pre><param1>=<value>;<param2>=<value>;<param3>=<value></pre> <p>PowerExchange for Cassandra JDBC는 다음과 같은 JDBC 연결 매개 변수를 지원합니다.</p> <ul style="list-style-type: none"> - BinaryColumnLength - DecimalColumnScale - EnableCaseSensitive - EnableNullInsert - EnablePaging - RowsPerPage - StringColumnLength - VTableSeparator
SSL 모드	PowerExchange for Cassandra JDBC에는 적용되지 않습니다. 비활성화됨 을 선택합니다.
SSL TrustStore 경로	PowerExchange for Cassandra JDBC에는 적용되지 않습니다.
SSL TrustStore 암호	PowerExchange for Cassandra JDBC에는 적용되지 않습니다.
SSL 키 저장소 경로	PowerExchange for Cassandra JDBC에는 적용되지 않습니다.
SSL 키 저장소 암호	PowerExchange for Cassandra JDBC에는 적용되지 않습니다.

DataSift 연결 속성

DataSift 스트림에서 데이터를 추출하려면 DataSift 연결을 사용합니다. DataSift 연결은 소셜 미디어 연결입니다. Administrator 도구 또는 Developer tool에서 DataSift 연결을 생성하고 관리할 수 있습니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 표에는 DataSift 연결 속성이 설명되어 있습니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결의 설명입니다. 설명은 765자를 초과할 수 없습니다.
위치	연결을 생성하려는 도메인입니다.
유형	연결 유형입니다. DataSift를 선택합니다.
사용자 이름	DataSift 계정의 사용자 이름입니다.
API 키	API 키입니다. DataSift 계정의 대시보드 또는 설정 페이지에 표시되는 개발자 API 키입니다.

Databricks 연결 속성

Databricks 클러스터에서 매핑을 실행하려면 Databricks 연결을 사용합니다.

Databricks 연결은 클러스터 유형 연결입니다. Administrator 도구 또는 Developer tool에서 Databricks 연결을 생성하고 관리할 수 있습니다. infacmd를 사용하여 Databricks 연결을 생성할 수 있습니다. 데이터 통합 서비스와 Databricks 클러스터 간의 통신이 가능하도록 Databricks 연결에서 속성을 구성합니다.

다음 테이블에는 Databricks 연결에 대한 일반 연결 속성이 설명되어 있습니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 있습니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다.~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	선택 사항입니다. 연결의 설명입니다. 설명은 4,000자를 초과할 수 없습니다.
연결 유형	Databricks를 선택합니다.
클러스터 구성	Databricks 환경과 연결된 클러스터 구성의 이름입니다. 클라우드 프로비저닝 구성을 구성하지 않는 경우 필요합니다.

속성	설명
클라우드 프로비저닝 구성	Databricks 클라우드 플랫폼과 연결된 클라우드 프로비저닝 구성의 이름입니다. 클러스터 구성을 구성하지 않는 경우 필요합니다.
준비 디렉터리	Databricks Spark 엔진이 런타임 파일을 준비하는 디렉터리입니다. 존재하지 않는 디렉터리를 지정하면 데이터 통합 서비스가 런타임 시 디렉터리를 생성합니다. 디렉터리 경로를 제공하지 않는 경우 런타임 준비 파일은 <code>/<클러스터 준비 디렉터리>/DATABRICKS</code> 에 기록됩니다.
고급 속성	<p>Databricks 환경에 고유한 고급 속성 목록입니다.</p> <p>데이터 통합 서비스 및 Databricks 연결에서 Databricks 환경에 대한 런타임 속성을 구성할 수 있습니다. 상위 수준에서 구성된 속성을 하위 수준에서 값을 설정하여 재정의할 수 있습니다. 예를 들어 데이터 통합 서비스 사용자 지정 속성에서 속성을 구성하는 Databricks 연결에서 속성을 재정의할 수 있습니다. 데이터 통합 서비스는 다음 우선 순위에 따라 속성 재정의의 처리합니다.</p> <ol style="list-style-type: none"> 1. Databricks 연결 고급 속성 2. 데이터 통합 서비스 사용자 지정 속성 <p>참고: 타사 설명서, Informatica 설명서 또는 Informatica 글로벌 고객 지원 센터에 문의하기 전에 이러한 속성 값을 변경하지 않는 것이 좋습니다. 속성에 대한 지식 없이 값을 변경하면 성능 저하 또는 기타 예상치 못한 결과가 발생할 수 있습니다.</p>

고급 속성

Databricks 구성 섹션의 **고급 속성**에서 다음 속성을 구성합니다.

`infaspark.json.parser.mode`

파서가 손상된 JSON 레코드를 처리하는 방법을 지정합니다. 다음 모드 중 하나로 값을 설정할 수 있습니다.

- **DROPMALFORMED.** 파서가 모든 손상된 레코드를 무시합니다. 기본 모드입니다.
- **PERMISSIVE.** 파서가 손상된 레코드에서 비표준 필드를 **null**로 수락합니다.
- **FAILFAST.** 손상된 레코드가 있고 Spark 응용 프로그램이 중단된 경우 파서가 예외를 생성합니다.

`infaspark.json.parser.multiLine`

파서가 JSON 파일의 다중 행 레코드를 읽을 수 있는지 여부를 지정합니다. 이 값을 **true** 또는 **false**로 설정할 수 있습니다. 기본값은 **false**입니다. Spark 버전 2.2.x 이상을 사용하는 비윈시 배포에만 적용됩니다.

`infaspark.flatfile.writer.nullValue`

Databricks Spark 엔진이 대상에 쓸 때 null 값을 빈 문자열("")로 변환합니다. 예: 12, AB,"",23p09udj.

Databricks Spark 엔진은 빈 문자열을 문자열 열에 쓸 수 있지만 빈 문자열을 비문자열 열에 쓰려고 하면 유형 불일치로 인해 매핑이 실패합니다.

Databricks Spark 엔진에서 빈 문자열을 null 값으로 다시 변환한 후 대상에 쓸 수 있도록 하려면 Databricks Spark 연결에서 속성을 구성합니다.

TRUE로 설정합니다.

`infaspark.pythontx.exec`

Databricks Spark 엔진에서 Python 변환을 실행하는 데 필요합니다. Databricks 클러스터의 작업자 노드에 있는 Python 실행 파일 바이너리의 위치를 설정합니다.

런타임 시 클러스터를 프로비저닝하는 경우 Databricks 클라우드 프로비저닝 구성에서 이 속성을 설정합니다. 또는 Databricks 연결에서 설정합니다.

예를 들어 다음으로 설정합니다.

```
infaspark.pythontx.exec=/databricks/python3/bin/python3
```

```
infaspark.pythontx.executorEnv.PYTHONHOME
```

Databricks Spark 엔진에서 Python 변환을 실행하는 데 필요합니다. Databricks 클러스터의 작업자 노드에 있는 Python 설치 디렉터리의 위치를 설정합니다.

런타임 시 클러스터를 프로비저닝하는 경우 Databricks 클라우드 프로비저닝 구성에서 이 속성을 설정합니다. 또는 Databricks 연결에서 설정합니다.

예를 들어 다음으로 설정합니다.

```
infaspark.pythontx.executorEnv.PYTHONHOME=/databricks/python3
```

Facebook 연결 속성

Facebook 웹 사이트에서 데이터에 액세스하려면 Facebook 연결을 사용합니다. Facebook 연결은 소셜 미디어 연결입니다. Administrator 도구 또는 Developer tool에서 Facebook 연결을 생성하고 관리할 수 있습니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 표에는 Facebook 연결 속성이 설명되어 있습니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결의 설명입니다. 설명은 765자를 초과할 수 없습니다.
위치	연결을 생성하려는 도메인입니다.
유형	연결 유형입니다. Facebook을 선택합니다.
OAuth 세부 정보가 있으니까?	OAuth를 구성할지 여부를 나타냅니다. 다음 값 중 하나를 선택합니다. - 예. 액세스 토큰이 있음을 나타냅니다. - 아니요. OAuth 유틸리티를 시작합니다.
소비자 키	Facebook에서 응용 프로그램을 생성할 때 얻게 되는 App ID입니다. Facebook에서는 이 키를 사용하여 응용 프로그램을 식별합니다.
소비자 암호	Facebook에서 응용 프로그램을 생성할 때 얻게 되는 App 암호입니다. Facebook에서는 이 암호를 사용하여 소비자 키의 소유권을 설정합니다.
액세스 토큰	OAuth 유틸리티가 반환하는 액세스 토큰입니다. Facebook에서는 사용자 자격 증명 대신 이 토큰을 사용하여 보호된 리소스에 액세스합니다.

속성	설명
액세스 암호	Facebook 연결에는 액세스 암호가 필요하지 않습니다.
범위	응용 프로그램에 대한 사용 권한입니다. OAuth를 구성하는 데 필요한 사용 권한을 입력합니다.

Greenplum 연결 속성

Greenplum 데이터베이스에 연결하려면 Greenplum 연결을 사용합니다. Greenplum 연결은 관계형 유형 연결입니다. Administrator 도구 또는 Developer tool에서 Greenplum 연결을 생성하고 관리할 수 있습니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

Greenplum 연결을 생성할 때 메타데이터 및 데이터 액세스 관련 정보를 입력합니다.

다음 표에는 Greenplum 연결 속성이 설명되어 있습니다.

속성	설명
이름	Greenplum 관계형 연결의 이름입니다.
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. 255자 이하여야 하며 도메인에서 고유해야 합니다. 연결을 생성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결에 대한 설명입니다. 설명은 765자를 초과할 수 없습니다.
위치	연결을 생성하려는 도메인입니다.
유형	연결 유형입니다.

메타데이터를 가져오려면 사용자 이름, 암호, 드라이버 이름 및 연결 문자열이 필요합니다. 다음 테이블에는 메타데이터 액세스에 대한 속성이 설명되어 있습니다.

속성	설명
사용자 이름	Greenplum 데이터베이스에 액세스하는 사용 권한을 가진 사용자 이름입니다.
암호	Greenplum 데이터베이스에 연결하기 위한 암호입니다.
드라이버 이름	Greenplum JDBC 드라이버의 이름입니다. 예: com.pivotal.jdbc.GreenplumDriver 드라이버에 대한 자세한 내용은 Greenplum 설명서를 참조하십시오.
연결 문자열	다음 연결 URL을 사용합니다. jdbc:pivotal:greenplum://<호스트 이름>:<포트>;DatabaseName=<database_name> 연결 URL에 대한 자세한 내용은 Greenplum 설명서를 참조하십시오.

PowerExchange for Greenplum은 호스트 이름, 포트 이름 및 데이터베이스 이름을 사용하여 Greenplum gpload 대량 로드 유틸리티에 로드 사양을 제공하는 제어 파일을 생성합니다. SSL 활성화 옵션과 인증서 경로를 사용하여 SSL을 통해 Greenplum 서버에 대한 보안 통신을 설정합니다.

다음 표에는 데이터 액세스를 위한 연결 속성이 설명되어 있습니다.

속성	설명
호스트 이름	Greenplum 서버의 호스트 이름 또는 IP 주소입니다.
포트 번호	Greenplum 서버 포트 번호입니다. 0을 입력하면 gpload 유틸리티가 환경 변수 \$PGPORT에서 읽습니다. 기본값은 5432입니다.
데이터베이스 이름	데이터베이스 이름입니다.
SSL 활성화	SSL을 통해 gpload 유틸리티와 Greenplum 서버 간에 보안 통신을 설정하려면 이 옵션을 선택합니다.
인증서 경로	Greenplum 서버에 대한 SSL 인증서가 저장되어 있는 경로입니다. 인증서 경로에 존재해야 하는 파일에 대한 자세한 정보는 gpload 설명서를 참조하십시오.

Google Analytics 연결 속성

Google Analytics 연결을 설정할 때 연결 속성을 구성해야 합니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 테이블에는 Google Analytics 연결 속성이 설명되어 있습니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 있습니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 생성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	선택 사항입니다. 연결의 설명입니다. 설명은 4,000자를 초과할 수 없습니다.
위치	연결을 작성하려는 도메인입니다.
유형	연결 유형입니다. Google Analytics를 선택합니다.
서비스 계정 ID	서비스 계정을 생성한 후 다운로드하는 JSON 파일의 client_email 값을 지정합니다.

속성	설명
서비스 계정 키	서비스 계정을 생성한 후 다운로드하는 JSON 파일의 <code>private_key</code> 값을 지정합니다.
APIVersion	PowerExchange for Google Analytics가 Google Analytics 보고서를 읽을 때 사용하는 API입니다. Core Reporting API v3을 선택합니다. 참고: PowerExchange for Google Analytics는 Analytics Reporting API v4를 지원하지 않습니다.

Google BigQuery 연결 속성

Google BigQuery 연결을 설정할 때 연결 속성을 구성해야 합니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 테이블에는 Google BigQuery 연결 속성이 설명되어 있습니다.

속성	설명
서비스 계정 ID	Google BigQuery에서 서비스 계정을 생성한 후 다운로드하는 JSON 파일의 <code>client_email</code> 값을 지정합니다.
서비스 계정 키	Google BigQuery에서 서비스 계정을 생성한 후 다운로드하는 JSON 파일의 <code>private_key</code> 값을 지정합니다.
연결 모드	Google BigQuery의 데이터를 읽거나 Google BigQuery에 데이터를 쓸 때 사용할 모드입니다. 다음 연결 모드 중 하나를 선택합니다. <ul style="list-style-type: none"> - 단순. 레코드 데이터 유형 필드의 각 필드를 매핑의 개별 필드로 평면화합니다. - 하이브리드. 레코드 데이터 유형 필드를 포함하여 Google BigQuery 테이블의 모든 최상위 수준 필드를 표시합니다. PowerExchange for Google BigQuery는 최상위 수준 레코드 데이터 유형 필드를 매핑에서 문자열 데이터 유형의 단일 필드로 표시합니다. - 복합. Google BigQuery 테이블의 모든 열을 매핑에서 문자열 데이터 유형의 단일 필드로 표시합니다. 기본값은 단순입니다.
스키마 정의 파일 경로	Google BigQuery 테이블의 샘플 스키마를 사용하여 JSON 파일을 생성해야 하는 클라이언트 시스템의 디렉터리를 지정합니다. JSON 파일 이름은 Google BigQuery 테이블 이름과 동일합니다. 또는 Google BigQuery 테이블의 샘플 스키마를 사용하여 JSON 파일을 생성해야 하는 Google Cloud Storage의 저장소 경로를 지정할 수 있습니다. Google Cloud Storage의 지정된 저장소 경로에서 로컬 시스템으로 JSON 파일을 다운로드할 수 있습니다.
프로젝트 ID	Google BigQuery에서 서비스 계정을 생성한 후 다운로드하는 JSON 파일의 <code>project_id</code> 값을 지정합니다. 동일한 서비스 계정으로 여러 프로젝트를 생성한 경우 연결하려는 데이터 집합이 포함된 프로젝트의 ID를 입력합니다.
저장소 경로	이 속성은 다량의 데이터를 읽거나 쓰는 경우 적용됩니다. 데이터를 임시로 저장할 로컬 준비 파일을 생성하는 Google Cloud Storage의 경로입니다. 버킷 이름을 입력하거나 버킷 이름과 폴더 이름을 입력할 수 있습니다. 예를 들어 <code>gs://<bucket_name></code> 또는 <code>gs://<bucket_name>/<folder_name></code> 을 입력합니다.

속성	설명
데이터 집합 ID	PowerExchange for Google BigQuery에는 적용할 수 없습니다.
사용자 지정 쿼리에 레거시 SQL 사용	PowerExchange for Google BigQuery에는 적용할 수 없습니다.
사용자 지정 쿼리의 데이터 집합 이름	PowerExchange for Google BigQuery에는 적용할 수 없습니다.
지역 ID	<p>Google BigQuery 데이터 집합이 상주하는 지역 이름입니다.</p> <p>예를 들어 라스베이거스 지역에 상주하는 Google BigQuery 데이터 집합에 연결하려는 경우 us-west4를 지역 ID로 지정합니다.</p> <p>참고: 저장소 경로 연결 속성에서 버킷 이름 또는 동일한 지역에 상주하는 버킷 이름 또는 폴더 이름을 Google BigQuery의 데이터 집합으로 지정해야 합니다.</p> <p>Google BigQuery에서 지원되는 지역에 대한 자세한 내용은 다음 Google BigQuery 설명서를 참조하십시오. https://cloud.google.com/bigquery/docs/locations</p>

연결 모드

다음 연결 모드 중 하나를 사용하도록 Google BigQuery 연결을 구성할 수 있습니다.

단순 모드

단순 모드를 사용하는 경우 PowerExchange for Google BigQuery는 레코드 데이터 유형 필드 내의 각 필드를 Google BigQuery 데이터 개체의 개별 필드로 평면화합니다.

하이브리드 모드

하이브리드 모드를 사용하는 경우 PowerExchange for Google BigQuery는 레코드 데이터 유형 필드를 포함하여 Google BigQuery 테이블의 모든 최상위 수준 필드를 표시합니다. PowerExchange for Google BigQuery는 최상위 수준 레코드 데이터 유형 필드를 Google BigQuery 데이터 개체에서 문자열 데이터 유형의 단일 필드로 표시합니다.

복합 모드

복합 모드를 사용하는 경우 PowerExchange for Google BigQuery는 Google BigQuery 테이블의 모든 열을 Google BigQuery 데이터 개체에서 문자열 데이터 유형의 단일 필드로 표시합니다.

Google Cloud Spanner 연결 속성

Google Cloud Spanner 연결을 설정할 때 연결 속성을 구성해야 합니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 테이블에는 Google Cloud Spanner 연결 속성이 설명되어 있습니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 있습니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다.~`!\$%^&*()-+=[] \;\:;'"<, > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 생성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	선택 사항입니다. 연결의 설명입니다. 설명은 4,000자를 초과할 수 없습니다.
위치	연결을 작성하려는 도메인입니다.
유형	연결 유형입니다. Google Cloud Spanner를 선택합니다.
프로젝트 ID	서비스 계정을 생성한 후 다운로드하는 JSON 파일의 project_id 값을 지정합니다. 동일한 서비스 계정으로 여러 프로젝트를 생성한 경우 연결하려는 버킷이 포함된 프로젝트의 ID를 입력합니다.
서비스 계정 ID	서비스 계정을 생성한 후 다운로드하는 JSON 파일의 client_email 값을 지정합니다.
서비스 계정 키	서비스 계정을 생성한 후 다운로드하는 JSON 파일의 private_key 값을 지정합니다.
인스턴스 ID	Google Cloud Spanner에 생성한 인스턴스의 이름입니다.

Google Cloud Storage 연결 속성

Google Cloud Storage 연결을 설정할 때 연결 속성을 구성해야 합니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 테이블에는 Google Cloud Storage 연결 속성이 설명되어 있습니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 있습니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다.~`!\$%^&*()-+=[] \;\:;'"<, > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 생성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	선택 사항입니다. 연결의 설명입니다. 설명은 4,000자를 초과할 수 없습니다.

속성	설명
위치	연결을 작성하려는 도메인입니다.
유형	연결 유형입니다. Google Cloud Storage를 선택합니다.
프로젝트 ID	서비스 계정을 생성한 후 다운로드하는 JSON 파일의 project_id 값을 지정합니다. 동일한 서비스 계정으로 여러 프로젝트를 생성한 경우 연결하려는 버킷이 포함된 프로젝트의 ID를 입력합니다.
서비스 계정 ID	서비스 계정을 생성한 후 다운로드하는 JSON 파일의 client_email 값을 지정합니다.
서비스 계정 키	서비스 계정을 생성한 후 다운로드하는 JSON 파일의 private_key 값을 지정합니다.

Hadoop 연결 속성

Hadoop 연결을 사용하여 Hadoop 클러스터에서 실행되는 맵핑을 구성합니다. Hadoop 연결은 클러스터 유형 연결입니다. Administrator 도구 또는 Developer tool에서 Hadoop 연결을 생성하고 관리할 수 있습니다. infacmd를 사용하여 Hadoop 연결을 생성할 수 있습니다. Hadoop 연결 속성은 별도의 언급이 없는 한 대/소문자를 구분합니다.

Hadoop 클러스터 속성

데이터 통합 서비스와 Hadoop 클러스터 간의 통신이 가능하도록 Hadoop 연결에서 속성을 구성합니다.

다음 테이블에는 Hadoop 연결 관련 일반 연결 속성이 설명되어 있습니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 연결을 생성한 후 이 속성을 변경할 수 있습니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~ ` ! \$ % ^ & * () - + = { [] } \ ; " ' < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 생성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결의 설명입니다. 연결을 식별하는 데 사용할 수 있는 문자열을 입력합니다. 설명은 4,000자를 초과할 수 없습니다.
클러스터 구성	Hadoop 환경에 연결된 클러스터 구성의 이름입니다. 클라우드 프로비저닝 구성을 구성하지 않은 경우 필요합니다.
클라우드 프로비저닝 구성	Amazon AWS 또는 Microsoft Azure와 같은 클라우드 플랫폼과 연결된 클라우드 프로비저닝 구성의 이름입니다. 클러스터 구성을 구성하지 않은 경우 필요합니다.

속성	설명
클러스터 환경 변수*	<p>Hadoop 클러스터가 사용하는 환경 변수입니다.</p> <p>Cloudera CDH 6.x 클러스터 또는 Cloudera CDP 클러스터를 사용하는 경우 로컬 설정을 클러스터 환경 변수로 구성합니다. Cloudera Manager에서, 환경 변수를 다음 YARN 속성에도 추가해야 합니다.</p> <p>yarn.nodemanager.env-whitelist</p> <p>예를 들어, ORACLE_HOME 변수는 Oracle 데이터베이스 클라이언트 소프트웨어가 설치된 디렉터리를 나타냅니다.</p> <p>데이터 통합 서비스, Hadoop 연결 및 매핑에서 Hadoop 환경에 대한 런타임 속성을 구성할 수 있습니다. 상위 수준에서 구성된 속성을 하위 수준에서 값을 설정하여 재정의할 수 있습니다. 예를 들어 데이터 통합 서비스 사용자 지정 속성에서 속성을 구성하는 경우 Hadoop 연결 또는 매핑에서 속성을 재정의할 수 있습니다. 데이터 통합 서비스는 다음 우선 순위에 따라 속성 재정의 처리합니다.</p> <ol style="list-style-type: none"> 1. infacmd ms runMapping과 -cp 옵션을 사용하여 설정된 매핑 사용자 지정 속성 2. Hadoop 환경에 대한 매핑 런타임 속성 3. 런타임 엔진에 대한 Hadoop 연결 고급 속성 4. Hadoop 연결 고급 일반 속성, 환경 변수 및 클래스 경로 5. 데이터 통합 서비스 사용자 지정 속성
클러스터 라이브러리 경로*	<p>클러스터의 공유 라이브러리에 대한 경로입니다.</p> <p>\$DEFAULT_CLUSTER_LIBRARY_PATH 변수에는 기본 디렉터리 목록이 포함됩니다.</p>
클러스터 클래스 경로*	<p>Hadoop jar 파일과 필요한 라이브러리에 액세스할 수 있는 클래스 경로입니다.</p> <p>\$DEFAULT_CLUSTER_CLASSPATH 변수에는 기본 jar 파일 및 라이브러리에 대한 경로 목록이 포함됩니다.</p> <p>데이터 통합 서비스, Hadoop 연결 및 매핑에서 Hadoop 환경에 대한 런타임 속성을 구성할 수 있습니다. 상위 수준에서 구성된 속성을 하위 수준에서 값을 설정하여 재정의할 수 있습니다. 예를 들어 데이터 통합 서비스 사용자 지정 속성에서 속성을 구성하는 경우 Hadoop 연결 또는 매핑에서 속성을 재정의할 수 있습니다. 데이터 통합 서비스는 다음 우선 순위에 따라 속성 재정의 처리합니다.</p> <ol style="list-style-type: none"> 1. infacmd ms runMapping과 -cp 옵션을 사용하여 설정된 매핑 사용자 지정 속성 2. Hadoop 환경에 대한 매핑 런타임 속성 3. 런타임 엔진에 대한 Hadoop 연결 고급 속성 4. Hadoop 연결 고급 일반 속성, 환경 변수 및 클래스 경로 5. 데이터 통합 서비스 사용자 지정 속성
클러스터 실행 파일 경로*	<p>클러스터의 실행 파일 경로입니다.</p> <p>\$DEFAULT_CLUSTER_EXEC_PATH 변수에는 기본 실행 파일에 대한 경로 목록이 포함됩니다.</p>
<p>* 타사 설명서, Informatica 설명서 또는 Informatica 글로벌 고객 지원 센터에 문의하기 전에 이러한 속성 값을 변경하지 않는 것이 좋습니다. 속성에 대한 지식 없이 값을 변경하면 성능 저하 또는 기타 예상치 못한 결과가 발생할 수 있습니다.</p>	

공통 속성

다음 테이블에는 Hadoop 연결에 대해 구성하는 공통 연결 속성이 설명되어 있습니다.

속성	설명
가장 사용자 이름	<p>Hadoop 클러스터가 Kerberos 인증을 사용하는 경우에 필요합니다. Hadoop 가장 사용자. 데이터 통합 서비스가 Hadoop 환경에서 매핑을 실행하기 위해 가장하는 사용자 이름입니다.</p> <p>데이터 통합 서비스는 구성된 사용자를 기반으로 매핑을 실행합니다. 데이터 통합 서비스는 다음 우선 순위에 따라 사용자로 매핑을 실행합니다.</p> <ol style="list-style-type: none"> 1. 운영 체제 프로필 사용자. 프로필 사용자가 구성된 경우 운영 체제 프로필 사용자로 매핑이 실행됩니다. 운영 체제 프로필 사용자가 없으면 Hadoop 가장 사용자로 매핑이 실행됩니다. 2. Hadoop 가장 사용자입니다. 운영 체제 프로필 사용자가 구성되지 않은 경우 Hadoop 가장 사용자로 매핑이 실행됩니다. Hadoop 가장 사용자가 구성되지 않은 경우 데이터 통합 서비스는 데이터 통합 서비스 사용자로 매핑을 실행합니다. 3. Informatica 서비스 사용자. 운영 체제 프로필 사용자 및 Hadoop 가장 사용자가 구성되지 않은 경우 Informatica 데몬을 시작한 사용자를 사용하여 매핑이 실행됩니다.
임시 테이블 압축 코덱	<p>압축 코덱 클래스 이름을 위한 Hadoop 압축 라이브러리입니다.</p> <p>참고: Spark 엔진은 임시 테이블에 대한 압축 설정을 지원하지 않습니다. Spark 엔진에서 매핑을 실행하면 Spark 엔진이 임시 테이블을 압축되지 않은 파일 형식으로 저장합니다.</p>
코덱 클래스 이름	<p>데이터 압축을 할 수 있게 하고 임시 준비 테이블의 성능을 향상시키는 코덱 클래스 이름입니다.</p>
Hive 준비 데이터베이스 이름	<p>Hive 준비 테이블의 네임스페이스입니다. 지정된 데이터베이스 이름이 없는 테이블에는 default라는 이름을 사용합니다.</p> <p>네임스페이스를 구성하지 않으면 데이터 통합 서비스에서 Hive 대상 연결의 Hive 데이터베이스 이름을 사용하여 준비 테이블을 생성합니다.</p> <p>원시 환경에서 Hive에 데이터를 쓰는 매핑을 실행하는 경우 Hive 연결에 Hive 준비 데이터베이스 이름을 구성해야 합니다. 데이터 통합 서비스는 Hadoop 연결에 구성된 값을 무시합니다.</p>
환경 SQL	<p>Hadoop 환경을 설정하는 SQL 명령입니다. 데이터 통합 서비스는 HiveServer2 작업에서 생성된 각 Hive 스크립트를 시작할 때 환경 SQL을 실행합니다.</p> <p>환경 SQL을 사용할 때는 다음과 같은 규칙 및 지침이 적용됩니다.</p> <ul style="list-style-type: none"> - 또한 환경 SQL을 사용하여 PreSQL 명령 또는 사용자 지정 쿼리에서 사용하려는 Hadoop 또는 Hive 매개 변수를 정의할 수도 있습니다. - 환경 SQL 속성에 여러 값을 사용할 경우 값 사이에 공백이 없어야 합니다.

속성	설명
엔진 유형	<p>데이터 통합 서비스는 Spark 엔진에서 HiveServer2 태스크를 실행하여 HiveServer2를 사용하여 일부 작업의 부분을 처리합니다. 관리 도구를 통해 클러스터 구성을 가져오는 경우 연결을 생성하도록 선택할 수 있습니다. 엔진 유형 속성은 기본적으로 배포를 기반으로 채워집니다.</p> <p>수동으로 연결을 생성하는 경우 엔진 유형을 구성해야 합니다. Hadoop 배포에 따라 다음과 같은 엔진 유형을 사용합니다.</p> <ul style="list-style-type: none"> - Amazon EMR - Tez - Azure HDI - Tez - Cloudera CDH - MRv2 - Cloudera CDP - Tez - Hortonworks HDP - Tez - MapR - MRv2
고급 속성	<p>Hadoop 환경에 고유한 고급 속성 목록입니다. 속성은 Blaze 및 Spark 엔진에 공통됩니다. 고급 속성에는 기본 속성 목록이 포함됩니다.</p> <p>데이터 통합 서비스, Hadoop 연결 및 매핑에서 Hadoop 환경에 대한 런타임 속성을 구성할 수 있습니다. 상위 수준에서 구성된 속성을 하위 수준에서 값을 설정하여 재정의할 수 있습니다. 예를 들어 데이터 통합 서비스 사용자 지정 속성에서 속성을 구성하는 경우 Hadoop 연결 또는 매핑에서 속성을 재정의할 수 있습니다. 데이터 통합 서비스는 다음 우선 순위에서 속성 재정의 처리합니다.</p> <ol style="list-style-type: none"> 1. infacmd ms runMapping과 -cp 옵션을 사용하여 설정된 매핑 사용자 지정 속성 2. Hadoop 환경에 대한 매핑 런타임 속성 3. 런타임 엔진에 대한 Hadoop 연결 고급 속성 4. Hadoop 연결 고급 일반 속성, 환경 변수 및 클래스 경로 5. 데이터 통합 서비스 사용자 지정 속성 <p>참고: 타사 설명서, Informatica 설명서 또는 Informatica 글로벌 고객 지원 센터에 문의하기 전에 이러한 속성 값을 변경하지 않는 것이 좋습니다. 속성에 대한 지식 없이 값을 변경하면 성능 저하 또는 기타 예상치 못한 결과가 발생할 수 있습니다.</p>

거부 디렉터리 속성

다음 테이블에는 Hadoop 거부 디렉터리에 대해 구성하는 연결 속성이 설명되어 있습니다.

속성	설명
Hadoop에 거부 파일 쓰기	<p>Blaze 엔진을 사용하여 매핑을 실행하려면 확인란을 선택하여 거부 파일을 이동할 위치를 지정합니다. 확인란을 선택하면 데이터 통합 서비스가 거부 파일을 속성(거부 파일 디렉터리)에 나열된 HDFS 위치로 이동합니다.</p> <p>기본적으로 데이터 통합 서비스는 RejectDir 시스템 매개 변수에 따라 거부 파일을 저장합니다.</p>
거부 파일 디렉터리	매핑을 실행할 때 HDFS에 있는 Hadoop 매핑 파일의 디렉터리입니다.

Blaze 구성

다음 테이블에는 **Blaze** 엔진에 대해 구성하는 연결 속성이 설명되어 있습니다.

속성	설명
Blaze 준비 디렉터리	Blaze 엔진이 임시 파일을 저장하는 데 사용하는 디렉터리의 HDFS 파일 경로입니다. 디렉터리가 있는지 확인하십시오. YARN 사용자, Blaze 엔진 사용자 및 매핑 가장 사용자는 이 디렉터리에 대해 쓰기 권한을 가지고 있어야 합니다. 기본값은 /blaze/workdir입니다. 이 속성을 지우면 준비 파일을 Hadoop 준비 디렉터리인 /tmp/blaze_<사용자 이름>에 기록합니다.
Blaze 사용자 이름	Blaze 서비스 및 Blaze 서비스 로그의 소유자입니다. Hadoop 클러스터가 Kerberos 인증을 사용하는 경우, 기본 사용자는 데이터 통합 서비스의 SPN 사용자입니다. Hadoop 클러스터가 Kerberos 인증을 사용하지 않고 Blaze 사용자가 구성되지 않은 경우 기본 사용자는 데이터 통합 서비스 사용자입니다.
최소 포트	Blaze 엔진의 포트 번호 범위에 대한 최소값입니다. 기본값은 12300입니다.
최대 포트	Blaze 엔진의 포트 번호 범위에 대한 최대값입니다. 기본값은 12600입니다.
YARN 대기열 이름	클러스터의 사용 가능한 리소스를 지정하는 Blaze 엔진에서 사용하는 YARN 스케줄러 대기열 이름입니다. 참고: 클러스터에서 YARN 선점이 활성화되어 있는 경우 Hadoop 관리자를 통해 Blaze 엔진과 연결된 대기열에는 선점이 비활성화되어 있는지 확인합니다.
Blaze 작업 모니터 주소	Blaze 작업 모니터의 호스트 이름과 포트 번호입니다. 다음 형식을 사용합니다. <호스트 이름>:<포트> 항목 설명 - <호스트 이름>은 Blaze 작업 모니터 서버의 호스트 이름 또는 IP 주소입니다. - <포트>는 Blaze 작업 모니터가 RPC(원격 프로시저 호출)를 수신하는 포트입니다. 예를 들어 다음을 입력합니다. myhostname:9080
Blaze YARN 노드 레이블	Blaze 엔진이 실행되는 Hadoop 클러스터 노드를 결정하는 노드 레이블입니다. 노드 레이블을 지정하지 않으면 Blaze 엔진이 기본 파티션의 노드에서 실행됩니다. Hadoop 클러스터가 노드 레이블에 대한 논리 연산자를 지원하는 경우 노드 레이블 목록을 지정할 수 있다. 노드 레이블을 나열하려면 &&(AND), (OR) 및 !(NOT) 연산자를 사용하십시오. 참고: Cloudera CDH 클러스터에서 노드 레이블을 사용할 수 없습니다.
고급 속성	Blaze 엔진에 고유한 고급 속성 목록입니다. 고급 속성에는 기본 속성 목록이 포함됩니다. 데이터 통합 서비스, Hadoop 연결 및 매핑에서 Hadoop 환경에 대한 런타임 속성을 구성할 수 있습니다. 상위 수준에서 구성된 속성을 하위 수준에서 값을 설정하여 재정의할 수 있습니다. 예를 들어 데이터 통합 서비스 사용자 지정 속성에서 속성을 구성하는 경우 Hadoop 연결 또는 매핑에서 속성을 재정의할 수 있습니다. 데이터 통합 서비스는 다음 우선 순위에 따라 속성 재정의 처리합니다. 1. infacmd ms runMapping과 -cp 옵션을 사용하여 설정된 매핑 사용자 지정 속성 2. Hadoop 환경에 대한 매핑 런타임 속성 3. 런타임 엔진에 대한 Hadoop 연결 고급 속성 4. Hadoop 연결 고급 일반 속성, 환경 변수 및 클래스 경로 5. 데이터 통합 서비스 사용자 지정 속성 참고: 타사 설명서, Informatica 설명서 또는 Informatica 글로벌 고객 지원 센터에 문의하기 전에 이러한 속성 값을 변경하지 않는 것이 좋습니다. 속성에 대한 지식 없이 값을 변경하면 성능 저하 또는 기타 예상치 못한 결과가 발생할 수 있습니다.

Spark 구성

다음 테이블에는 Spark 엔진에 대해 구성하는 연결 속성이 설명되어 있습니다.

속성	설명
Spark 준비 디렉터리	<p>Spark 엔진이 실행 중인 작업에 대한 임시 파일을 저장하는 데 사용하는 디렉터리의 HDFS 파일 경로입니다. YARN 사용자, 데이터 통합 서비스 사용자 및 매핑 가장 사용자는 이 디렉터리에 대해 쓰기 권한을 가지고 있어야 합니다.</p> <p>파일 경로를 지정하지 않는 경우 기본적으로 임시 파일은 Hadoop 준비 디렉터리 /tmp/SPARK_<사용자 이름>에 기록됩니다.</p> <p>Spark 엔진에서 Sqoop 작업을 실행하는 경우 데이터 통합 서비스는 Spark 준비 디렉터리 안에 Sqoop 준비 디렉터리를 생성하여 임시 파일을 저장합니다. <Spark 준비 디렉터리>/sqoop_staging</p>
Spark 이벤트 로그 디렉터리	<p>선택 사항입니다. Spark 엔진이 이벤트를 기록하는 데 사용하는 디렉터리의 HDFS 파일 경로입니다.</p>
YARN 대기열 이름	<p>클러스터의 사용 가능한 리소스를 지정하는 Spark 엔진에서 사용하는 YARN 스케줄러 대기열 이름입니다. 이름은 대/소문자를 구분합니다.</p>
고급 속성	<p>Spark 엔진에 고유한 고급 속성 목록입니다. 고급 속성에는 기본 속성 목록이 포함됩니다.</p> <p>데이터 통합 서비스, Hadoop 연결 및 매핑에서 Hadoop 환경에 대한 런타임 속성을 구성할 수 있습니다. 상위 수준에서 구성된 속성을 하위 수준에서 값을 설정하여 재정의할 수 있습니다. 예를 들어 데이터 통합 서비스 사용자 지정 속성에서 속성을 구성하는 경우 Hadoop 연결 또는 매핑에서 속성을 재정의할 수 있습니다. 데이터 통합 서비스는 다음 우선 순위에 따라 속성 재정의 처리합니다.</p> <ol style="list-style-type: none"> 1. infacmd ms runMapping과 -cp 옵션을 사용하여 설정된 매핑 사용자 지정 속성 2. Hadoop 환경에 대한 매핑 런타임 속성 3. 런타임 엔진에 대한 Hadoop 연결 고급 속성 4. Hadoop 연결 고급 일반 속성, 환경 변수 및 클래스 경로 5. 데이터 통합 서비스 사용자 지정 속성 <p>참고: 타사 설명서, Informatica 설명서 또는 Informatica 글로벌 고객 지원 센터에 문의하기 전에 이러한 속성 값을 변경하지 않는 것이 좋습니다. 속성에 대한 지식 없이 값을 변경하면 성능 저하 또는 기타 예상치 못한 결과가 발생할 수 있습니다.</p>

HBase 연결 속성

HBase 연결을 사용하여 HBase에 액세스합니다. HBase 연결은 NoSQL 연결입니다. Administrator 도구 또는 Developer tool에서 HBase 연결을 생성하고 관리할 수 있습니다. HBase 연결 속성은 별도의 언급이 없는 한 대/소문자를 구분합니다.

다음 테이블에는 HBase 연결 속성이 설명되어 있습니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 연결을 생성한 후 이 속성을 변경할 수 있습니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 생성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결의 설명입니다. 설명은 4,000자를 초과할 수 없습니다.
위치	연결을 생성하려는 도메인입니다.
유형	연결 유형입니다. HBase를 선택합니다.
데이터베이스 유형	연결할 데이터베이스 유형입니다. HBase 테이블에 대한 연결을 생성하려면 HBase를 선택합니다.

HDFS 연결 속성

HDFS(Hadoop 파일 시스템) 연결을 사용하여 Hadoop 클러스터의 데이터에 액세스합니다. HDFS 연결은 파일 시스템 유형 연결입니다. Administrator 도구, Analyst 도구 또는 Developer tool에서 HDFS 연결을 생성하고 관리할 수 있습니다. HDFS 연결 속성은 별도의 언급이 없는 한 대/소문자를 구분합니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 테이블에는 HDFS 연결 속성이 설명되어 있습니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결의 설명입니다. 설명은 765자를 초과할 수 없습니다.

속성	설명
위치	연결을 생성하려는 도메인입니다. Analyst 도구에서는 사용되지 않습니다.
유형	연결 유형입니다. 기본값은 Hadoop 파일 시스템입니다.
사용자 이름	HDFS에 액세스하는 데 필요한 사용자 이름입니다.
NameNode URI	저장소 시스템 액세스를 위한 URI입니다. fs.defaultFS에 대한 값은 클러스터 구성의 core-site.xml 구성 설정에서 찾을 수 있습니다. 참고: 클러스터 구성을 가져올 때 연결을 생성하면, NameNode URI 속성이 기본적으로 채워지고 클러스터 구성을 새로 고칠 때마다 업데이트됩니다. 이 속성을 수동으로 설정하거나 값을 재정의 하면 새로 고침 작업을 수행해도 속성이 업데이트되지 않습니다.

여러 저장소 유형 액세스

다양한 저장소 유형에 연결하려면 연결 매개 변수의 NameNode URI 속성을 사용합니다. 다음 테이블에는 저장소 유형과 함께 저장소 유형의 NameNode URI 형식이 나열되어 있습니다.

저장소	NameNode URI 형식
HDFS	hdfs://<namenode>:<port> 설명: - <노드 이름>는 NameNode의 호스트 이름 또는 IP 주소입니다. - <포트>는 NameNode가 RPC(원격 프로시저 호출)를 수신하는 포트입니다. NameNode고가용성의 경우 hdfs://<nameservice>입니다.
MapR-FS	maprfs:///
HDInsight의 WASB	wasb://<container_name>@<account_name>.blob.core.windows.net/<path> 설명: - <container_name>은 특정 Azure Storage Blob 컨테이너를 식별합니다. 참고: <container_name>은 선택 사항입니다. - <account_name>은 Azure Storage Blob 개체를 식별합니다. 예: wasb://infabdmoffering1storage.blob.core.windows.net/infabdmoffering1cluster/mr-history
HDInsight의 ADLS	adl://home

Azure HDInsight 클러스터에서 클러스터 구성을 생성하는 경우 클러스터 구성은 ADLS 또는 WASB를 기본 저장소로 사용합니다. ADLS 또는 WASB를 보조 저장소로 사용하여 클러스터 구성을 생성할 수 없습니다. HDFS 연결에서 NameNode URI 속성을 편집하여 로컬 HDFS 위치에 연결할 수 있습니다.

MapR-DB용 HBase 연결 속성

HBase 연결을 사용하여 MapR-DB 테이블에 연결합니다. HBase 연결은 NoSQL 연결입니다. Administrator 도구 또는 Developer tool에서 HBase 연결을 생성하고 관리할 수 있습니다. HBase 연결 속성은 별도의 언급이 없는 한 대/소문자를 구분합니다.

다음 테이블에는 MapR-DB의 HBase 연결 속성이 설명되어 있습니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 연결을 생성한 후 이 속성을 변경할 수 있습니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 생성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결에 대한 설명입니다. 설명은 4,000자를 초과할 수 없습니다.
위치	연결을 생성하려는 도메인입니다.
유형	연결 유형입니다. HBase를 선택합니다.
데이터베이스 유형	연결할 데이터베이스 유형입니다. MapR-DB 테이블에 대한 연결을 생성하려면 MapR-DB를 선택합니다.
클러스터 구성	Hadoop 환경에 연결된 클러스터 구성의 이름입니다.
MapR-DB 데이터베이스 경로	연결할 MapR-DB 테이블이 포함된 데이터베이스 경로입니다. 올바른 MapR 클러스터 경로를 입력합니다. MapR-DB에 대한 HBase 데이터 개체를 생성하는 경우 이 데이터베이스 경로 필드에 지정한 MapR-DB 경로에 있는 테이블만 검색할 수 있습니다. 지정된 경로의 하위 디렉터리에 있는 테이블에는 액세스할 수 없습니다. 예를 들어 경로를 /user/customers/로 지정한 경우 customers 디렉터리의 테이블에 액세스할 수 있습니다. 그러나 customers 디렉터리에 이름이 regions인 하위 디렉터리가 포함되는 경우 다음 디렉터리의 테이블에는 액세스할 수 없습니다. /user/customers/regions

Hive 연결 속성

Hive 연결을 사용하여 Hive 데이터에 액세스할 수 있습니다. Hive 연결은 데이터베이스 유형 연결입니다. Administrator 도구, Analyst 도구 또는 Developer tool에서 Hive 연결을 작성하고 관리할 수 있습니다. 따로 언급되어 있지 않은 한 Hive 연결 속성을 대/소문자를 구분합니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 테이블에서는 Hive 연결 속성에 대해 설명합니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 연결을 생성한 후 이 속성을 변경할 수 있습니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 생성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결의 설명입니다. 설명은 4000자를 초과할 수 없습니다.
위치	연결을 생성하려는 도메인입니다. Analyst 도구에서는 사용되지 않습니다.
유형	연결 유형입니다. Hive를 선택합니다.
LDAP 사용자 이름	<p>데이터 통합 서비스가 Hadoop 클러스터에서 매핑을 실행하기 위해 가장하는 사용자의 LDAP 사용자 이름입니다. 사용자 이름은 원시 환경에 대해 메타데이터 연결 문자열 또는 데이터 액세스 연결 문자열에 지정하는 JDBC 연결 문자열에 따라 다릅니다.</p> <p>Hadoop 클러스터가 Kerberos 인증을 사용하는 경우 JDBC 연결 문자열의 사용자 이름과 이 사용자 이름이 동일해야 합니다. 그렇지 않으면 사용자 이름은 JDBC 드라이버의 동작에 따라 달라집니다. Hive JDBC 드라이버를 사용하여 사용자 이름을 여러 가지 방법으로 지정할 수 있고 사용자 이름은 JDBC URL의 일부가 될 수 있습니다.</p> <p>Hadoop 클러스터가 Kerberos 인증을 사용하지 않는 경우 사용자 이름은 JDBC 드라이버의 동작에 따라 다릅니다.</p> <p>사용자 이름을 지정하지 않으면 Hadoop 클러스터가 다음과 같은 기준에 따라 작업을 인증합니다.</p> <ul style="list-style-type: none"> - Hadoop 클러스터가 Kerberos 인증을 사용하지 않습니다. 데이터 통합 서비스를 실행하는 시스템의 운영 체제 프로필 사용자 이름에 따라 작업을 인증합니다. - Hadoop 클러스터가 Kerberos 인증을 사용합니다. 데이터 통합 서비스의 SPN에 따라 작업을 인증합니다. LDAP 사용자 이름은 무시됩니다.
암호	LDAP 사용자 이름에 대한 암호입니다.

속성	설명
환경 SQL	<p>Hadoop 환경을 설정하는 SQL 명령입니다. 원시 환경 유형에서 데이터 통합 서비스는 Hive 메타스토어에 대한 연결을 생성할 때마다 환경 SQL을 실행합니다. Hive 연결을 사용하여 Hadoop 클러스터에서 프로필을 실행하는 경우 데이터 통합 서비스는 각 Hive 세션이 시작될 때 환경 SQL을 실행합니다.</p> <p>두 연결 모드 모두에서 환경 SQL을 사용할 때 다음과 같은 규칙 및 지침이 적용됩니다.</p> <ul style="list-style-type: none"> - 환경 SQL을 사용하여 Hive 쿼리를 지정합니다. - 환경 SQL을 사용하여 Hive 사용자 정의 함수의 클래스 경로를 설정한 후 환경 SQL 또는 PreSQL을 사용하여 Hive 사용자 정의 함수를 지정합니다. 데이터 개체 속성에서 PreSQL을 사용하여 클래스 경로를 지정할 수는 없습니다. Hive 사용자 정의 함수를 사용하는 경우 .jar 파일을 다음 디렉터리에 복사해야 합니다. <p><Informatica 설치 디렉터리>/services/shared/hadoop/<Hadoop 배포 이름>/extras/hive-auxjars</p> <ul style="list-style-type: none"> - 또한 환경 SQL을 사용하여 PreSQL 명령 또는 사용자 지정 쿼리에서 사용하려는 Hadoop 또는 Hive 매개 변수를 정의할 수도 있습니다. - 환경 SQL 속성에 여러 값을 사용할 경우 값 사이에 공백이 없어야 합니다.
SQL 식별자 문자	<p>특수 문자 및 예약된 SQL 키워드(예: WHERE)를 식별하는 데 사용되는 문자의 유형입니다. 데이터 통합 서비스에서 특수 문자와 예약된 SQL 키워드 주변에 선택된 문자를 배치합니다. 데이터 통합 서비스는 대/소문자가 혼합된 식별자 지원 속성에도 이 문자를 사용합니다.</p>

소스 또는 대상으로 Hive에 액세스하기 위한 속성

다음 테이블에서는 소스 또는 대상으로 Hive에 액세스하기 위해 구성하는 연결 속성에 대해 설명합니다.

속성	설명
JDBC 드라이버 클래스 이름	<p>Hive JDBC 드라이버 클래스의 이름입니다. 이 옵션을 비워두면 Developer tool은 배포와 함께 제공되는 기본 Apache Hive JDBC 드라이버를 사용합니다. 기본 Apache Hive JDBC 드라이버가 요구 사항에 맞지 않으면 드라이버 클래스 이름을 지정하여 Apache Hive JDBC 드라이버를 타사 Hive JDBC 드라이버로 재정의할 수 있습니다.</p>
메타데이터 연결 문자열	<p>Hadoop 서버에서 메타데이터에 액세스하는 데 사용되는 JDBC 연결 URI입니다. PowerExchange for Hive를 사용하여 HiveServer 서비스 또는 HiveServer2 서비스와 통신할 수 있습니다. HiveServer에 연결하려면 연결 문자열을 다음과 같은 형식으로 지정합니다.</p> <p>jdbc:hive2://<hostname>:<port>/<db></p> <p>여기서</p> <ul style="list-style-type: none"> - <호스트 이름>은(는) HiveServer2가 실행되는 시스템의 이름 또는 IP 주소입니다. - <포트>은(는) HiveServer2가 수신하는 포트 번호입니다. - <db>은(는) 연결하려는 데이터베이스 이름입니다. 데이터베이스 이름을 제공하지 않을 경우 데이터 통합 서비스는 기본 데이터베이스 세부 정보를 사용합니다. <p>HiveServer 2에 연결하려면 Apache Hive가 해당 특정 Hadoop 배포에 대해 구현하는 연결 문자열 형식을 사용하십시오. Apache Hive 연결 문자열 형식에 대한 자세한 내용은 Apache Hive 설명서를 참조하십시오.</p> <p>사용자 가장의 경우 hive.server2.proxy.user=<xyz>를 JDBC 연결 URI에 추가해야 합니다. 사용자 가장을 구성하지 않으면 현재 사용자의 자격 증명을 사용하여 HiveServer2에 연결됩니다.</p> <p>Hadoop 클러스터에서 SSL 또는 TLS 인증을 사용하는 경우 JDBC 연결 URI에 ssl=true를 추가해야 합니다. 예를 들면 다음과 같습니다. jdbc:hive2://<hostname>:<port>/<db>;ssl=true</p> <p>SSL 또는 TLS 인증에 자체 서명된 인증서를 사용하는 경우 클라이언트 시스템 및 데이터 통합 서비스 시스템에서 인증서 파일을 사용할 수 있는지 확인하십시오. 자세한 내용은 <i>Data Engineering Integration 가이드</i>를 참조하십시오.</p>

속성	설명
Hive JDBC 서버 바이패스	<p>JDBC 드라이버 모드입니다. 내장된 JDBC 드라이버 모드를 사용하도록 확인란을 선택합니다.</p> <p>JDBC 포함된 모드를 사용하려면 다음 작업을 수행합니다.</p> <ul style="list-style-type: none"> - Hive 클라이언트 및 Informatica 서비스가 동일한 시스템이 설치되어 있는지 확인합니다. - Hadoop 클러스터에서 매핑을 실행하도록 Hive 연결 속성을 구성합니다. <p>포함되지 않은 모드를 선택하는 경우 데이터 액세스 연결 문자열을 구성해야 합니다.</p> <p>Informatica에서는 JDBC 내장 모드를 사용하는 것을 권장합니다.</p>
세분화된 권한 부여	<p>Hive 소스의 세분화된 권한 부여를 준수하는 옵션을 선택하는 경우 매핑은 다음을 준수합니다.</p> <ul style="list-style-type: none"> - 행 및 열 수준 제한. Sentry 또는 Ranger 보안 모드가 활성화된 Hadoop 클러스터에 적용됩니다. - 데이터 마스킹 규칙. Dynamic Data Masking을 통해 중요한 데이터를 포함하는 열에 설정된 마스킹 규칙에 적용됩니다. <p>옵션을 선택하지 않으면, Blaze 및 Spark 엔진에서 제한 및 마스킹 규칙이 무시되고 제한된 데이터 또는 중요한 데이터가 결과에 포함됩니다.</p>
데이터 액세스 연결 문자열	<p>Hadoop 데이터 저장소에서 데이터에 액세스하는 데 사용되는 연결 문자열입니다. HiveServer에 연결하려면 포함되지 않은 JDBC 모드 연결 문자열을 다음과 같은 형식으로 지정합니다.</p> <pre>jdbc:hive2://<hostname>:<port>/<db></pre> <p>여기서</p> <ul style="list-style-type: none"> - <호스트 이름>은(는) HiveServer2가 실행되는 시스템의 이름 또는 IP 주소입니다. - <포트>은(는) HiveServer2가 수신하는 포트 번호입니다. - <db>은(는) 연결하려는 데이터베이스입니다. 데이터베이스 이름을 제공하지 않을 경우 데이터 통합 서비스는 기본 데이터베이스 세부 정보를 사용합니다. <p>HiveServer 2에 연결하려면 Apache Hive가 해당 특정 Hadoop 배포에 대해 구현하는 연결 문자열 형식을 사용하십시오. Apache Hive 연결 문자열 형식에 대한 자세한 내용은 Apache Hive 설명서를 참조하십시오.</p> <p>사용자 가장의 경우 hive.server2.proxy.user=<xyz>를 JDBC 연결 URI에 추가해야 합니다. 사용자 가장을 구성하지 않으면 현재 사용자의 자격 증명을 사용하여 HiveServer2에 연결됩니다.</p> <p>Hadoop 클러스터에서 SSL 또는 TLS 인증을 사용하는 경우 JDBC 연결 URI에 ssl=true를 추가해야 합니다. 예를 들면 다음과 같습니다. jdbc:hive2://<hostname>:<port>/<db>;ssl=true</p> <p>자체 서명된 인증서를 SSL 또는 TLS 인증에 사용하는 경우 클라이언트 시스템과 데이터 통합 서비스 시스템에서 인증서 파일을 사용할 수 있는지 확인하십시오. 자세한 내용은 <i>Data Engineering Integration 가이드</i>를 참조하십시오.</p>
HDFS의 Hive 준비 디렉터리	<p>Hive 준비 테이블에 대한 HDFS 디렉터리입니다. Hadoop 가장 사용자 및 매핑 가장 사용자에게 실행 권한을 부여해야 합니다.</p> <p>이 옵션은 원시 환경의 Hive 대상에 데이터를 쓰는 경우 적용되고 필요합니다.</p>
Hive 준비 데이터베이스 이름	<p>Hive 준비 테이블의 네임스페이스입니다.</p> <p>Hive 준비 데이터베이스 이름은 데이터 액세스 연결 문자열에서 자동으로 업데이트됩니다. 기본 이름을 재정의하려는 경우에는 Hive 연결에서 Hive 준비 데이터베이스 이름을 구성해야 합니다.</p> <p>이 옵션은 원시 환경에서 Hive 대상에 데이터를 쓰는 매핑을 실행하는 경우 적용됩니다.</p> <p>Blaze 또는 Spark 엔진에서 매핑을 실행하는 경우 Hive 연결에 Hive 준비 데이터베이스 이름을 구성하지 않아도 됩니다. 데이터 통합 서비스는 Hadoop 연결에 구성된 값을 사용합니다.</p>

HTTP 연결 속성

HTTP 연결을 사용하여 REST 웹 서비스 소비자 변환을 웹 서비스에 연결합니다. HTTP 연결은 웹 유형 연결입니다. Developer tool에서 HTTP 연결을 작성합니다. Administrator 도구 또는 Developer tool에서 HTTP 연결을 관리할 수 있습니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 테이블에는 HTTP 연결 속성이 설명되어 있습니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
사용자 이름	웹 서비스에 연결하는 사용자 이름입니다. HTTP 인증 또는 WS-Security를 활성화한 경우 사용자 이름을 입력합니다. 웹 서비스 소비자 변환에 WS-Security 포트가 포함된 경우 변환이 입력 포트를 통해 동적 사용자 이름을 받습니다. 데이터 통합 서비스가 연결에 정의된 사용자 이름을 재정의합니다.
암호	사용자 이름의 암호입니다. HTTP 인증 또는 WS-Security를 활성화한 경우 암호를 입력합니다. 웹 서비스 소비자 변환에 WS-Security 포트가 포함된 경우 변환이 입력 포트를 통해 동적 암호를 받습니다. 데이터 통합 서비스가 연결에 정의된 암호를 재정의합니다.
URL 끝점	액세스하려는 웹 서비스의 URL입니다. 데이터 통합 서비스가 WSDL 파일에 정의된 URL을 재정의합니다. 웹 서비스 소비자 변환에 URL 끝점 포트가 포함된 경우 변환이 입력 포트를 통해 동적으로 URL을 받습니다. 데이터 통합 서비스가 연결에 정의된 URL을 재정의합니다.
제한 시간	데이터 통합 서비스가 연결을 닫기 전에 웹 서비스 공급자의 응답을 대기하는 시간(초)입니다. 1초에서 10,000초 사이의 제한 시간 값을 지정합니다.
HTTP 인증 유형	HTTP를 통한 사용자 인증 유형입니다. 다음 값 중 하나를 선택합니다. <ul style="list-style-type: none"> - 없음. 인증하지 않습니다. - 자동. 데이터 통합 서비스가 웹 서비스 공급자의 인증 유형을 선택합니다. - 기본. 웹 서비스 공급자의 도메인에 대한 사용자 이름과 암호를 제공해야 합니다. 데이터 통합 서비스가 인증을 위해 웹 서비스 공급자에게 사용자 이름과 암호를 전송합니다. - 다이제스트. 웹 서비스 공급자의 도메인에 대한 사용자 이름과 암호를 제공해야 합니다. 데이터 통합 서비스가 사용자 이름과 암호에서 암호화된 메시지 다이제스트를 생성하여 웹 서비스 공급자에게 전송합니다. 공급자가 사용자 이름과 암호에 대한 임시 값을 생성하여 도메인 컨트롤러의 Active Directory에 저장합니다. 공급자는 이 값을 메시지 다이제스트와 비교합니다. 두 값이 일치하면 웹 서비스 공급자가 사용자를 인증합니다. - NTLM. 도메인 이름, 서버 이름 또는 기본 사용자 이름과 암호를 제공해야 합니다. 웹 서비스 공급자는 사용자가 연결된 도메인을 기반으로 사용자를 인증합니다. 웹 서비스 공급자는 Windows 도메인 컨트롤러에서 사용자 이름 및 암호를 가져와 사용자가 제공한 사용자 이름 및 암호와 비교합니다. 두 값이 일치하면 웹 서비스 공급자가 사용자를 인증합니다. NTLM 인증에서는 도메인 컨트롤러의 Active Directory에 암호화된 암호를 저장하지 않습니다.
트러스트 인증서 파일	웹 서비스의 SSL 인증서를 인증할 때 데이터 통합 서비스가 사용하는 트러스트된 인증서의 번들이 포함된 파일입니다. 파일 이름 및 전체 디렉터리 경로를 입력합니다. 기본값은 <Informatica 설치 디렉터리>/services/shared/bin/ca-bundle.crt입니다.

속성	설명
클라이언트 인증서 파일 이름	클라이언트를 인증할 때 웹 서비스가 사용하는 클라이언트 인증서입니다. 웹 서비스에서 데이터 통합 서비스를 인증해야 하는 경우 클라이언트 인증서 파일을 지정합니다.
클라이언트 인증서 암호	클라이언트 인증서에 대한 암호입니다. 웹 서비스에서 데이터 통합 서비스를 인증해야 하는 경우 클라이언트 인증서 암호를 지정합니다.
클라이언트 인증서 유형	클라이언트 인증서 파일의 형식입니다. 다음 값 중 하나를 선택합니다. - PEM. 확장명이 .pem인 파일입니다. - DER. 확장명이 .cer 또는 .der인 파일입니다. 웹 서비스에서 데이터 통합 서비스를 인증해야 하는 경우 클라이언트 인증서 유형을 지정합니다.
개인 키 파일 이름	클라이언트 인증서에 대한 개인 키 파일입니다. 웹 서비스에서 데이터 통합 서비스를 인증해야 하는 경우 개인 키 파일을 지정합니다.
개인 키 암호	클라이언트 인증서의 개인 키에 대한 암호입니다. 웹 서비스에서 데이터 통합 서비스를 인증해야 하는 경우 개인 키 암호를 지정합니다.
개인 키 유형	개인 키의 유형입니다. PEM이 지원되는 유형입니다.

IBM DB2 연결 속성

IBM DB2 연결을 사용하여 IBM DB2에 액세스합니다. IBM DB2 연결은 관계형 데이터베이스 연결입니다. Administrator 도구, Developer tool 또는 Analyst 도구에서 IBM DB2 연결을 작성하고 관리할 수 있습니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 테이블에는 DB2 연결 속성이 설명되어 있습니다.

속성	설명
데이터베이스 유형	데이터베이스 유형입니다.
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 생성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결의 설명입니다. 설명은 765자를 초과할 수 없습니다.
사용자 이름	데이터베이스 사용자 이름입니다.
암호	데이터베이스 사용자 이름에 대한 암호입니다.
통과 보안 활성화됨	연결에 대한 통과 보안을 활성화합니다. 연결에 대한 통과 보안을 활성화한 경우 도메인은 연결 개체에 정의된 자격 증명 대신 클라이언트 사용자 이름과 암호를 사용하여 해당하는 데이터베이스에 로그인합니다.

속성	설명
데이터 액세스를 위한 연결 문자열	<p>데이터베이스에서 메타데이터에 액세스하는 데 사용되는 DB2 연결 URL입니다.</p> <p>dbname</p> <p>여기서 dbname은 DB2 클라이언트에 구성된 별칭입니다.</p>
메타데이터 액세스 속성: 연결 문자열	<p>다음 메타데이터 연결 문자열 URL을 사용합니다.</p> <p>jdbc:informatica:db2://<호스트 이름>:<포트>;DatabaseName=<데이터베이스 이름></p> <p>테이블을 가져올 경우 기본적으로 모든 테이블이 기본 스키마 이름 아래에 표시됩니다. 기본 스키마 대신 특정 스키마 아래에서 테이블을 보려면 테이블을 가져올 스키마의 이름을 지정할 수 있습니다. ischename 매개 변수를 URL에 포함하여 스키마 이름을 지정합니다. 예를 들어 특정 스키마에서 테이블을 가져오려면 다음 구문을 사용합니다.</p> <p>jdbc:informatica:db2://<호스트 이름>:<포트>;DatabaseName=<데이터베이스 이름>;ischename=<schema_name></p> <p>여러 스키마에서 테이블을 검색하여 가져오려면 ischename 매개 변수에 스키마 이름 여러 개를 지정할 수 있습니다. 스키마 이름은 대/소문자를 구분합니다. 여러 스키마 이름을 지정할 때 특수 문자를 사용할 수 없습니다. 여러 스키마 이름을 구분하려면 파이프() 문자를 사용합니다. 예를 들어 스키마 세 개에서 테이블을 검색하여 가져오려면 다음 구문을 사용합니다.</p> <p>jdbc:informatica:db2://<호스트 이름>:<포트>;DatabaseName=<데이터베이스 이름>;ischename=<schema_name1> <schema_name2> <schema_name3></p> <p>스키마 이름 여러 개를 지정할 경우 기본 스키마만 표시 옵션을 지워야 지정된 스키마 이름 아래에서 테이블을 볼 수 있습니다.</p>

속성	설명
AdvancedJDBCSecurityOptions	<p>보안 데이터베이스에 메타데이터 액세스하기 위한 데이터베이스 매개 변수입니다. Informatica는 AdvancedJDBCSecurityOptions 필드 값을 중요한 데이터로 처리하고 매개 변수 문자열을 암호화하여 저장합니다.</p> <p>보안 데이터베이스에 연결하려면 다음 매개 변수를 포함합니다.</p> <ul style="list-style-type: none"> - EncryptionMethod. 필수 사항입니다. 네트워크를 통해 전송되는 경우 데이터가 암호화되었는지를 나타냅니다. 이 매개 변수는 SSL로 설정되어야 합니다. - ValidateServerCertificate. 선택 사항입니다. 데이터베이스 서버에서 보낸 인증서에 대해 Informatica에서 유효성을 검사하는지를 나타냅니다. <p>이 매개 변수를 True로 설정하면 데이터베이스 서버에서 보낸 인증서에 대해 Informatica에서 유효성을 검사합니다. HostNameInCertificate 매개 변수를 지정하면 Informatica에서 인증서의 호스트 이름에 대한 유효성도 검사합니다.</p> <p>이 매개 변수를 false로 설정하면 데이터베이스 서버에서 보낸 인증서에 대해 Informatica에서 유효성을 검사하지 않습니다. Informatica에서 사용자가 지정한 트러스트 저장소 정보를 모두 무시합니다.</p> <ul style="list-style-type: none"> - HostNameInCertificate. 선택 사항입니다. 보안 데이터베이스를 호스팅하는 시스템의 호스트 이름입니다. 호스트 이름을 지정하면 Informatica에서 SSL 인증서의 호스트 이름에 대해 연결 문자열에 포함된 호스트 이름의 유효성을 검사합니다. - cryptoProtocolVersion. 선택 사항입니다. IBM DB2 인스턴스에 대해 TLS를 활성화하는 경우 cryptoProtocolVersion 매개 변수를 다음과 같이 설정합니다. cryptoProtocolVersion=TLSv<버전 번호>. 예: cryptoProtocolVersion=TLSv1.2 <p>참고: 버전 번호는 서버에 대해 구성된 TLS 버전과 동일해야 합니다.</p> <ul style="list-style-type: none"> - TrustStore. 필수 사항입니다. 트러스트 저장소 파일의 경로 및 파일 이름입니다. <p>참고: SSL 또는 TLS를 구성하고 파일 이름만 지정하는 경우 트러스트 저장소 파일을 Informatica 설치 디렉터리로 복사해야 합니다. 연결을 테스트하고 메타데이터를 가져오려면 트러스트 저장소 파일을 다음 디렉터리로 복사합니다.</p> <p><Informatica 클라이언트 설치 디렉터리>/clients/DeveloperClient</p> <p>매핑을 실행하려면 트러스트 저장소 파일을 다음 디렉터리로 복사합니다.</p> <p><Informatica 서버 설치 디렉터리>/tomcat/bin</p> <ul style="list-style-type: none"> - TrustStorePassword. 필수 사항입니다. 보안 데이터베이스에 대한 트러스트 저장소 파일의 암호입니다. <p>참고: Informatica에서 보안 JDBC 매개 변수를 연결 문자열에 추가합니다. 보안 JDBC 매개 변수를 직접 연결 문자열에 포함한 경우 AdvancedJDBCSecurityOptions 필드에서 매개 변수를 입력하지 마십시오.</p>
데이터 액세스 속성: 연결 문자열	<p>데이터베이스에서 데이터에 액세스하는 데 사용되는 연결 문자열입니다.</p> <p>IBM DB2의 경우에는 <데이터베이스 이름>입니다.</p>
코드 페이지	<p>소스 데이터베이스에서 읽거나 대상 데이터베이스 또는 파일에 쓰는 데 사용되는 코드 페이지입니다.</p>
환경 SQL	<p>데이터베이스에 연결할 때 데이터베이스 환경을 설정하는 SQL 명령입니다. 데이터 통합 서비스에서 데이터베이스에 연결할 때마다 연결 환경 SQL을 실행합니다.</p>
트랜잭션 SQL	<p>데이터베이스에 연결할 때 데이터베이스 환경을 설정하는 SQL 명령입니다. 데이터 통합 서비스에서 각 트랜잭션을 시작할 때 트랜잭션 환경 SQL을 실행합니다.</p>
다시 시도 기간	<p>이 속성은 나중에 사용하기 위해 예약되어 있습니다.</p>
테이블스페이스	<p>데이터베이스의 테이블스페이스 이름입니다.</p>

속성	설명
SQL 식별자 문자	<p>데이터베이스에서 SQL 쿼리의 구분자로 분리된 식별자를 묶기 위해 사용하는 문자 유형입니다. 사용 가능한 문자는 데이터베이스 유형에 따라 다릅니다.</p> <p>데이터베이스에서 일반 식별자를 사용할 경우 (없음)을 선택합니다. 데이터 통합 서비스에서 SQL 쿼리를 생성할 때 식별자 주위에 구분자로 분리된 문자를 배치하지 않습니다.</p> <p>데이터베이스에서 구분자로 분리된 식별자를 사용할 경우 문자를 선택합니다. 데이터 통합 서비스가 SQL 쿼리를 생성할 때 구분자로 분리된 식별자를 이 문자로 묶습니다.</p>
대/소문자가 혼합된 식별자 지원	<p>데이터베이스에서 대/소문자를 구분하는 식별자를 사용할 경우 활성화합니다. 이 속성을 활성화하면 데이터 통합 서비스가 SQL 식별자 문자 속성에 대해 선택한 문자로 모든 식별자를 묶습니다.</p> <p>SQL 식별자 문자 속성이 없음으로 설정된 경우 대/소문자가 혼합된 식별자 지원 속성이 비활성화됩니다.</p>
ODBC 공급자	<p>ODBC. ODBC가 연결되는 데이터베이스의 유형입니다. 푸시다운 최적화를 위해 데이터베이스 유형을 지정하여 데이터 통합 서비스가 원시 데이터베이스 SQL을 생성할 수 있도록 합니다. 옵션은 다음과 같습니다.</p> <ul style="list-style-type: none"> - 기타 - Sybase - Microsoft_SQL_Server <p>기본값은 기타입니다.</p>

IBM DB2 for i5/OS 연결 속성

IBM DB2 for i5/OS 연결을 사용하여 IBM DB2 for i5/OS의 테이블에 액세스합니다. IBM DB2 for i5/OS 연결은 관계형 데이터베이스 연결입니다. Administrator 도구 또는 Developer tool에서 IBM DB2 for i5/OS 연결을 작성하고 관리할 수 있습니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 테이블에는 DB2 for i5/OS 연결 속성이 설명되어 있습니다.

속성	설명
이름	<p>연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다.</p> <p>~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /</p>
ID	<p>데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.</p>
설명	<p>연결의 설명입니다. 설명은 255자를 초과할 수 없습니다.</p>
연결 유형	<p>연결 유형(DB2I)입니다.</p>
사용자 이름	<p>데이터베이스 사용자 이름입니다.</p>

속성	설명
암호	<p>지정된 사용자 이름에 대한 암호 또는 유효한 PowerExchange 암호입니다.</p> <p>PowerExchange 암호는 9자에서 31자 사이일 수 있으며 다음 문자를 포함할 수 있습니다.</p> <ul style="list-style-type: none"> - 대문자 및 소문자 - 숫자 0~9 - 공백 - 다음 특수 문자: ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>참고: 첫 문자는 아포스트로피입니다.</p> <p>암호는 작은따옴표(""), 큰따옴표(") 또는 통화 기호를 포함할 수 없습니다.</p> <p>암호를 사용하려면 PowerExchange Listener가 DBMOVER 멤버에서 SECURITY=(1,N) 이상의 보안 설정으로 실행되고 있는지 확인하십시오. 자세한 내용은 <i>PowerExchange Reference Manual</i>에서 "SECURITY Statement"를 참조하십시오.</p>
통과 보안 활성화됨	연결에 대한 통과 보안을 활성화합니다.
데이터베이스 이름	데이터베이스 인스턴스 이름입니다.
위치	DB2에 연결하는 PowerExchange Listener의 위치에 대한 노드 이름입니다. 노드 이름은 PowerExchange dbmover.cfg 구성 파일에서 NODE 문의 첫 번째 매개 변수에 정의되어 있습니다.
환경 SQL	데이터베이스에 연결할 때 데이터베이스 환경을 설정하는 SQL 명령입니다. 데이터 통합 서비스에서 데이터베이스에 연결할 때마다 연결 환경 SQL을 실행합니다.
데이터베이스 파일 재정의	<p>다음과 같은 형식으로 i5/OS 데이터베이스 파일 재정의의 지정합니다.</p> <p>from_file/to_library/to_file/to_member</p> <p>여기서</p> <ul style="list-style-type: none"> - from_file은 재정의할 파일입니다. - to_library는 사용할 새 라이브러리입니다. - to_file은 새 라이브러리에서 사용할 파일입니다. - to_member는 선택 사항이며 사용할 새 라이브러리 및 파일의 멤버입니다. *아무 것도 지정하지 않으면 FIRST가 사용됩니다. <p>단일 연결에서 최대 8개의 고유한 파일 재정의의 지정할 수 있습니다. 단일 재정의는 단일 소스 또는 대상에 적용됩니다. 둘 이상의 파일 재정의의 지정하는 경우 파일 재정의의 문자열을 큰따옴표("")로 묶고 각 파일 재정의의 사이에 공백을 포함시킵니다.</p> <p>참고: 라이브러리 목록 및 데이터베이스 파일 재정의의 모두 지정하고 한 테이블이 둘 모두에 존재하는 경우 데이터베이스 파일 재정의의 값이 우선 적용됩니다.</p>
라이브러리 목록	<p>PowerExchange가 Select, Insert, Delete 또는 Update 문에 대한 테이블 이름을 한정하기 위해 검색하는 라이브러리의 목록입니다. PowerExchange는 테이블 이름이 한정되지 않는 경우 이 목록을 검색합니다.</p> <p>라이브러리는 쉼표로 구분합니다.</p> <p>참고: 라이브러리 목록 및 데이터베이스 파일 재정의의 모두 지정하고 한 테이블이 둘 모두에 존재하는 경우 데이터베이스 파일 재정의의 값이 우선 적용됩니다.</p>
코드 페이지	소스 데이터베이스에서 읽거나 대상 데이터베이스 또는 파일에 기록하는 데 사용되는 코드 페이지입니다.

속성	설명
사용할 SQL 식별자 문자	<p>데이터베이스에서 SQL 쿼리의 구분자로 분리된 식별자를 묶기 위해 사용하는 문자 유형입니다. 사용 가능한 문자는 데이터베이스 유형에 따라 다릅니다.</p> <p>데이터베이스에서 일반 식별자를 사용할 경우 (없음)을 선택합니다. 데이터 통합 서비스에서 SQL 쿼리를 생성할 때 식별자 주위에 구분자로 분리된 문자를 배치하지 않습니다.</p> <p>데이터베이스에서 구분자로 분리된 식별자를 사용할 경우 문자를 선택합니다. 데이터 통합 서비스가 SQL 쿼리를 생성할 때 구분자로 분리된 식별자를 이 문자로 묶습니다.</p>
대/소문자가 혼합된 식별자 지원	<p>데이터베이스에서 대/소문자를 구분하는 식별자를 사용할 경우 활성화합니다. 이 속성을 활성화하면 데이터 통합 서비스가 SQL 식별자 문자 속성에 대해 선택한 문자로 모든 식별자를 묶습니다.</p> <p>SQL 식별자 문자 속성이 없음으로 설정된 경우 대/소문자가 혼합된 식별자 지원 속성이 비활성화됩니다.</p>
분리 수준	<p>트랜잭션의 범위를 커밋합니다. 다음 옵션 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - 없음 - CS. 커서 안정성(Cursor stability)입니다. - RR. 반복 가능한 읽기(Repeatable Read)입니다. - CHG. 변경(Change)입니다. - ALL <p>기본값은 CS입니다.</p>
암호화 유형	<p>선택 사항입니다. 데이터 통합 서비스가 사용하는 암호화 유형입니다. 다음 옵션 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - 없음 - AES <p>기본값은 없음입니다.</p> <p>참고: Informatica에서는 암호화 유형 및 암호화 수준 연결 속성을 구성하는 대신 SSL(Secure Sockets Layer) 인증을 사용할 것을 권장합니다. SSL 인증은 더 엄격한 보안을 제공하며 다양한 Informatica 제품에서 사용됩니다.</p> <p>PowerExchange 네트워크에서 SSL 인증을 구현하는 것에 대한 자세한 내용은 <i>PowerExchange 참조 설명서</i>를 참조하십시오.</p>
암호화 수준	<p>AES를 암호화 유형으로 선택하는 경우 다음 옵션 중 하나를 선택하여 데이터 통합 서비스가 사용하는 암호화 수준을 나타냅니다.</p> <ul style="list-style-type: none"> - 1. 128비트 암호화 키를 사용합니다. - 2. 192비트 암호화 키를 사용합니다. - 3. 256비트 암호화 키를 사용합니다. <p>AES를 암호화 유형으로 선택하지 않는 경우 이 옵션은 무시됩니다.</p> <p>기본값은 1입니다.</p>
간격 크기	<p>선택 사항입니다. 소스 시스템이 PowerExchange Listener로 전달할 수 있는 데이터의 양입니다. 외부 응용 프로그램, 데이터베이스 또는 데이터 통합 서비스 노드가 병목 지점인 경우 간격 크기를 설정하십시오. 낮은 값을 사용할수록 성능이 빨라집니다.</p> <p>최소값과 기본값은 0입니다. 0 값이 최상의 성능을 제공합니다.</p>
행으로 해석	<p>선택 사항입니다. 간격 크기를 행 수로 표현하려면 이 옵션을 선택합니다. 간격 크기를 킬로바이트로 표현하려면 이 옵션을 선택 취소합니다. 기본적으로 이 옵션은 선택되지 않으며 간격 크기는 킬로바이트 단위입니다.</p>
압축	<p>선택 사항입니다. 소스 데이터 압축을 활성화하려면 이 옵션을 선택합니다. 데이터를 압축하면 Informatica 응용 프로그램이 네트워크를 통해 전송하는 데이터의 양을 줄일 수 있습니다. 기본적으로 이 옵션은 선택되지 않으며 압축이 비활성화됩니다.</p>

속성	설명
배열 크기	선택 사항입니다. 작업자 스레드를 위한 저장소 배열의 레코드 수입니다. 이 옵션은 작업자 스레드 옵션을 0보다 큰 값으로 설정한 경우 적용됩니다. 유효한 값은 25~5000입니다. 기본값은 25입니다.
쓰기 모드	<p>선택 사항입니다. 데이터 통합 서비스가 PowerExchange Listener로 데이터를 전송하는 모드입니다. 다음 쓰기 모드 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - CONFIRMWRITEON: PowerExchange Listener로 데이터를 전송하고 추가 데이터를 전송하기 전에 응답을 대기합니다. 오류 복구가 중요한 경우 이 옵션을 선택합니다. 하지만 이 옵션을 선택하면 성능이 저하될 수 있습니다. - CONFIRMWRITEOFF: 응답을 기다리지 않고 PowerExchange Listener로 데이터를 전송합니다. 오류 발생 시 대상 테이블을 다시 로드할 수 있는 경우 이 옵션을 사용합니다. - ASYNCHRONOUSWITHFAULTTOLERANCE: 응답을 기다리지 않고 PowerExchange Listener로 데이터를 전송합니다. 이 옵션도 오류 검색을 활성화합니다. 이 옵션은 CONFIRMWRITEOFF의 속도와 CONFIRMWRITEON의 데이터 무결성을 결합합니다. <p>기본값은 CONFIRMWRITEON입니다.</p>
거부 파일	거부 파일에 대한 PWXR 기본 접두사를 재정의합니다. PowerExchange는 쓰기 모드가 ASYNCHRONOUSWITHFAULTTOLERANCE인 경우 대상 시스템에서 거부 파일을 작성합니다. 거부 파일 작성을 방지하려면 PWXDISABLE를 입력합니다.

IBM DB2 for z/OS 연결 속성

IBM DB2 for z/OS 연결을 사용하여 IBM DB2 for z/OS의 테이블에 액세스합니다. IBM DB2 for z/OS 연결은 관계형 데이터베이스 연결입니다. Administrator 도구 또는 Developer tool에서 IBM DB2 for z/OS 연결을 생성하고 관리할 수 있습니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 테이블에는 DB2 for z/OS 연결 속성이 설명되어 있습니다.

속성	설명
이름	<p>연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다.</p> <p>~ ` ! \$ % ^ & * () - + = { [}] \ ; " ' < , > . ? /</p>
ID	<p>데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.</p>
설명	연결에 대한 설명입니다. 설명은 255자를 초과할 수 없습니다.
연결 유형	연결 유형(DB2Z)입니다.
사용자 이름	데이터베이스 사용자 이름입니다.

속성	설명
암호	<p>지정된 사용자 이름에 대한 암호 또는 유효한 PowerExchange 암호입니다.</p> <p>PowerExchange 암호는 9자에서 128자 사이일 수 있으며 다음 문자를 포함할 수 있습니다.</p> <ul style="list-style-type: none"> - 대문자 및 소문자 - 숫자 0~9 - 공백 - 다음 특수 문자: ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>참고: 첫 문자는 아포스트로피입니다.</p> <p>암호는 작은따옴표('), 큰따옴표(") 또는 통화 기호를 포함할 수 없습니다.</p> <p>암호를 사용하려면 PowerExchange Listener가 DBMOVER 멤버에서 SECURITY=(1,N) 이상의 보안 설정으로 실행되고 있는지 확인하십시오. 자세한 내용은 <i>PowerExchange Reference Manual</i>에서 "SECURITY Statement"를 참조하십시오.</p> <p>IBM IRRPHREX 종료에서 허용 가능한 문자는 PowerExchange 암호에서 허용 가능한 문자에 영향을 미치지 않습니다.</p> <p>참고: 유효한 RACF 암호는 최대 100자일 수 있습니다. PowerExchange는 유효성 검사를 위해 암호를 RACF로 전달할 때 100자를 넘는 암호를 자릅니다.</p>
통과 보안 활성화됨	연결에 대한 통과 보안을 활성화합니다.
DB2 하위 시스템 ID	DB2 하위 시스템의 이름입니다.
위치	DB2에 연결하는 PowerExchange Listener의 위치에 대한 노드 이름입니다. 노드 이름은 PowerExchange dbmover.cfg 구성 파일에서 NODE 문의 첫 번째 매개 변수에 정의되어 있습니다.
환경 SQL	데이터베이스에 연결할 때 데이터베이스 환경을 설정하는 SQL 명령입니다. 데이터 통합 서비스에서 데이터베이스에 연결할 때마다 연결 환경 SQL을 실행합니다.
상관 ID	DB2 요청에 대한 DB2 상관 ID를 형성하기 위해 PWX 접두사에 연결할 값입니다.
코드 페이지	소스 데이터베이스에서 읽거나 대상 데이터베이스 또는 파일에 기록하는 데 사용되는 코드 페이지입니다.
사용할 SQL 식별자 문자	<p>데이터베이스에서 SQL 쿼리의 구분자로 분리된 식별자를 묶기 위해 사용하는 문자 유형입니다. 사용 가능한 문자는 데이터베이스 유형에 따라 다릅니다.</p> <p>데이터베이스에서 일반 식별자를 사용할 경우 (없음)을 선택합니다. 데이터 통합 서비스에서 SQL 쿼리를 생성할 때 식별자 주위에 구분자로 분리된 문자를 배치하지 않습니다.</p> <p>데이터베이스에서 구분자로 분리된 식별자를 사용할 경우 문자를 선택합니다. 데이터 통합 서비스가 SQL 쿼리를 생성할 때 구분자로 분리된 식별자를 이 문자로 묶습니다.</p>
대/소문자가 혼합된 식별자 지원	<p>데이터베이스에서 대/소문자를 구분하는 식별자를 사용할 경우 활성화합니다. 이 속성을 활성화하면 데이터 통합 서비스가 SQL 식별자 문자 속성에 대해 선택한 문자로 모든 식별자를 묶습니다.</p> <p>SQL 식별자 문자 속성이 없음으로 설정된 경우 대/소문자가 혼합된 식별자 지원 속성이 비활성화됩니다.</p>

속성	설명
암호화 유형	<p>선택 사항입니다. 데이터 통합 서비스가 사용하는 암호화 유형입니다. 다음 옵션 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - 없음 - AES <p>기본값은 없음입니다.</p> <p>참고: Informatica에서는 암호화 유형 및 수준 연결 속성을 구성하는 대신 SSL(Secure Sockets Layer) 인증을 사용하는 것을 권장합니다. SSL 인증은 더 엄격한 보안을 제공하며 다양한 Informatica 제품에서 사용됩니다.</p> <p>PowerExchange 네트워크에서 SSL 인증을 구현하는 것에 대한 자세한 내용은 <i>PowerExchange 참조 설 명서</i>를 참조하십시오.</p>
암호화 수준	<p>AES를 암호화 유형으로 선택하는 경우 다음 옵션 중 하나를 선택하여 데이터 통합 서비스가 사용하는 암호화 수준을 나타냅니다.</p> <ul style="list-style-type: none"> - 1. 128비트 암호화 키를 사용합니다. - 2. 192비트 암호화 키를 사용합니다. - 3. 256비트 암호화 키를 사용합니다. <p>AES를 암호화 유형으로 선택하지 않는 경우 이 옵션은 무시됩니다.</p> <p>기본값은 1입니다.</p>
간격 크기	<p>선택 사항입니다. 소스 시스템이 PowerExchange Listener로 전달할 수 있는 데이터의 양입니다. 외부 응용 프로그램, 데이터베이스 또는 데이터 통합 서비스 노드가 병목 지점인 경우 간격 크기를 설정하십시오. 낮은 값을 사용할수록 성능이 빨라집니다.</p> <p>최소값과 기본값은 0입니다. 0 값이 최상의 성능을 제공합니다.</p>
행으로 해석	<p>선택 사항입니다. 간격 크기를 행 수로 표현하려면 이 옵션을 선택합니다. 간격 크기를 킬로바이트로 표현하려면 이 옵션을 선택 취소합니다. 기본적으로 이 옵션은 선택되지 않으며 간격 크기는 킬로바이트 단위입니다.</p>
압축	<p>선택 사항입니다. 소스 데이터 압축을 활성화하려면 이 옵션을 선택합니다. 데이터를 압축하면 Informatica 응용 프로그램이 네트워크를 통해 전송하는 데이터의 양을 줄일 수 있습니다. 기본적으로 이 옵션은 선택되지 않으며 압축이 비활성화됩니다.</p>
오프로드 처리	<p>선택 사항입니다. 일부 대량 데이터 처리를 소스 시스템에서 데이터 통합 서비스 시스템으로 오프로드할지 여부를 제어합니다. 다음 옵션 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - AUTO. 데이터 통합 서비스가 오프로드 처리를 사용할지 여부를 결정합니다. - 예. 오프로드 처리를 사용합니다. - 아니요. 오프로드 처리를 사용하지 않습니다. <p>기본값은 아니요입니다.</p>
작업자 스레드	<p>선택 사항입니다. 오프로드 처리가 활성화된 경우 데이터 통합 서비스가 대량 데이터를 처리하기 위해 사용하는 스레드 수입입니다. 최적의 성능을 위해서는 이 값이 데이터 통합 서비스 시스템에서 사용할 수 있는 프로세서 수를 초과하지 않아야 합니다. 올바른 값은 1~64입니다. 기본값은 다중 스레딩을 비활성화하는 0입니다.</p>
배열 크기	<p>선택 사항입니다. 작업자 스레드를 위한 저장소 배열의 레코드 수입입니다. 이 옵션은 작업자 스레드 옵션을 0보다 큰 값으로 설정한 경우 적용됩니다. 유효한 값은 1~5000입니다. 기본값은 25입니다.</p>

속성	설명
쓰기 모드	<p>데이터 통합 서비스가 PowerExchange Listener로 데이터를 전송하는 모드입니다. 다음 쓰기 모드 중 하나를 구성합니다.</p> <ul style="list-style-type: none"> - CONFIRMWRITEON. PowerExchange Listener로 데이터를 전송하고 추가 데이터를 전송하기 전에 응답을 대기합니다. 오류 복구가 중요한 경우 선택합니다. 이 옵션을 선택하면 성능이 저하될 수 있습니다. - CONFIRMWRITEOFF. 응답을 기다리지 않고 PowerExchange Listener로 데이터를 전송합니다. 오류가 발생한 경우 대상 테이블을 다시 로드할 수 있다면 이 옵션을 사용합니다. - ASYNCHRONOUSWITHFAULTTOLERANCE. 응답을 기다리지 않고 PowerExchange Listener로 데이터를 전송합니다. 이 옵션은 오류를 검색하는 기능도 제공합니다. 이 옵션은 쓰기 확인 끄기의 속도와 쓰기 확인 켜기의 데이터 무결성을 제공합니다. 기본값은 CONFIRMWRITEON입니다.
거부 파일	<p>거부 파일에 대한 PWXR 기본 접두사를 재정의합니다. PowerExchange는 쓰기 모드가 ASYNCHRONOUSWITHFAULTTOLERANCE인 경우 대상 시스템에서 거부 파일을 생성합니다. 거부 파일 생성을 방지하려면 PWXDISABLE를 입력합니다.</p>

IMS 연결 속성

IMS 연결을 사용하여 IMS 데이터베이스에 액세스합니다. IMS 연결은 비관계형 메인프레임 데이터베이스 유형 연결입니다. 데이터 통합 서비스는 PowerExchange를 통해 IMS에 연결합니다. Developer tool에서 IMS 연결을 작성합니다. Administrator 도구 또는 Developer tool에서 IMS 연결을 관리할 수 있습니다.

다음 테이블에는 IMS 연결 속성이 설명되어 있습니다.

옵션	설명
위치	IMS에 연결하는 PowerExchange Listener의 위치에 대한 노드 이름입니다. 노드 이름은 PowerExchange dbmover.cfg 구성 파일에서 NODE 문의 첫 번째 매개 변수에 정의되어 있습니다.
사용자 이름	데이터베이스 사용자 이름입니다.

옵션	설명
암호	<p>지정된 데이터베이스 사용자 이름에 대한 암호 또는 유효한 PowerExchange 암호입니다. PowerExchange 암호는 9자에서 128자 사이일 수 있으며 다음 문자를 포함할 수 있습니다.</p> <ul style="list-style-type: none"> - 대문자 및 소문자 - 숫자 0~9 - 공백 - 다음 특수 문자: ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>참고: 첫 문자는 아포스트로피입니다.</p> <p>암호는 작은따옴표('), 큰따옴표(") 또는 통화 기호를 포함할 수 없습니다.</p> <p>IBM IRRPHREX 종료에서 허용 가능한 문자는 PowerExchange 암호에서 허용 가능한 문자에 영향을 미치지 않습니다.</p> <p>참고: 유효한 RACF 암호는 최대 100자일 수 있습니다. PowerExchange는 유효성 검사를 위해 암호를 RACF로 전달할 때 100자를 넘는 암호를 자릅니다.</p> <p>IMS 연결에 암호를 사용하려면 다음 요구 사항이 충족되는지 확인하십시오.</p> <ul style="list-style-type: none"> - PowerExchange Listener가 DBMOVER 멤버에서 SECURITY=(1,N) 이상의 보안 설정으로 실행되고 있어야 합니다. 자세한 내용은 <i>PowerExchange Reference Manual</i>에서 "SECURITY Statement"를 참조하십시오. - <i>PowerExchange Navigator 사용자 가이드</i>에 설명된 대로 IMS에 대한 ODBA 액세스를 구성해야 합니다. - 액세스 방법으로 IMS ODBA를 지정하는 IMS 데이터 맵을 사용해야 합니다. DL/1 BATCH 액세스 방법에서는 암호를 지원하지 않는 netport 작업을 사용해야 하므로 DL/1 BATCH 액세스 방법을 지정하는 데이터 맵을 사용하지 마십시오. - IMS에 대한 ODBA 액세스를 사용하려면 IMS 데이터베이스가 IMS 제어 영역에서 온라인 상태여야 합니다.
코드 페이지	<p>필수. 데이터 소스에서 읽거나 데이터 소스에 기록할 때 사용하는 코드 페이지의 이름입니다. 대개 이 값은 ISO 코드 페이지 이름입니다(예: ISO-8859-6).</p>
통과 보안 활성화됨	<p>연결에 대한 통과 보안을 활성화합니다.</p>
암호화 유형	<p>데이터 통합 서비스가 사용하는 암호화 유형입니다. 다음 옵션 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - 없음 - AES <p>기본값은 없음입니다.</p> <p>참고: Informatica에서는 암호화 유형 및 수준 연결 속성을 구성하는 대신 SSL(Secure Sockets Layer) 인증을 사용하는 것을 권장합니다. SSL 인증은 더 엄격한 보안을 제공하며 다양한 Informatica 제품에서 사용됩니다.</p> <p>PowerExchange 네트워크에서 SSL 인증을 구현하는 것에 대한 자세한 내용은 <i>PowerExchange 참조 설명서</i>를 참조하십시오.</p>
[암호화] 수준	<p>AES를 암호화 유형으로 선택하는 경우 다음 옵션 중 하나를 선택하여 데이터 통합 서비스가 사용하는 암호화 수준을 나타냅니다.</p> <ul style="list-style-type: none"> - 1. 128비트 암호화 키를 사용합니다. - 2. 192비트 암호화 키를 사용합니다. - 3. 256비트 암호화 키를 사용합니다. <p>AES를 암호화 유형으로 선택하지 않는 경우 이 옵션은 무시됩니다.</p> <p>기본값은 1입니다.</p>
간격 크기	<p>선택 사항입니다. 소스 시스템이 PowerExchange Listener로 전달할 수 있는 데이터의 양입니다. 외부 응용 프로그램, 데이터베이스 또는 데이터 통합 서비스 노드가 병목 지점인 경우 간격 크기를 설정하십시오. 낮은 값을 사용할수록 성능이 빨라집니다.</p> <p>최소값과 기본값은 0입니다. 0 값이 최상의 성능을 제공합니다.</p>

옵션	설명
행으로 해석	선택 사항입니다. 간격 크기를 행 수로 표현하려면 이 옵션을 선택합니다. 간격 크기를 킬로바이트로 표현하려면 이 옵션을 선택 취소합니다. 기본적으로 이 옵션은 선택되지 않으며 간격 크기는 킬로바이트 단위입니다.
압축	선택 사항입니다. 소스 데이터 압축을 활성화하려면 이 옵션을 선택합니다. 데이터를 압축하면 Informatica 응용 프로그램이 네트워크를 통해 전송하는 데이터의 양을 줄일 수 있습니다. 기본적으로 이 옵션은 선택되지 않으며 압축이 비활성화됩니다.
오프로드 처리	선택 사항입니다. 일부 대량 데이터 처리를 소스 시스템에서 데이터 통합 서비스 시스템으로 오프로드할지 여부를 제어합니다. 다음 옵션 중 하나를 선택합니다. <ul style="list-style-type: none"> - AUTO. 데이터 통합 서비스가 오프로드 처리를 사용할지 여부를 결정합니다. - 예. 오프로드 처리를 사용합니다. - 아니요. 오프로드 처리를 사용하지 않습니다. 기본값은 AUTO입니다.
작업자 스레드	선택 사항입니다. 오프로드 처리가 활성화된 경우 데이터 통합 서비스가 대량 데이터를 처리하기 위해 사용하는 스레드 수입니다. 최적의 성능을 위해서는 이 값이 데이터 통합 서비스 시스템에서 사용할 수 있는 프로세서 수를 초과하지 않아야 합니다. 올바른 값은 1~64입니다. 기본값은 다중 스레딩을 비활성화하는 0입니다.
배열 크기	선택 사항입니다. 작업자 스레드를 위한 저장소 배열의 레코드 수입니다. 이 옵션은 작업자 스레드 옵션을 0보다 큰 값으로 설정한 경우 적용됩니다. 유효한 값은 1~5000입니다. 기본값은 25입니다.
쓰기 모드	선택 사항입니다. 데이터 통합 서비스가 PowerExchange Listener로 데이터를 전송하는 모드입니다. 다음 쓰기 모드 중 하나를 선택합니다. <ul style="list-style-type: none"> - CONFIRMWRITEON. PowerExchange Listener로 데이터를 전송하고 추가 데이터를 전송하기 전에 응답을 대기합니다. 오류 복구가 중요한 경우 이 옵션을 선택합니다. 하지만 이 옵션을 선택하면 성능이 저하될 수 있습니다. - CONFIRMWRITEOFF. 응답을 기다리지 않고 PowerExchange Listener로 데이터를 전송합니다. 오류 발생 시 대상 테이블을 다시 로드할 수 있는 경우 이 옵션을 사용합니다. - ASYNCHRONOUSWITHFAULTTOLERANCE. 응답을 기다리지 않고 PowerExchange Listener로 데이터를 전송합니다. 이 옵션도 오류 검색을 활성화합니다. 이 옵션은 CONFIRMWRITEOFF의 속도와 CONFIRMWRITEON의 데이터 무결성을 결합합니다. 기본값은 CONFIRMWRITEON입니다.

JDBC 연결 속성

JDBC 연결을 사용하여 데이터베이스의 테이블에 액세스할 수 있습니다. Administrator 도구, Developer tool 또는 Analyst 도구에서 JDBC 연결을 생성 및 관리할 수 있습니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 테이블에는 JDBC 연결 속성이 설명되어 있습니다.

속성	설명
데이터베이스 유형	데이터베이스 유형입니다.
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 생성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결의 설명입니다. 설명은 765자를 초과할 수 없습니다.
사용자 이름	데이터베이스 사용자 이름입니다.
암호	데이터베이스 사용자 이름에 대한 암호입니다.
JDBC 드라이버 클래스 이름	JDBC 드라이버 클래스의 이름입니다. 다음 목록에서 해당하는 데이터베이스 유형에 입력할 수 있는 드라이버 클래스 이름을 제공합니다. <ul style="list-style-type: none"> - Oracle용 DataDirect JDBC 드라이버 클래스 이름: <code>com.informatica.jdbc.oracle.OracleDriver</code> - IBM DB2용 DataDirect JDBC 드라이버 클래스 이름: <code>com.informatica.jdbc.db2.DB2Driver</code> - Microsoft SQL Server용 DataDirect JDBC 드라이버 클래스 이름: <code>com.informatica.jdbc.sqlserver.SQLServerDriver</code> - Sybase ASE용 DataDirect JDBC 드라이버 클래스 이름: <code>com.informatica.jdbc.sybase.SybaseDriver</code> - Informix용 DataDirect JDBC 드라이버 클래스 이름: <code>com.informatica.jdbc.informix.InformixDriver</code> - MySQL용 DataDirect JDBC 드라이버 클래스 이름: <code>com.informatica.jdbc.mysql.MySQLDriver</code> - Databricks Delta Lake용 JDBC 드라이버: Databricks에서 다운로드한 드라이버의 이름입니다. 드라이버에 대한 자세한 내용은 <i>Data Engineering Integration 가이드</i>의 "Databricks 통합을 시작하기 전에" 장에서 저장소 액세스 구성에 대한 항목을 참조하십시오. 특정 데이터베이스와 같이 사용할 드라이버 클래스에 대한 자세한 정보는 공급업체 설명서를 참조하십시오.
연결 문자열	데이터베이스에 연결하기 위한 연결 문자열입니다. 다음 연결 문자열을 사용합니다. <code>jdbc:<subprotocol>:<subname></code> 특정 드라이버와 함께 사용하는 연결 문자열에 대한 자세한 내용은 공급업체 설명서를 참조하십시오.
환경 SQL	선택 사항입니다. 데이터베이스에 연결할 때 데이터베이스 환경을 설정하는 SQL 명령을 입력합니다. 데이터 통합 서비스에서 데이터베이스에 연결할 때마다 연결 환경 SQL을 실행합니다. 참고: Sqoop를 활성화하면 Sqoop가 이 속성을 무시합니다.
트랜잭션 SQL	선택 사항입니다. 데이터베이스에 연결할 때 데이터베이스 환경을 설정하는 SQL 명령을 입력합니다. 데이터 통합 서비스에서 각 트랜잭션을 시작할 때 트랜잭션 환경 SQL을 실행합니다. 참고: Sqoop를 활성화하면 Sqoop가 이 속성을 무시합니다.

속성	설명
SQL 식별자 문자	<p>데이터베이스에서 SQL 쿼리의 구분자로 분리된 식별자를 묶기 위해 사용하는 문자 유형입니다. 사용 가능한 문자는 데이터베이스 유형에 따라 다릅니다.</p> <p>데이터베이스에서 일반 식별자를 사용할 경우 (없음)을 선택합니다. 데이터 통합 서비스에서 SQL 쿼리를 생성할 때 식별자 주위에 구분자로 분리된 문자를 배치하지 않습니다.</p> <p>데이터베이스에서 구분자로 분리된 식별자를 사용할 경우 문자를 선택합니다. 데이터 통합 서비스가 SQL 쿼리를 생성할 때 구분자로 분리된 식별자를 이 문자로 묶습니다.</p> <p>참고: Sqoop를 활성화하면 Sqoop가 이 속성을 무시합니다.</p>
대/소문자가 혼합된 식별자 지원	<p>데이터베이스에서 대/소문자를 구분하는 식별자를 사용할 경우 활성화합니다. 이 속성을 활성화하면 데이터 통합 서비스가 SQL 식별자 문자 속성에 대해 선택한 문자로 모든 식별자를 묶습니다.</p> <p>SQL 식별자 문자 속성이 없음으로 설정된 경우 대/소문자가 혼합된 식별자 지원 속성이 비활성화됩니다.</p> <p>참고: Sqoop를 활성화하면 런타임 시 대상을 생성하거나 바꾸도록 DDL 스크립트를 생성하고 실행할 때 Sqoop가 이 속성을 수용합니다. 다른 모든 시나리오에서는 Sqoop가 이 속성을 무시합니다.</p>
Sqoop 커넥터 사용	<p>JDBC 연결을 사용하는 데이터 개체에 대해 Sqoop 연결을 활성화합니다. 데이터 통합 서비스는 Sqoop를 통해 Hadoop 런타임 환경에서 매핑을 실행합니다.</p> <p>JDBC와 호환되는 데이터베이스를 기반으로 하는 관계형 데이터 개체, 사용자 지정 데이터 개체 및 논리적 데이터 개체에 대해 Sqoop 연결을 구성할 수 있습니다.</p> <p>Sqoop 연결을 활성화하려면 Sqoop v1.x를 선택합니다.</p> <p>기본값은 없음입니다.</p>
Sqoop 인수	<p>Sqoop가 데이터베이스에 연결할 때 사용해야 하는 인수를 입력합니다. 여러 인수는 공백으로 구분합니다.</p> <p>Sqoop용 TDCH(Teradata Connector for Hadoop) 특수 커넥터를 사용하여 Blaze 엔진에서 매핑을 실행하려면 Sqoop 인수에 TDCH 연결 팩터리 클래스를 정의해야 합니다. 연결 팩터리 클래스는 사용할 TDCH Sqoop 커넥터에 따라 다릅니다.</p> <ul style="list-style-type: none"> - Cloudera Connector Powered by Teradata를 사용하려면 다음 Sqoop 인수를 구성합니다. <ul style="list-style-type: none"> -Dscoop.connection.factories=com.cloudera.connector.teradata.TeradataManagerFactory - Hortonworks Connector for Teradata(Teradata Connector for Hadoop 기술 제공)를 사용하려면 다음 Sqoop 인수를 구성합니다. <ul style="list-style-type: none"> -Dscoop.connection.factories=org.apache.sqoop.teradata.TeradataManagerFactory <p>Spark 엔진에서 매핑을 실행하려면 Sqoop 인수에 TDCH 연결 팩터리 클래스를 정의할 필요가 없습니다. 데이터 통합 서비스는 기본적으로 Cloudera Connector Powered by Teradata 및 Hortonworks Connector for Teradata(Teradata Connector for Hadoop 기술 제공)를 호출합니다.</p> <p>참고: 특수 Cloudera 또는 Hortonworks 커넥터 대신 일반 JDBC 커넥터로 매핑을 실행하려면 JDBC 연결에서 --driver 및 --connection-manager Sqoop 인수를 정의해야 합니다. 매핑의 읽기 또는 쓰기 변환에서 --driver 및 --connection-manager 인수를 정의하면 Sqoop은 인수를 무시합니다.</p> <p>Sqoop 인수를 입력하지 않는 경우 데이터 통합 서비스가 JDBC 연결 속성을 기반으로 Sqoop 명령을 구성합니다.</p>

JDBC V2 연결 속성

JDBC V2 연결을 설정하는 경우 연결 속성을 구성해야 합니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 테이블에는 JDBC V2 연결 속성이 설명되어 있습니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 있습니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~`!\$%^&*()-+={}[]\ ;";'<,>./
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결의 설명입니다. 설명은 4,000자를 초과할 수 없습니다.
위치	연결을 작성하려는 도메인입니다.
유형	연결 유형입니다. JDBC V2를 선택합니다.

세부 정보 탭에는 JDBC V2 연결에 대한 연결 특성이 포함되어 있습니다. 다음 테이블에는 연결 특성이 설명되어 있습니다.

속성	설명
사용자 이름	데이터베이스 사용자 이름입니다. 유형 4 JDBC 드라이버를 지원하는 데이터베이스에 대한 액세스 권한이 있는 사용자 이름.
암호	데이터베이스 사용자 이름에 대한 암호입니다.
스키마 이름	선택 사항입니다. 데이터베이스에서 연결할 스키마 이름입니다. 스키마 이름을 지정하지 않으면 데이터베이스에서 사용할 수 있는 모든 스키마가 나열됩니다.
JDBC 드라이버 클래스 이름	JDBC 드라이버 클래스의 이름입니다. 다음 목록에서 해당하는 데이터베이스 유형에 입력할 수 있는 드라이버 클래스 이름을 제공합니다. - Azure SQL 데이터베이스에 대한 JDBC 드라이버 클래스 이름: com.microsoft.sqlserver.jdbc.SQLServerDriver - Aurora PostgreSQL에 대한 JDBC 드라이버 클래스 이름: org.postgresql.Driver 특정 데이터베이스에서 사용할 드라이버 클래스에 대한 자세한 내용은 타사 공급업체 설명서를 참조하십시오.
연결 문자열	데이터베이스에 연결하기 위한 연결 문자열입니다. 다음 연결 문자열을 사용합니다. jdbc:<subprotocol>:<subname> 다음 목록에서는 각 데이터베이스 유형에 입력할 수 있는 샘플 연결 문자열을 제공합니다. - Azure SQL 데이터베이스 JDBC 드라이버에 대한 연결 문자열: jdbc:sqlserver://<host>:<port>;database=<database_name> - Aurora PostgreSQL JDBC 드라이버에 대한 연결 문자열: jdbc:postgresql://<host>:<port>[/<database_name>] 특정 드라이버에서 사용할 연결 문자열에 대한 자세한 내용은 타사 공급업체 설명서를 참조하십시오.
하위 유형	연결하려는 데이터베이스 유형입니다. 다음 데이터베이스 유형 중에서 선택하여 연결할 수 있습니다. - Azure SQL 데이터베이스. Azure SQL 데이터베이스에 연결합니다. - PostgreSQL. Aurora PostgreSQL 데이터베이스에 연결합니다. - 기타. 유형 4 JDBC 드라이버를 지원하는 임의의 데이터베이스에 연결합니다.

속성	설명
대/소문자가 혼합된 식별자 지원	<p>데이터베이스에서 대/소문자를 구분하는 식별자를 사용할 경우 활성화합니다. 이 속성을 활성화하면 데이터 통합 서비스가 SQL 식별자 문자 속성에 대해 선택한 문자로 모든 식별자를 묶습니다.</p> <p>예를 들어 Aurora PostgreSQL 데이터베이스는 대/소문자가 혼합된 문자를 지원합니다. Aurora PostgreSQL 데이터베이스에 연결하려면 이 속성을 활성화해야 합니다.</p> <p>SQL 식별자 문자 속성이 없음으로 설정된 경우 대/소문자가 혼합된 식별자 지원 속성이 비활성화됩니다.</p>
SQL 식별자 문자	<p>데이터베이스에서 SQL 쿼리의 구분자로 분리된 식별자를 묶기 위해 사용하는 문자 유형입니다. 사용 가능한 문자는 데이터베이스 유형에 따라 다릅니다.</p> <p>데이터베이스에서 일반 식별자를 사용할 경우 (없음)을 선택합니다. 데이터 통합 서비스에서 SQL 쿼리를 생성할 때 식별자 주위에 구분자로 분리된 문자를 배치하지 않습니다.</p> <p>데이터베이스에서 구분자로 분리된 식별자를 사용할 경우 문자를 선택합니다. 데이터 통합 서비스가 SQL 쿼리를 생성할 때 구분자로 분리된 식별자를 이 문자로 묶습니다.</p>

JD Edwards EnterpriseOne 연결 속성

JD Edwards EnterpriseOne 개체에 연결하려면 JD Edwards EnterpriseOne 연결을 사용합니다.

다음 테이블에는 JD Edwards EnterpriseOne 연결 속성이 설명되어 있습니다.

속성	설명
이름	<p>연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 128자를 초과하거나 공백을 포함하거나 다음과 같은 특수 문자를 포함할 수 없습니다.</p> <p>~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /</p>
ID	<p>데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 생성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.</p>
설명	연결의 설명입니다. 설명은 765자를 초과할 수 없습니다.
위치	연결을 생성하려는 Informatica 도메인입니다.
유형	연결 유형입니다. JD Edwards EnterpriseOne을 선택합니다.
호스트 이름	JD Edwards EnterpriseOne 서버 호스트 이름입니다.
엔터프라이즈 포트	JD Edwards EnterpriseOne 서버 포트 번호입니다. 기본값은 6016입니다.
사용자 이름	JD Edwards EnterpriseOne 데이터베이스 사용자 이름입니다.
암호	JD Edwards EnterpriseOne 데이터베이스 사용자의 암호입니다.
환경	연결하려는 JD Edwards EnterpriseOne 환경의 이름입니다.
역할	JD Edwards EnterpriseOne 사용자의 역할입니다. 기본값은 *ALL입니다.

속성	설명
사용자 이름	JD Edwards EnterpriseOne 데이터베이스 사용자 이름입니다.
암호	데이터베이스 사용자의 암호입니다.
드라이버 클래스 이름	<p>다음 목록에서 해당하는 데이터베이스 유형에 입력할 수 있는 드라이버 클래스 이름을 제공합니다.</p> <ul style="list-style-type: none"> - Oracle용 DataDirect JDBC 드라이버 클래스 이름: <code>com.informatica.jdbc.oracle.OracleDriver</code> - IBM DB2용 DataDirect JDBC 드라이버 클래스 이름: <code>com.informatica.jdbc.db2.DB2Driver</code> - Microsoft SQL Server용 DataDirect JDBC 드라이버 클래스 이름: <code>com.informatica.jdbc.sqlserver.SQLServerDriver</code> <p>특정 데이터베이스와 같이 사용할 드라이버 클래스에 대한 자세한 정보는 공급 업체 설명서를 참조하십시오.</p>
연결 문자열	<p>데이터베이스에 연결하기 위한 연결 문자열입니다. 다음 연결 문자열을 사용합니다.</p> <p>JDBC 연결 문자열에서는 다음 구문을 사용합니다.</p> <ul style="list-style-type: none"> - Oracle: <code>jdbc:informatica:oracle://<호스트 이름>:<포트>,ServiceName=<db 서비스 이름></code> - DB2: <code>jdbc:informatica:db2://<호스트 이름>:<포트>;databaseName=<db 이름></code> - Microsoft SQL: <code>jdbc:informatica:sqlserver://<호스트 이름>:<포트>;databaseName=<db 이름></code>

LDAP 연결 속성

LDAP 개체에 연결하려면 LDAP 연결을 사용합니다.

다음 테이블에는 LDAP 연결 속성이 설명되어 있습니다.

속성	설명
이름	<p>연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 128자를 초과하거나 공백을 포함하거나 다음과 같은 특수 문자를 포함할 수 없습니다.</p> <p>~ ` ! \$ % ^ & * () - + = { [] } \ ; " ' < , > . ? /</p>
ID	<p>데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 생성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.</p>
설명	연결의 설명입니다. 설명은 765자를 초과할 수 없습니다.
위치	연결을 생성하려는 Informatica 도메인입니다.
유형	연결 유형입니다. LDAP를 선택합니다.
호스트 이름	LDAP 디렉터리 서버 호스트 이름입니다. 기본값은 localhost입니다.

속성	설명
포트	LDAP 디렉터리 서버 포트 번호입니다. 기본값은 389입니다.
익명 연결	LDAP 디렉터리 서버와의 익명 연결을 설정합니다. 인증 없이 익명 사용자로 디렉터리 서버에 액세스하려면 익명 연결을 선택합니다. 참고: Active Directory와의 익명 연결을 설정할 수 없습니다.
사용자 이름	LDAP 디렉터리 서버에 연결하기 위한 LDAP 사용자 이름입니다.
암호	LDAP 디렉터리 서버에 연결하기 위한 암호입니다.
보안 연결	TLS 프로토콜을 통해 LDAP 디렉터리 서버와의 보안 연결을 설정합니다.
TrustStore 파일 이름	LDAP 디렉터리 서버와의 보안 연결을 설정하기 위한 TLS 인증서가 포함된 트러스트 저장소의 파일 이름입니다. 기본값은 infa_truststore.jks입니다. 보안 연결을 선택하는 경우에 필요합니다. 트러스트 저장소 파일 이름 및 암호는 LDAP 관리자에게 문의하십시오.
TrustStore 암호	SSL 인증서가 포함된 트러스트 저장소 파일의 암호입니다.
KeyStore 파일 이름	LDAP 디렉터리 서버와의 보안 통신을 설정하는 데 필요한 키 및 인증서가 포함된 키 저장소의 파일 이름입니다. 보안 연결을 선택하는 경우에 필요합니다. 키 저장소 파일 이름 및 암호는 LDAP 관리자에게 문의하십시오.
KeyStore 암호	보안 통신에 필요한 키 저장소 파일의 암호입니다.

LinkedIn 연결 속성

LinkedIn 웹 사이트에서 데이터를 추출하려면 LinkedIn 연결을 사용합니다. LinkedIn 연결은 소셜 미디어 유형 연결입니다. Administrator 도구 또는 Developer tool에서 LinkedIn 연결을 생성하고 관리할 수 있습니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 표에는 LinkedIn 연결 속성이 설명되어 있습니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결의 설명입니다. 설명은 765자를 초과할 수 없습니다.
위치	연결을 생성하려는 도메인입니다.

속성	설명
유형	연결 유형입니다. LinkedIn을 선택합니다.
OAuth 세부 정보가 있으십니까?	OAuth를 구성할지 여부를 나타냅니다. 다음 값 중 하나를 선택합니다. - 예. 액세스 토큰 및 암호가 있음을 나타냅니다. - 아니요. OAuth 유틸리티를 시작합니다.
소비자 키	LinkedIn에서 응용 프로그램을 생성할 때 얻게 되는 API 키입니다. LinkedIn에서는 이 키를 사용하여 응용 프로그램을 식별합니다.
소비자 암호	LinkedIn에서 응용 프로그램을 생성할 때 얻게 되는 암호 키입니다. LinkedIn에서는 이 암호를 사용하여 소비자 키의 소유권을 설정합니다.
액세스 토큰	OAuth 유틸리티가 반환하는 액세스 토큰입니다. LinkedIn 응용 프로그램에서는 사용자 자격 증명 대신 이 토큰을 사용하여 보호된 리소스에 액세스합니다.
액세스 암호	OAuth 유틸리티가 반환하는 액세스 암호입니다. 이 암호는 토큰의 소유권을 설정합니다.
범위	선택 사항입니다. 응용 프로그램에 대한 사용 권한입니다. OAuth를 구성하는 데 사용한 사용 권한을 입력합니다.

Microsoft Azure Blob Storage 연결 속성

Microsoft Azure Blob Storage에 액세스하려면 Microsoft Azure SQL Blob Storage 연결을 사용합니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

Administrator 도구 또는 Developer tool에서 Microsoft Azure Blob Storage 연결을 생성하고 관리할 수 있습니다. 다음 테이블에는 Microsoft Azure Blob Storage 연결 속성이 설명되어 있습니다.

속성	설명
이름	Microsoft Azure Blob Storage 연결의 이름입니다.
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결에 대한 설명입니다.
위치	연결을 작성하려는 도메인입니다.
유형	연결 유형입니다. Azure Blob Storage를 선택합니다.

연결 세부 정보 탭에는 Microsoft Azure Blob Storage 연결의 연결 특성이 포함되어 있습니다. 다음 테이블에는 연결 특성이 설명되어 있습니다.

속성	설명
계정 이름	Microsoft Azure 저장소 계정의 이름입니다.
권한 부여 유형	권한 부여 유형입니다. 다음과 같은 권한 부여 메커니즘을 선택할 수 있습니다. <ul style="list-style-type: none"> - 공유 키 권한 부여 - 공유 액세스 서명
계정 키	Microsoft Azure Storage 액세스 키입니다. 공유 키 권한 부여를 선택할 때 적용됩니다.
SAS 토큰	사용자 계정에 대해 Microsoft Azure 포털에서 생성하는 SAS 토큰이 포함된 SAS URI. 공유 액세스 서명 권한 부여 유형을 선택할 때 적용됩니다. 참고: 올바른 SAS 토큰이 포함된 올바른 SAS URI를 제공해야 합니다.
컨테이너 이름	절대 경로가 있는 하위 폴더 또는 루트 컨테이너입니다. 참고: 복합 파일을 가져오려면 루트 컨테이너만 지정합니다.
끝점 접미사	Microsoft Azure 끝점의 유형입니다. 다음과 같은 끝점을 선택할 수 있습니다. <ul style="list-style-type: none"> - core.windows.net: 기본값 - core.usgovcloudapi.net: US Government Microsoft Azure 끝점을 선택하려는 경우 - core.chinacloudapi.cn: 해당 없음

Microsoft Azure Cosmos DB SQL API 연결 속성

Cosmos DB 데이터베이스에 연결하려면 Microsoft Azure Cosmos DB 연결을 사용합니다. Microsoft Azure Cosmos DB 연결을 생성할 때 메타데이터 및 데이터 액세스 관련 정보를 입력합니다.

다음 테이블에는 Microsoft Azure Cosmos DB 연결 속성이 설명되어 있습니다.

속성	설명
이름	Cosmos DB 연결의 이름입니다.
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 생성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결에 대한 설명입니다. 설명은 765자를 초과할 수 없습니다.
위치	Cosmos DB 연결을 저장할 모델 리포지토리의 프로젝트 또는 폴더입니다.
유형	Microsoft Azure Cosmos DB SQL API를 선택합니다.
Cosmos DB URI	Microsoft Azure Cosmos DB 계정의 URI입니다.

속성	설명
키	Microsoft Azure Cosmos DB 계정 내의 리소스에 대한 전체 관리 액세스 권한을 제공할 기본 및 보조 키입니다.
데이터베이스	JSON 문서를 읽거나 쓸 컬렉션을 포함하는 데이터베이스의 이름입니다.

참고: Azure 포털의 키 설정에서 Cosmos DB URI 및 키 값을 찾을 수 있습니다. 자세한 내용은 Azure 관리자에게 문의하십시오.

Microsoft Azure Data Lake Storage Gen1 연결 속성

Microsoft Azure Data Lake Storage Gen1에 액세스하려면 Microsoft Azure Data Lake Storage Gen1 연결을 사용합니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

Administrator 도구 또는 Developer tool에서 Microsoft Azure Data Lake Storage Gen1 연결을 생성하고 관리할 수 있습니다. 다음 테이블에는 Microsoft Azure Data Lake Storage Gen1 연결 속성이 설명되어 있습니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 있습니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~`!\$%^&*()-+=[] \\:;\"'<, > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 생성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결의 설명입니다. 설명은 4,000자를 초과할 수 없습니다.
위치	연결을 작성하려는 도메인입니다.
유형	연결 유형입니다. Microsoft Azure Data Lake Storage Gen1을 선택합니다.

다음 테이블에는 메타데이터 액세스에 대한 속성이 설명되어 있습니다.

속성	설명
ADLS 계정 이름	Microsoft Azure Data Lake Storage Gen1의 이름입니다.
ClientID	Active Directory의 OAuth 인증을 완료할 때 사용할 응용 프로그램의 ID입니다.
클라이언트 암호	Active Directory의 OAuth 인증을 완료할 때 사용할 클라이언트 암호 키입니다.

속성	설명
디렉터리	데이터를 읽거나 쓸 때 사용하는 Microsoft Azure Data Lake Storage Gen1 디렉터리입니다. 기본값은 루트 디렉터리입니다.
AuthEndpoint	클라이언트 ID에 기반한 액세스 코드가 생성되고 클라이언트 암호가 입력되는 OAuth 2.0 토큰 끝점입니다.

클라이언트 ID, 클라이언트 암호 및 인증 끝점 생성에 대한 자세한 내용은 **Azure** 관리자에게 문의하거나 **Microsoft Azure Data Lake Storage Gen1** 설명서를 참조하십시오.

Microsoft Azure Data Lake Storage Gen2 연결 속성

Microsoft Azure Data Lake Storage Gen2에 액세스하려면 Microsoft Azure Data Lake Storage Gen2 연결을 사용합니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

Administrator 도구 또는 Developer tool에서 Microsoft Azure Data Lake Storage Gen2 연결을 생성하고 관리할 수 있습니다. 다음 테이블에는 Microsoft Azure Data Lake Storage Gen2 연결 속성이 설명되어 있습니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 있습니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~ ` ! \$ % ^ & * () - + = { [] } \ ; " ' < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 생성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결의 설명입니다. 설명은 4,000자를 초과할 수 없습니다.
위치	연결을 작성하려는 도메인입니다.
유형	연결 유형입니다. Microsoft Azure Data Lake Storage Gen2를 선택합니다.

다음 테이블에는 메타데이터 액세스에 대한 속성이 설명되어 있습니다.

속성	설명
계정 이름	Microsoft Azure Data Lake Storage Gen2 계정 이름 또는 서비스 이름입니다.
클라이언트 ID	Azure Active Directory(AD)의 OAuth 인증을 완료할 때 사용할 응용 프로그램의 ID입니다.
클라이언트 암호	Azure AD의 OAuth 인증을 완료할 때 사용할 클라이언트 암호 키입니다.
테넌트 ID	Azure AD의 디렉터리 ID입니다.
파일 시스템 이름	Microsoft Azure Data Lake Storage Gen2에 있는 기존 파일 시스템의 이름입니다.

속성	설명
디렉터리 경로	파일 시스템 이름이 없는 기존 디렉터리의 경로입니다. 기본 디렉터리는 없습니다. 다음 구문 중 하나를 선택할 수 있습니다. <ul style="list-style-type: none"> - /(루트 디렉터리의 경우) - /dir1 - dir1/dir2
Adls Gen2 끝점	Microsoft Azure 끝점의 유형입니다. 다음과 같은 끝점을 선택할 수 있습니다. <ul style="list-style-type: none"> - core.windows.net: 기본값 - core.usgovcloudapi.net: Azure Government 끝점을 선택하는 경우에 사용합니다.

클라이언트 ID, 클라이언트 암호, 테넌트 ID 및 파일 시스템 이름 생성에 대한 자세한 내용은 Azure 관리자에게 문의하거나 Microsoft Azure Data Lake Storage Gen2 설명서를 참조하십시오.

Microsoft Azure SQL 데이터 웨어하우스 연결 속성

Microsoft Azure 데이터 웨어하우스에 액세스하려면 Microsoft Azure SQL 데이터 웨어하우스 연결을 사용합니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

Administrator 도구 또는 Developer tool에서 Microsoft Azure SQL 데이터 웨어하우스 연결을 생성하고 관리할 수 있습니다. 다음 테이블에는 Microsoft Azure SQL 데이터 웨어하우스 연결 속성이 설명되어 있습니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 있습니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~`!\$%^&*()-+=[{ } \ ; ' " < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결의 설명입니다. 설명은 4,000자를 초과할 수 없습니다.
위치	연결을 작성하려는 도메인입니다.
유형	연결 유형입니다. Azure SQL 데이터 웨어하우스를 선택합니다.

다음 테이블에는 메타데이터 액세스에 대한 속성이 설명되어 있습니다.

속성	설명
Azure DW JDBC URL	Microsoft Azure 데이터 웨어하우스 JDBC 연결 문자열입니다. 예를 들어 다음과 같은 연결 문자열을 입력할 수 있습니다. jdbc:sqlserver:// <서버>.database.windows.net:1433;database=<데이터베이스>. 관리자는 Microsoft Azure 포털에서 URL을 다운로드할 수 있습니다.
Azure DW JDBC 사용자 이름	Microsoft Azure SQL 데이터 웨어하우스 계정에 연결하기 위한 사용자 이름입니다. Microsoft Azure SQL 데이터 웨어하우스에서 데이터를 읽고 쓰고 잘라낼 수 있는 사용 권한이 있어야 합니다.
Azure DW JDBC 암호	Microsoft Azure SQL 데이터 웨어하우스 계정에 연결하기 위한 암호입니다.
Azure DW 스키마 이름	Microsoft Azure SQL 데이터 웨어하우스의 스키마 이름입니다.
Azure Blob 계정 이름	파일을 준비하기 위한 Microsoft Azure 저장소 계정의 이름입니다.
Azure Blob 계정 키	Blob 저장소 계정에 대한 액세스를 인증하는 키입니다.
Blob 끝점	Microsoft Azure 끝점의 유형입니다. 다음과 같은 끝점을 선택할 수 있습니다. <ul style="list-style-type: none"> - core.windows.net: 기본값 - core.usgovcloudapi.net: US Government Microsoft Azure 끝점을 선택하려는 경우 - core.chinacloudapi.cn: 해당 없음 원시 환경 및 Spark 엔진에서 매핑을 실행하는 경우 US Government Microsoft Azure 끝점을 구성할 수 있습니다.
VNet 규칙	VNet(가상 네트워크)에 상주하는 Microsoft Azure SQL 데이터 웨어하우스 끝점에 연결하려면 활성화합니다.

MS SQL Server 연결 속성

Microsoft SQL Server 연결을 사용하여 Microsoft SQL Server에 액세스합니다. Microsoft SQL Server 연결은 Microsoft SQL Server 관계형 데이터베이스에 대한 연결입니다. Administrator 도구 또는 Developer tool에서 Microsoft SQL Server 연결을 생성하고 관리할 수 있습니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 테이블에는 MS SQL Server 연결 속성이 설명되어 있습니다.

속성	설명
데이터베이스 유형	데이터베이스 유형입니다.
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /

속성	설명
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 생성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결의 설명입니다. 설명은 765자를 초과할 수 없습니다.
트러스트된 연결 사용	응용 프로그램 서비스가 Windows 인증을 사용하여 데이터베이스에 액세스하도록 합니다. 응용 프로그램 서비스를 시작하는 사용자 이름은 데이터베이스 액세스 권한이 있는 올바른 Windows 사용자여야 합니다. 기본적으로 이 옵션은 비활성화되어 있습니다. 참고: Windows 및 NTLM 인증은 Linux에서 호스팅되는 Microsoft SQL Server 2017 버전에 대해 인증되지 않았습니다.
사용자 이름	데이터베이스 사용자 이름입니다. Microsoft SQL Server가 NTLMv1 또는 NTLMv2 인증을 사용하는 경우에 필요합니다.
암호	데이터베이스 사용자 이름에 대한 암호입니다. Microsoft SQL Server가 NTLMv1 또는 NTLMv2 인증을 사용하는 경우에 필요합니다.
통과 보안 활성화됨	연결에 대한 통과 보안을 활성화합니다. 연결에 대한 통과 보안을 활성화한 경우 도메인은 연결 개체에 정의된 자격 증명 대신 클라이언트 사용자 이름과 암호를 사용하여 해당하는 데이터베이스에 로그인합니다.
메타데이터 액세스 속성: 연결 문자열	<p>데이터베이스에서 메타데이터에 액세스하는 데 사용되는 연결 문자열입니다. 다음 연결 문자열을 사용합니다.</p> <pre>jdbc:informatica:sqlserver://<host name>:<port>;DatabaseName=<database name></pre> <p>NTLM 인증과의 연결을 테스트하려면 연결 문자열에 다음 매개 변수를 포함합니다.</p> <ul style="list-style-type: none"> - AuthenticationMethod. 사용할 NTLM 인증 버전입니다. <p>참고: UNIX는 NTLM이 아닌 NTLMv1 및 NTLMv2를 지원합니다.</p> <ul style="list-style-type: none"> - 도메인. SQL 서버가 속한 도메인입니다. <p>다음 예는 이름이 Informatica.com인 NT 도메인에서 NTLMv2 인증을 사용하는 SQL 서버에 대한 연결 문자열을 보여 줍니다.</p> <pre>jdbc:informatica:sqlserver://host01:1433;DatabaseName=SQL1;AuthenticationMethod=ntlm2java;Domain=Informatica.com</pre> <p>NTLM 인증에 연결하는 경우 MS SQL Server 연결 속성의 트러스트된 연결 사용 옵션을 활성화할 수 있습니다. NTLMv1 또는 NTLMv2 인증에 연결하는 경우 연결 속성에서 사용자 이름 및 암호를 제공해야 합니다.</p>

속성	설명
AdvancedJDBCSecurityOptions	<p>보안 데이터베이스에 메타데이터 액세스하기 위한 데이터베이스 매개 변수입니다. Informatica는 AdvancedJDBCSecurityOptions 필드 값을 중요한 데이터로 처리하고 매개 변수 문자열을 암호화하여 저장합니다.</p> <p>보안 데이터베이스에 연결하려면 다음 매개 변수를 포함합니다.</p> <ul style="list-style-type: none"> - EncryptionMethod. 필수 사항입니다. 네트워크를 통해 전송되는 경우 데이터가 암호화되었는지를 나타냅니다. 이 매개 변수는 SSL로 설정되어야 합니다. - ValidateServerCertificate. 선택 사항입니다. 데이터베이스 서버에서 보낸 인증서에 대해 Informatica에서 유효성을 검사하는지를 나타냅니다. <p>이 매개 변수를 True로 설정하면 데이터베이스 서버에서 보낸 인증서에 대해 Informatica에서 유효성을 검사합니다. HostNameInCertificate 매개 변수를 지정하면 Informatica에서 인증서의 호스트 이름에 대한 유효성도 검사합니다.</p> <p>이 매개 변수를 false로 설정하면 데이터베이스 서버에서 보낸 인증서에 대해 Informatica에서 유효성을 검사하지 않습니다. Informatica에서 사용자가 지정한 트러스트 저장소 정보를 모두 무시합니다.</p> <ul style="list-style-type: none"> - HostNameInCertificate. 선택 사항입니다. 보안 데이터베이스를 호스팅하는 시스템의 호스트 이름입니다. 호스트 이름을 지정하면 Informatica에서 SSL 인증서의 호스트 이름에 대해 연결 문자열에 포함된 호스트 이름의 유효성을 검사합니다. - cryptoProtocolVersion. 선택 사항입니다. Microsoft SQL Server 인스턴스에 대해 TLS를 활성화하는 경우 cryptoProtocolVersion 매개 변수를 다음과 같이 설정합니다. <p>cryptoProtocolVersion=TLSv<버전 번호>. 예: cryptoProtocolVersion=TLSv1.2</p> <p>참고: 버전 번호는 서버에 대해 구성된 TLS 버전과 동일해야 합니다.</p> <ul style="list-style-type: none"> - TrustStore. 필수 사항입니다. 트러스트 저장소 파일의 경로 및 파일 이름입니다. <p>참고: SSL 또는 TLS를 구성하고 파일 이름만 지정하는 경우 트러스트 저장소 파일을 Informatica 설치 디렉터리로 복사해야 합니다. 연결을 테스트하고 메타데이터를 가져오려면 트러스트 저장소 파일을 다음 디렉터리로 복사합니다.</p> <p><Informatica 클라이언트 설치 디렉터리>/clients/DeveloperClient</p> <p>매핑을 실행하려면 트러스트 저장소 파일을 다음 디렉터리로 복사합니다.</p> <p><Informatica 서버 설치 디렉터리>/tomcat/bin</p> <ul style="list-style-type: none"> - TrustStorePassword. 필수 사항입니다. 보안 데이터베이스에 대한 트러스트 저장소 파일의 암호입니다. <p>ODBC에는 적용되지 않습니다.</p> <p>참고: Informatica에서 보안 JDBC 매개 변수를 연결 문자열에 추가합니다. 보안 JDBC 매개 변수를 직접 연결 문자열에 포함한 경우 AdvancedJDBCSecurityOptions 필드에서 매개 변수를 입력하지 마십시오.</p>
데이터 액세스 속성: 공급자 유형	<p>Microsoft SQL Server 데이터베이스에 연결하는 데 사용하려는 연결 공급자입니다. 다음과 같은 공급자 유형을 선택할 수 있습니다.</p> <ul style="list-style-type: none"> - ODBC - Oldeb(더 이상 사용되지 않음) <p>기본값은 ODBC입니다.</p> <p>참고: Microsoft SQL Server 연결 사용자 인터페이스에는 OLEDB 공급자 유형이 더 이상 사용되지 않는 것으로 표시되지만 Informatica에서는 OLEDB 공급자 유형이 지원됩니다. OLEDB 공급자 유형 지원 방침에 대한 자세한 내용은 다음 기술 자료 문서를 참조하십시오. KB 522895</p>
DSN 사용	<p>데이터 통합 서비스가 연결을 위한 데이터 소스 이름을 사용할 수 있습니다.</p> <p>DSN 사용 옵션을 선택하는 경우 데이터 통합 서비스가 DSN에서 데이터베이스 및 서버 이름을 검색합니다.</p> <p>DSN 사용 옵션을 선택하지 않는 경우 데이터베이스 및 서버 이름을 제공해야 합니다.</p>

속성	설명
연결 문자열	<p>DSN 모드를 활성화하지 않는 경우 다음 연결 문자열을 사용합니다.</p> <p><server name>@<database name></p> <p>DSN 모드를 활성화하는 경우 다음 연결 문자열을 사용합니다.</p> <p><DSN Name></p>
코드 페이지	소스 데이터베이스에서 읽거나 대상 데이터베이스 또는 파일에 쓰는 데 사용되는 코드 페이지입니다.
도메인 이름	도메인의 이름입니다.
패킷 크기	데이터를 전송하는 데 사용되는 패킷 크기입니다. Microsoft SQL Server에 대한 원시 드라이버를 최적화하는 데 사용됩니다.
소유자 이름	<p>스키마 소유자의 이름입니다.</p> <p>참고: 동적 매핑을 통해서 또는 DDL 생성 및 실행 옵션을 통해 테이블 DDL을 생성하는 경우 DDL 메타데이터는 스키마 이름 및 소유자 이름 속성을 포함하지 않습니다.</p>
스키마 이름	<p>데이터베이스에 있는 스키마의 이름입니다. 스키마 이름이 데이터베이스 사용자 이름과 다를 경우 프로파일링 웨어하우스에 대한 스키마 이름을 지정해야 합니다. 스키마 이름이 데이터베이스 사용자 이름과 다르고 사용자 관리 캐시 테이블을 구성하는 경우 데이터 개체 캐시 데이터베이스의 스키마 이름을 지정해야 합니다.</p> <p>참고: 동적 매핑을 통해서 또는 DDL 생성 및 실행 옵션을 통해 테이블 DDL을 생성하는 경우 DDL 메타데이터는 스키마 이름 및 소유자 이름 속성을 포함하지 않습니다.</p>
환경 SQL	데이터베이스에 연결할 때 데이터베이스 환경을 설정하는 SQL 명령입니다. 데이터 통합 서비스에서 데이터베이스에 연결할 때마다 연결 환경 SQL을 실행합니다.
트랜잭션 SQL	데이터베이스에 연결할 때 데이터베이스 환경을 설정하는 SQL 명령입니다. 데이터 통합 서비스에서 각 트랜잭션을 시작할 때 트랜잭션 환경 SQL을 실행합니다.
다시 시도 기간	이 속성은 나중에 사용하기 위해 예약되어 있습니다.
SQL 식별자 문자	<p>데이터베이스에서 SQL 쿼리의 구분자로 분리된 식별자를 묶기 위해 사용하는 문자 유형입니다. 사용 가능한 문자는 데이터베이스 유형에 따라 다릅니다.</p> <p>데이터베이스에서 일반 식별자를 사용할 경우 (없음)을 선택합니다. 데이터 통합 서비스에서 SQL 쿼리를 생성할 때 식별자 주위에 구분자로 분리된 문자를 배치하지 않습니다.</p> <p>데이터베이스에서 구분자로 분리된 식별자를 사용할 경우 문자를 선택합니다. 데이터 통합 서비스가 SQL 쿼리를 생성할 때 구분자로 분리된 식별자를 이 문자로 묶습니다.</p>
대/소문자가 혼합된 식별자 지원	<p>데이터베이스에서 대/소문자를 구분하는 식별자를 사용할 경우 활성화합니다. 이 속성을 활성화하면 데이터 통합 서비스가 SQL 식별자 문자 속성에 대해 선택한 문자로 모든 식별자를 묶습니다.</p> <p>SQL 식별자 문자 속성이 없음으로 설정된 경우 대/소문자가 혼합된 식별자 지원 속성이 비활성화됩니다.</p>
ODBC 공급자	<p>ODBC. ODBC가 연결되는 데이터베이스의 유형입니다. 푸시다운 최적화를 위해 데이터베이스 유형을 지정하여 데이터 통합 서비스가 원시 데이터베이스 SQL을 생성할 수 있도록 합니다. 옵션은 다음과 같습니다.</p> <ul style="list-style-type: none"> - 기타 - Sybase - Microsoft_SQL_Server <p>기본값은 기타입니다.</p>

Netezza 연결 속성

Netezza 연결을 사용하여 Netezza 데이터베이스에 액세스합니다. Netezza 연결은 데이터베이스 연결입니다. Administrator 도구 또는 Developer tool에서 Netezza 연결을 생성하고 관리할 수 있습니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 테이블에는 Netezza 연결 속성이 설명되어 있습니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결에 대한 설명입니다. 설명은 765자를 초과할 수 없습니다.
위치	연결을 생성하려는 도메인입니다.
유형	연결 유형입니다. Netezza를 선택합니다.
사용자 이름	Netezza 데이터베이스에 액세스하기 위한 적절한 사용 권한이 있는 사용자 이름입니다.
암호	데이터베이스 사용자 이름에 대한 암호입니다.
JDBC URL	Developer tool이 Netezza 데이터베이스에 연결할 때 사용해야 하는 JDBC URL입니다. 다음 형식을 사용합니다. jdbc:netezza://<호스트 이름>:<포트>/<데이터베이스 이름>
연결 문자열	Netezza 데이터베이스에 연결하기 위해 사용하려는 ODBC 데이터 소스의 이름입니다.
제한 시간	Developer tool이 연결을 닫기 전에 Netezza 데이터베이스에서 응답을 기다리는 시간(초)입니다.

OData 연결 속성

OData 연결을 사용하여 OData URL에 액세스합니다. OData 연결은 웹 연결입니다. Administrator 도구 또는 Developer tool에서 OData 연결을 생성하고 관리할 수 있습니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 테이블에는 OData 연결 속성이 설명되어 있습니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결에 대한 설명입니다. 설명은 4,000자를 초과할 수 없습니다.
위치	연결을 생성하려는 도메인입니다.
유형	연결 유형입니다. OData를 선택합니다.
사용자 이름	선택 사항입니다. OData 리소스에서 데이터를 읽기 위한 적절한 사용 권한이 있는 사용자 이름입니다.
암호	선택 사항입니다. OData URL 사용자 이름의 암호입니다.
URL	읽으려는 데이터를 노출하는 OData 서비스 루트 URL입니다.
보안 유형	선택 사항입니다. Developer tool이 OData 서버와의 보안 연결을 설정하는 데 사용해야 하는 보안 프로토콜입니다. 다음 값 중 하나를 선택합니다. - 없음 - SSL - TLS 기본값은 없음입니다.
TrustStore 파일 이름	보안 유형을 선택하는 경우 필수입니다. OData 서버에 대한 공용 인증서가 포함된 트러스트 저장소 파일의 이름입니다. 기본값은 infa_truststore.jks입니다.
암호	보안 유형을 선택하는 경우 필수입니다. OData 서버에 대한 공용 인증서가 포함된 트러스트 저장소 파일의 암호입니다.
KeyStore 파일 이름	보안 유형을 선택하는 경우 필수입니다. OData 서버에 대한 개인 키가 포함된 키 저장소 파일의 이름입니다. 기본값은 infa_truststore.jks입니다.
암호	보안 유형을 선택하는 경우 필수입니다. OData 서버에 대한 개인 키가 포함된 키 저장소 파일의 암호입니다.

ODBC 연결 속성

ODBC 연결을 사용하여 ODBC 데이터에 액세스합니다. ODBC 연결은 관계형 데이터베이스 연결입니다. Administrator 도구, Developer tool 또는 Analyst 도구에서 ODBC 연결을 작성하고 관리할 수 있습니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 테이블에는 ODBC 연결 속성이 설명되어 있습니다.

속성	설명
데이터베이스 유형	데이터베이스 유형입니다.
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 생성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결의 설명입니다. 설명은 765자를 초과할 수 없습니다.
사용자 이름	데이터베이스 사용자 이름입니다.
암호	데이터베이스 사용자 이름에 대한 암호입니다.
통과 보안 활성화됨	연결에 대한 통과 보안을 활성화합니다. 연결에 대한 통과 보안을 활성화한 경우 도메인은 연결 개체에 정의된 자격 증명 대신 클라이언트 사용자 이름과 암호를 사용하여 해당하는 데이터베이스에 로그인합니다.
데이터 액세스 속성: 연결 문자열	데이터베이스에서 메타데이터에 액세스하는 데 사용되는 ODBC 연결 URL입니다. <데이터 소스 이름>
코드 페이지	소스 데이터베이스에서 읽거나 대상 데이터베이스 또는 파일에 쓰는 데 사용되는 코드 페이지입니다.
환경 SQL	데이터베이스에 연결할 때 데이터베이스 환경을 설정하는 SQL 명령입니다. 데이터 통합 서비스에서 데이터베이스에 연결할 때마다 연결 환경 SQL을 실행합니다.
트랜잭션 SQL	데이터베이스에 연결할 때 데이터베이스 환경을 설정하는 SQL 명령입니다. 데이터 통합 서비스에서 각 트랜잭션을 시작할 때 트랜잭션 환경 SQL을 실행합니다.
다시 시도 기간	이 속성은 나중에 사용하기 위해 예약되어 있습니다.
SQL 식별자 문자	데이터베이스에서 SQL 쿼리의 구분자로 분리된 식별자를 묶기 위해 사용하는 문자 유형입니다. 사용 가능한 문자는 데이터베이스 유형에 따라 다릅니다. 데이터베이스에서 일반 식별자를 사용할 경우 (없음)을 선택합니다. 데이터 통합 서비스에서 SQL 쿼리를 생성할 때 식별자 주위에 구분자로 분리된 문자를 배치하지 않습니다. 데이터베이스에서 구분자로 분리된 식별자를 사용할 경우 문자를 선택합니다. 데이터 통합 서비스가 SQL 쿼리를 생성할 때 구분자로 분리된 식별자를 이 문자로 묶습니다.

속성	설명
대/소문자가 혼합된 식별자 지원	데이터베이스에서 대/소문자를 구분하는 식별자를 사용할 경우 활성화합니다. 이 속성을 활성화하면 데이터 통합 서비스가 SQL 식별자 문자 속성에 대해 선택한 문자로 모든 식별자를 묶습니다. SQL 식별자 문자 속성이 없음으로 설정된 경우 대/소문자가 혼합된 식별자 지원 속성이 비활성화됩니다.
ODBC 공급자	ODBC가 연결되는 데이터베이스의 유형입니다. 푸시다운 최적화를 위해 데이터베이스 유형을 지정하여 데이터 통합 서비스가 원시 데이터베이스 SQL을 생성할 수 있도록 합니다. 옵션은 다음과 같습니다. - 기타 - Sybase - Microsoft_SQL_Server - Snowflake 기본값은 기타입니다.

참고: 데이터 통합 서비스가 UNIX 또는 Linux에서 실행되는 경우 ODBC 연결을 사용하여 Microsoft SQL Server에 연결합니다. 데이터 통합 서비스가 Windows에서 실행되는 경우 원시 연결을 사용하여 Microsoft SQL Server에 연결합니다.

Oracle 연결 속성

Oracle 연결을 사용하여 Oracle 데이터베이스에 연결합니다. Oracle 연결은 관계형 연결 유형입니다. Administrator 도구, Developer tool 또는 Analyst 도구에서 Oracle 연결을 작성하고 관리할 수 있습니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 표에는 Oracle 연결 속성이 설명되어 있습니다.

속성	설명
데이터베이스 유형	데이터베이스 유형입니다.
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 생성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결의 설명입니다. 설명은 765자를 초과할 수 없습니다.
사용자 이름	데이터베이스 사용자 이름입니다.
암호	데이터베이스 사용자 이름에 대한 암호입니다.
통과 보안 활성화됨	연결에 대한 통과 보안을 활성화합니다. 연결에 대한 통과 보안을 활성화한 경우 도메인은 연결 개체에 정의된 자격 증명 대신 클라이언트 사용자 이름과 암호를 사용하여 해당하는 데이터베이스에 로그인합니다.

속성	설명
메타데이터 액세스 속성: 연결 문자열	<p>데이터베이스에서 메타데이터에 액세스하는 데 사용되는 연결 문자열입니다. 다음 연결 문자열을 사용합니다.</p> <pre>jdbc:informatica:oracle://<host_name>:<port>;SID=<database name></pre> <p>다음 연결 문자열을 사용하여 Oracle 연결 관리자를 통해 Oracle 데이터베이스에 연결합니다.</p> <pre>jdbc:informatica:oracle:TNSNamesFile=<fully qualified path to the tnsnames.ora file>;TNSServerName=<TNS server name>;</pre>
AdvancedJDBCSecurityOptions	<p>보안 데이터베이스에 메타데이터 액세스하기 위한 데이터베이스 매개 변수입니다. Informatica는 AdvancedJDBCSecurityOptions 필드 값을 중요한 데이터로 처리하고 매개 변수 문자열을 암호화하여 저장합니다.</p> <p>보안 데이터베이스에 연결하려면 다음 매개 변수를 포함합니다.</p> <ul style="list-style-type: none"> - EncryptionMethod. 필수 사항입니다. 네트워크를 통해 전송되는 경우 데이터가 암호화되었는지를 나타냅니다. 이 매개 변수는 SSL로 설정되어야 합니다. - ValidateServerCertificate. 선택 사항입니다. 데이터베이스 서버에서 보낸 인증서에 대해 Informatica에서 유효성을 검사하는지를 나타냅니다. 이 매개 변수를 True로 설정하면 데이터베이스 서버에서 보낸 인증서에 대해 Informatica에서 유효성을 검사합니다. HostNameInCertificate 매개 변수를 지정하면 Informatica에서 인증서의 호스트 이름에 대한 유효성도 검사합니다. 이 매개 변수를 false로 설정하면 데이터베이스 서버에서 보낸 인증서에 대해 Informatica에서 유효성을 검사하지 않습니다. Informatica에서 사용자가 지정한 트러스트 저장소 정보를 모두 무시합니다. - HostNameInCertificate. 선택 사항입니다. 보안 데이터베이스를 호스팅하는 시스템의 호스트 이름입니다. 호스트 이름을 지정하면 Informatica에서 SSL 인증서의 호스트 이름에 대해 연결 문자열에 포함된 호스트 이름의 유효성을 검사합니다. - cryptoProtocolVersion. 선택 사항입니다. Oracle 인스턴스에 대해 TLS를 활성화하는 경우 cryptoProtocolVersion 매개 변수를 다음과 같이 설정합니다. cryptoProtocolVersion=TLSv<버전 번호>. 예: cryptoProtocolVersion=TLSv1.2 참고: 버전 번호는 서버에 대해 구성한 TLS 버전과 동일해야 합니다. - TrustStore. 필수 사항입니다. 트러스트 저장소 파일의 경로 및 파일 이름입니다. 참고: SSL 또는 TLS를 구성하고 파일 이름만 지정하는 경우 트러스트 저장소 파일을 Informatica 설치 디렉터리로 복사해야 합니다. 연결을 테스트하고 메타데이터를 가져오려면 트러스트 저장소 파일을 다음 디렉터리로 복사합니다. <Informatica 클라이언트 설치 디렉터리>/clients/DeveloperClient 매핑을 실행하려면 트러스트 저장소 파일을 다음 디렉터리로 복사합니다. <Informatica 서버 설치 디렉터리>/tomcat/bin - TrustStorePassword. 필수 사항입니다. 보안 데이터베이스에 대한 트러스트 저장소 파일의 암호입니다. - KeyStore. 필수 사항입니다. 키 저장소 파일의 경로 및 파일 이름입니다. - KeyStorePassword. 필수 사항입니다. 보안 데이터베이스에 대한 키 저장소 파일의 암호입니다. <p>참고: Informatica에서 보안 JDBC 매개 변수를 연결 문자열에 추가합니다. 보안 JDBC 매개 변수를 직접 연결 문자열에 포함한 경우 AdvancedJDBCSecurityOptions 필드에서 매개 변수를 입력하지 마십시오.</p>
데이터 액세스 속성: 연결 문자열	<p>다음 연결 문자열을 사용합니다.</p> <pre><database name>.world</pre>

속성	설명
코드 페이지	소스 데이터베이스에서 읽거나 대상 데이터베이스 또는 파일에 쓰는 데 사용되는 코드 페이지입니다.
환경 SQL	데이터베이스에 연결할 때 데이터베이스 환경을 설정하는 SQL 명령입니다. 데이터 통합 서비스에서 데이터베이스에 연결할 때마다 연결 환경 SQL을 실행합니다.
트랜잭션 SQL	데이터베이스에 연결할 때 데이터베이스 환경을 설정하는 SQL 명령입니다. 데이터 통합 서비스에서 각 트랜잭션을 시작할 때 트랜잭션 환경 SQL을 실행합니다.
다시 시도 기간	이 속성은 나중에 사용하기 위해 예약되어 있습니다.
병렬 모드 설정	대량 모드에서 데이터를 테이블로 로드할 때 병렬 처리를 활성화합니다. 기본적으로 이 옵션은 비활성화되어 있습니다.
SQL 식별자 문자	데이터베이스에서 SQL 쿼리의 구분자로 분리된 식별자를 묶기 위해 사용하는 문자 유형입니다. 사용 가능한 문자는 데이터베이스 유형에 따라 다릅니다. 데이터베이스에서 일반 식별자를 사용할 경우 (없음)을 선택합니다. 데이터 통합 서비스에서 SQL 쿼리를 생성할 때 식별자 주위에 구분자로 분리된 문자를 배치하지 않습니다. 데이터베이스에서 구분자로 분리된 식별자를 사용할 경우 문자를 선택합니다. 데이터 통합 서비스가 SQL 쿼리를 생성할 때 구분자로 분리된 식별자를 이 문자로 묶습니다.
대/소문자가 혼합된 식별자 지원	데이터베이스에서 대/소문자를 구분하는 식별자를 사용할 경우 활성화합니다. 이 속성을 활성화하면 데이터 통합 서비스가 SQL 식별자 문자 속성에 대해 선택한 문자로 모든 식별자를 묶습니다. SQL 식별자 문자 속성이 없음으로 설정된 경우 대/소문자가 혼합된 식별자 지원 속성이 비활성화됩니다.

Salesforce 연결 속성

Salesforce 개체에 연결하려면 Salesforce 연결을 사용합니다. Salesforce 연결은 응용 프로그램 연결 유형입니다. Administrator 도구 또는 Developer tool에서 Salesforce 연결을 생성하고 관리할 수 있습니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 표에는 Salesforce 연결 속성이 설명되어 있습니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 128자를 초과하거나 공백을 포함하거나 다음과 같은 특수 문자를 포함할 수 없습니다. ~ ` ! \$ % ^ & * () - + = { [] } \ ; " ' < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 생성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결의 설명입니다. 설명은 765자를 초과할 수 없습니다.

속성	설명
위치	연결을 작성하려는 Informatica 도메인입니다.
유형	연결 유형입니다. 표준 연결 유형 또는 OAuth 연결 유형을 선택할 수 있습니다.
사용자 이름	표준 연결 유형에 적용됩니다. Salesforce 사용자 이름입니다.
사용자 암호	표준 연결 유형에 적용됩니다. Salesforce 사용자 이름의 암호입니다. 조직의 트러스트된 네트워크 외부에서 Salesforce에 액세스하려면 데스크톱 클라이언트 또는 API에 로그인하는 암호에 보안 토큰을 추가해야 합니다. 보안 토큰을 받거나 재설정하려면 Salesforce에 로그인하고 Setup > My Personal Information > Reset My Security Token을 클릭합니다. 암호는 대/소문자를 구분합니다.
서비스 URL	액세스하려는 Salesforce 서비스의 URL입니다. 예를 들면 다음과 같습니다. <code>https://login.salesforce.com/services/Soap/u/47.0</code> 테스트 또는 개발 환경에서 Salesforce Sandbox 테스트 환경에 액세스할 수 있습니다. Salesforce Sandbox에 대한 자세한 내용은 Salesforce 설명서를 참조하십시오.
새로 고침 토큰	OAuth 연결 유형에 적용됩니다. Salesforce의 새로 고침 토큰입니다.
소비자 키	OAuth 연결 유형에 적용됩니다. Salesforce에서 가져온 소비자 키로, 새로 고침 토큰을 생성하는 데 필요합니다. 소비자 키를 생성하는 방법에 대한 자세한 내용은 Salesforce 설명서를 참조하십시오.
소비자 암호	OAuth 연결 유형에 적용됩니다. Salesforce에서 가져온 소비자 암호로, 새로 고침 토큰을 생성하는 데 필요합니다. 소비자 암호를 생성하는 방법에 대한 자세한 내용은 Salesforce 설명서를 참조하십시오.

Salesforce Marketing Cloud 연결 속성

Salesforce Marketing Cloud 개체에 연결하려면 Salesforce Marketing Cloud 연결을 사용합니다. Administrator 도구 또는 Developer tool에서 Salesforce Marketing Cloud 연결을 생성하고 관리할 수 있습니다.

참고: 연결 속성의 순서는 속성을 보는 도구에 따라 다를 수 있습니다.

다음 테이블에는 Salesforce Marketing Cloud 연결 속성이 설명되어 있습니다.

연결 속성	설명
이름	Salesforce Marketing Cloud 연결의 이름입니다.
ID	데이터 통합 서비스에서 이 ID를 사용하여 연결을 식별합니다.
설명	선택 사항입니다. 연결의 설명입니다.
위치	연결을 생성하려는 Informatica 도메인입니다.

연결 속성	설명
유형	연결 유형입니다. Salesforce Marketing Cloud를 선택합니다.
Salesforce Marketing Cloud URL	데이터 통합 서비스에서 Salesforce Marketing Cloud WSDL에 연결하기 위해 사용하는 URL입니다. 다음 URL은 OAuth 1.0 URL에 대한 예입니다. <code>https://webservice.s7.exacttarget.com/etframework.wsdl</code> 다음 URL은 OAuth 2.0 URL에 대한 예입니다. <code>https://<SUBDOMAIN>.soap.marketingcloudapis.com/etframework.wsdl</code> Salesforce Marketing Cloud에서 OAuth 1.0에 대한 지원을 중단하기 전에 OAuth 2.0으로 업그레이드하는 것이 좋습니다.
사용자 이름	Salesforce Marketing Cloud 계정의 사용자 이름입니다.
암호	Salesforce Marketing Cloud 계정의 암호입니다.
클라이언트 ID	올바른 액세스 토큰을 생성하는 데 필요한 Salesforce Marketing Cloud의 클라이언트 ID입니다.
클라이언트 암호	올바른 액세스 토큰을 생성하는 데 필요한 Salesforce Marketing Cloud의 클라이언트 암호입니다.
로깅 활성화	로깅을 활성화하면 태스크에 대한 세션 로그를 볼 수 있습니다.
UTC 오프셋	보안 에이전트는 UTC 오프셋 연결 속성을 사용하여 UTC 오프셋 시간대로 Salesforce Marketing Cloud에서 데이터를 읽고 씁니다.
일괄 처리 크기	보안 에이전트가 대상에 일괄하여 쓸 수 있는 행의 수입니다. 데이터를 삽입 또는 업데이트하고 연락처 키를 지정하면 지정된 연락처 ID와 연결된 데이터가 Salesforce Marketing Cloud에 일괄 삽입 또는 업데이트됩니다. 데이터를 Salesforce Marketing Cloud에 Upsert할 때 연락처 키를 지정하지 마십시오.

SAP 연결 속성

SAP 테이블 또는 SAP BW 개체에 액세스하려면 SAP 연결을 사용합니다. SAP 연결은 엔터프라이즈 응용 프로그램 연결입니다. Administrator 도구 또는 개발자 도구에서 SAP 연결을 생성하고 관리할 수 있습니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 표에는 SAP 연결 속성이 설명되어 있습니다.

속성	설명
사용자 이름	필수 사항입니다. 액세스하려는 SAP 소스 시스템의 사용자 이름입니다.
암호	필수 사항입니다. 사용자 이름의 암호입니다.

속성	설명
연결 유형	<p>필수 사항입니다. 작성하려는 연결 유형입니다.</p> <p>다음 값 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - 응용 프로그램. 특정 SAP 응용 프로그램 서버에 연결하려면 응용 프로그램 연결을 작성합니다. - 부하 분산. SAP 부하 분산을 사용하려면 부하 분산 연결을 작성합니다. <p>기본값은 응용 프로그램입니다.</p> <p>선택하는 연결 유형에 따라 연결 세부 정보 대화 상자에서 해당하는 연결 속성 필드를 사용할 수 있습니다. 특정 연결 유형에 적용되지 않는 연결 속성 필드는 Developer tool에서 회색으로 표시됩니다.</p>
호스트 이름	<p>SAP 응용 프로그램 연결을 작성하는 경우 필수 항목입니다.</p> <p>연결하려는 SAP 서버의 호스트 이름 또는 IP 주소입니다.</p>
시스템 번호	<p>SAP 응용 프로그램 연결을 작성하는 경우 필수 항목입니다.</p> <p>SAP 시스템 번호입니다.</p>
메시지 호스트 이름	<p>SAP 부하 분산 연결을 작성하는 경우 필수 항목입니다.</p> <p>SAP 메시지 서버의 호스트 이름입니다.</p>
R3 이름/SysID	<p>SAP 부하 분산 연결을 작성하는 경우 필수 항목입니다.</p> <p>SAP 시스템의 이름입니다.</p>
그룹	<p>SAP 부하 분산 연결을 작성하는 경우 필수 항목입니다.</p> <p>SAP 응용 프로그램 서버의 그룹 이름입니다.</p>
클라이언트	<p>필수 사항입니다. SAP 클라이언트 번호입니다.</p>
언어	<p>선택 사항입니다. 매핑 및 워크플로우에 사용할 언어입니다.</p> <p>Developer tool 코드 페이지와 호환되어야 합니다.</p> <p>이 옵션을 비워 두면 Developer tool에서는 SAP 시스템의 기본 언어를 사용합니다.</p>
추적	<p>선택 사항입니다. SAP 시스템이 만드는 JCo 호출을 추적하려면 이 옵션을 사용합니다. SAP에서는 JCo 호출에 대한 정보를 추적 파일에 저장합니다.</p> <p>다음 값 중 하나를 지정합니다.</p> <ul style="list-style-type: none"> - 0. 꺼짐 - 1. 전체 <p>기본값은 0입니다.</p> <p>다음과 같은 디렉터리에서 추적 파일에 액세스할 수 있습니다.</p> <ul style="list-style-type: none"> - Informatica 서비스를 설치한 시스템의 <Informatica 설치 디렉터리>/tomcat/bin 디렉터리 - Developer tool을 설치한 시스템의 <Informatica 설치 디렉터리>/clients/DeveloperClient 디렉터리
추가 매개 변수	<p>선택 사항입니다. 사용하려는 다른 모든 연결 매개 변수를 입력합니다.</p> <p>다음 형식을 사용합니다.</p> <p><매개 변수 이름>=<값></p>
준비 디렉터리	<p>준비 파일이 작성될 SAP 시스템의 경로입니다.</p>
소스 디렉터리	<p>소스 파일이 들어 있는 경로입니다. 경로는 데이터 통합 서비스가 액세스할 수 있는 경로여야 합니다.</p>
FTP 사용	<p>SAP에 대한 FTP 액세스를 활성화합니다.</p>

속성	설명
FTP 사용자	FTP를 사용하는 경우 필수 항목입니다. FTP 서버에 연결하는 사용자 이름입니다.
FTP 암호	FTP를 사용하는 경우 필수 항목입니다. FTP 사용자의 암호입니다.
FTP 호스트	FTP를 사용하는 경우 필수 항목입니다. FTP 서버의 호스트 이름 또는 IP 주소입니다. 필요에 따라 1 ~ 65535(경계값 포함)의 포트 번호를 지정할 수 있습니다. FTP의 기본값은 21입니다. 다음 구문 중 하나를 사용하여 호스트 이름을 지정하십시오. - hostname:port_number - IP address:port_number 포트 번호를 지정할 때 호스트 시스템에서 FTP의 포트 번호를 활성화하십시오. SFTP를 활성화하는 경우 SFTP 서버의 호스트 이름 또는 포트 번호를 지정하십시오. SFTP의 기본값은 22입니다.
다시 시도 기간	연결이 실패할 경우에 데이터 통합 서비스가 FTP 호스트에 다시 연결하려고 시도하는 시간(초)입니다. 다시 시도 기간 내에 데이터 통합 서비스가 FTP 호스트에 다시 연결할 수 없으면 매핑 또는 워크플로우가 실패합니다. 기본값은 0입니다. 값이 0이면 무한한 다시 시도 기간을 나타냅니다.
SFTP 사용	SAP에 대한 SFTP 액세스를 활성화합니다.
공개 키 파일 이름	SFTP를 활성화하고, SFTP 서버가 공개 키 인증을 사용하는 경우 필수 항목입니다. 공개 키 파일 경로 및 파일 이름입니다.
개인 키 파일 이름	SFTP를 활성화하고, SFTP 서버가 공개 키 인증을 사용하는 경우 필수 항목입니다. 개인 키 파일 경로 및 파일 이름입니다.
개인 키 파일 이름 암호	SFTP를 활성화하고, SFTP 서버가 공개 키 인증을 사용하며 개인 키가 암호화되는 경우 필수 항목입니다. 개인 키 파일의 암호를 해독하는 암호입니다.
포트 범위	데이터 통합 서비스가 스트리밍 모드로 SAP 서버의 데이터를 읽을 때 사용해야 하는 HTTP 포트 범위입니다. 하이픈을 구분 기호로 사용하여 최대 포트 번호와 최소 포트 번호를 입력하십시오. 최소 포트 번호와 최대 포트 번호는 10000 - 65535 사이로 지정할 수 있습니다. 조직에 맞게 포트 범위를 지정할 수도 있습니다. 기본값은 10000-65535입니다.
HTTPS 사용	SAP 테이블의 데이터를 읽을 때 HTTPS 스트리밍을 활성화하려면 이 옵션을 선택합니다. 기본적으로 HTTPS 사용 확인란은 선택되지 않습니다. 스트리밍 모드에서 테이블 판독기 매핑에 대한 HTTPS를 구성하는 방법에 대한 자세한 내용은 Informatica 설명서 포털에서 "HTTPS Configuration for Table Reader Mappings in Streaming Mode for PowerExchange for SAP NetWeaver" 문서를 참조하십시오.
키 저장소 파일 경로	HTTPS를 사용하는 경우 필수 항목입니다. 개인 또는 공용 키 쌍 및 연결된 인증서를 포함하는 키 저장소 파일의 경로입니다.

속성	설명
키 저장소 암호	HTTPS를 사용하는 경우 필수 항목입니다. 키 저장소 파일의 암호입니다.
개인 키 암호	HTTPS를 사용하는 경우 필수 항목입니다. 개인 키 파일의 암호를 해독하는 암호입니다.

순차적 연결 속성

순차적 데이터 소스에 액세스하려면 순차적 연결을 사용합니다. **Developer tool**에서 순차적 연결을 생성합니다. **Administrator** 도구 또는 **Developer tool**에서 순차적 연결을 관리할 수 있습니다.

순차적 데이터 소스는 **PowerExchange**가 **SEQ** 액세스 방법으로 정의된 데이터 맵을 사용하여 액세스할 수 있는 데이터 소스입니다. 데이터 통합 서비스는 **PowerExchange**를 통해 데이터 소스에 연결합니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 표에는 순차적 연결 속성이 설명되어 있습니다.

옵션	설명
위치	순차적 데이터 집합에 연결하는 PowerExchange Listener의 위치에 대한 노드 이름입니다. 노드 이름은 PowerExchange dbmover.cfg 구성 파일에서 NODE 문의 첫 번째 매개 변수에 정의되어 있습니다.
사용자 이름	순차적 데이터 집합에 액세스할 수 있는 권한이 있는 사용자 이름입니다.
암호	<p>지정된 사용자 이름에 대한 암호 또는 유효한 PowerExchange 암호입니다.</p> <p>PowerExchange 암호는 9자에서 128자 사이일 수 있으며 다음 문자를 포함할 수 있습니다.</p> <ul style="list-style-type: none"> - 대문자 및 소문자 - 숫자 0~9 - 공백 - 다음 특수 문자: ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>참고: 첫 문자는 아포스트로피입니다.</p> <p>암호는 작은따옴표('), 큰따옴표(") 또는 통화 기호를 포함할 수 없습니다.</p> <p>암호를 사용하려면 PowerExchange 수신기가 DBMOVER 멤버에서 SECURITY=(1,N) 이상의 보안 설정으로 실행되고 있는지 확인하십시오. 자세한 내용은 <i>PowerExchange Reference Manual</i>에서 "SECURITY Statement"를 참조하십시오.</p> <p>IBM IRRPHREX 종료에서 허용 가능한 문자는 PowerExchange 암호에서 허용 가능한 문자에 영향을 미치지 않습니다.</p> <p>참고: 유효한 RACF 암호는 최대 100자일 수 있습니다. PowerExchange는 유효성 검사를 위해 암호를 RACF로 전달할 때 100자를 넘는 암호를 자릅니다.</p>
코드 페이지	필수. 순차적 데이터 집합에서 읽거나 해당 집합에 기록할 때 사용하는 코드 페이지의 이름입니다. 대개 이 값은 ISO 코드 페이지 이름입니다(예: ISO-8859-6).
통과 보안 활성화됨	연결에 대한 통과 보안을 활성화합니다.

옵션	설명
암호화 유형	<p>선택 사항입니다. 데이터 통합 서비스가 사용하는 암호화 유형입니다. 다음 옵션 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - 없음 - AES <p>기본값은 없음입니다.</p> <p>참고: Informatica에서는 암호화 유형 및 수준 연결 속성을 구성하는 대신 SSL(Secure Sockets Layer) 인증을 사용하는 것을 권장합니다. SSL 인증은 더 엄격한 보안을 제공하며 다양한 Informatica 제품에서 사용됩니다.</p> <p>PowerExchange 네트워크에서 SSL 인증을 구현하는 것에 대한 자세한 내용은 <i>PowerExchange 참조 설명서</i>를 참조하십시오.</p>
[암호화] 수준	<p>AES를 암호화 유형으로 선택하는 경우 다음 옵션 중 하나를 선택하여 데이터 통합 서비스가 사용하는 암호화 수준을 나타냅니다.</p> <ul style="list-style-type: none"> - 1. 128비트 암호화 키를 사용합니다. - 2. 192비트 암호화 키를 사용합니다. - 3. 256비트 암호화 키를 사용합니다. <p>AES를 암호화 유형으로 선택하지 않는 경우 이 옵션은 무시됩니다.</p> <p>기본값은 1입니다.</p>
간격 크기	<p>선택 사항입니다. 소스 시스템이 PowerExchange 수신기로 전달할 수 있는 데이터의 양입니다. 외부 응용 프로그램, 데이터베이스 또는 데이터 통합 서비스 노드가 병목 지점인 경우 간격 크기를 설정하십시오. 낮은 값을 사용할수록 성능이 빨라집니다.</p> <p>최소값과 기본값은 0입니다. 0 값이 최상의 성능을 제공합니다.</p>
행으로 해석	<p>(선택 사항) 간격 크기를 행 수로 표현하려면 이 옵션을 선택합니다. 간격 크기를 킬로바이트로 표현하려면 이 옵션을 선택 취소합니다. 기본적으로 이 옵션은 선택되지 않으며 간격 크기는 킬로바이트 단위입니다.</p>
압축	<p>선택 사항입니다. 소스 데이터 압축을 활성화하려면 이 옵션을 선택합니다. 데이터를 압축하면 Informatica 응용 프로그램이 네트워크를 통해 전송하는 데이터의 양을 줄일 수 있습니다. 기본적으로 이 옵션은 선택되지 않으며 압축이 비활성화됩니다.</p>
오프로드 처리	<p>선택 사항입니다. 일부 대량 데이터 처리를 소스 시스템에서 데이터 통합 서비스 시스템으로 오프로드할지 여부를 제어합니다. 다음 옵션 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - AUTO. 데이터 통합 서비스가 오프로드 처리를 사용할지 여부를 결정합니다. - 예. 오프로드 처리를 사용합니다. - 아니요. 오프로드 처리를 사용하지 않습니다. <p>기본값은 AUTO입니다.</p>
작업자 스레드	<p>선택 사항입니다. 오프로드 처리가 활성화된 경우 데이터 통합 서비스가 대량 데이터를 처리하기 위해 사용하는 스레드 수입니다. 최적의 성능을 위해서는 이 값이 데이터 통합 서비스 시스템에서 사용할 수 있는 프로세서 수를 초과하지 않아야 합니다. 올바른 값은 1~64입니다. 기본값은 다중 스레딩을 비활성화하는 0입니다.</p>

옵션	설명
배열 크기	(선택 사항) 작업자 스레드를 위한 저장소 배열의 레코드 수입입니다. 이 옵션은 작업자 스레드 옵션을 0보다 큰 값으로 설정한 경우 적용됩니다. 유효한 값은 25~5000입니다. 기본값은 25입니다.
쓰기 모드	<p>선택 사항입니다. 데이터 통합 서비스가 PowerExchange 수신기로 데이터를 전송하는 모드입니다. 다음 쓰기 모드 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - CONFIRMWRITEON. PowerExchange Listener로 데이터를 전송하고 추가 데이터를 전송하기 전에 응답을 대기합니다. 오류 복구가 중요한 경우 이 옵션을 선택합니다. 하지만 이 옵션을 선택하면 성능이 저하될 수 있습니다. - CONFIRMWRITEOFF. 응답을 기다리지 않고 PowerExchange Listener로 데이터를 전송합니다. 오류 발생 시 대상 테이블을 다시 로드할 수 있는 경우 이 옵션을 사용합니다. - ASYNCHRONOUSWITHFAULTTOLERANCE. 응답을 기다리지 않고 PowerExchange Listener로 데이터를 전송합니다. 이 옵션도 오류 검색을 활성화합니다. 이 옵션은 CONFIRMWRITEOFF의 속도와 CONFIRMWRITEON의 데이터 무결성을 결합합니다. 기본값은 CONFIRMWRITEON입니다.

Snowflake 연결 속성

Snowflake 연결을 설정할 때 연결 속성을 구성해야 합니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 테이블에는 Snowflake 연결 속성이 설명되어 있습니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 있습니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~ ` ! \$ % ^ & * () - + = { [] } \ ; " ' < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 생성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	선택 사항입니다. 연결의 설명입니다. 설명은 4,000자를 초과할 수 없습니다.
위치	연결을 작성하려는 도메인입니다.
유형	연결 유형입니다. Snowflake를 선택합니다.
사용자 이름	Snowflake 계정에 연결할 사용자 이름입니다.
암호	Snowflake 계정에 연결할 암호입니다.
계정	Snowflake 계정의 이름입니다.
웨어하우스	Snowflake 웨어하우스 이름입니다.

속성	설명
역할	사용자에게 할당된 Snowflake 역할입니다.
추가 JDBC URL 매개 변수	<p>하나 이상의 JDBC 연결 매개 변수를 다음 형식으로 입력하십시오.</p> <pre><param1>=<value>&<param2>=<value>&<param3>=<value>....</pre> <p>예를 들어</p> <pre>user=jon&warehouse=mywh&db=mydb&schema=public</pre> <p>Okta SSO 인증을 통해 Snowflake에 액세스하려면 SAML 2.0 프로토콜을 구현하는 웹 기반 IdP를 다음과 같은 형식으로 입력합니다.</p> <pre>authenticator=https://<Your_Okta_Account_Name>.okta.com</pre> <p>참고: Microsoft ADFS는 지원되지 않습니다.</p> <p>Okta 인증 구성에 대한 자세한 내용은 다음 웹 사이트를 참조하십시오. https://docs.snowflake.net/manuals/user-guide/admin-security-fed-auth-configure-snowflake.html#configuring-snowflake-to-use-federated-authentication</p>

Teradata Parallel Transporter 연결 속성

Teradata 테이블에 액세스하려면 Teradata PT 연결을 사용합니다. Teradata PT 연결은 데이터베이스 유형 연결입니다. Administrator 도구 또는 Developer tool에서 Teradata PT 연결을 생성하고 관리할 수 있습니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 테이블에는 Teradata PT 연결 속성이 설명되어 있습니다.

속성	설명
이름	<p>연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다.</p> <pre>~ ` ! \$ % ^ & * () - + = { [] } \ ; " ' < , > . ? /</pre>
ID	<p>데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.</p>
설명	연결에 대한 설명입니다. 설명은 765자를 초과할 수 없습니다.
위치	연결을 생성하려는 도메인입니다.
유형	연결 유형입니다. Teradata PT를 선택합니다.
사용자 이름	데이터베이스에 액세스하기 위한 적절한 읽기 및 쓰기 데이터베이스 사용 권한이 있는 Teradata 데이터베이스 사용자 이름입니다.
암호	Teradata 데이터베이스 사용자 이름에 대한 암호입니다.

속성	설명
드라이버 이름	Teradata JDBC 드라이버의 이름입니다.
연결 문자열	데이터베이스에서 메타데이터에 액세스하는 데 사용되는 연결 문자열입니다. 다음 연결 문자열을 사용합니다. <code>jdbc:teradata://<호스트 이름>/database=<데이터베이스 이름>,tmode=ANSI,charset=UTF8</code>

다음 테이블에는 데이터 액세스와 관련된 속성이 설명되어 있습니다.

속성	설명
TDPID	Teradata 데이터베이스 시스템의 이름 또는 IP 주소입니다.
데이터베이스 이름	Teradata 데이터베이스 이름입니다. 데이터베이스 이름을 입력하지 않으면 Teradata PT API에서 기본 로그인 데이터베이스 이름을 사용합니다.
데이터 코드 페이지	데이터베이스와 연결된 코드 페이지입니다. Teradata 대상에 데이터를 쓰는 매핑을 실행하는 경우 Teradata PT 연결의 코드 페이지가 Teradata 대상의 코드 페이지와 같아야 합니다. 기본값은 UTF-8입니다.
터네시티	Teradata 데이터베이스에서 최대 작업 수가 실행되고 있는 경우 Teradata PT API가 로그인을 계속 시도하는 기간(시)입니다. 0이 아닌 양의 정수여야 합니다. 기본값은 4입니다.
최대 세션 수	Teradata PT API가 Teradata 데이터베이스에 설정하는 최대 세션 수입니다. 0이 아닌 양의 정수여야 합니다. 기본값은 4입니다.
최소 세션 수	Teradata PT API 작업을 계속하는 데 필요한 Teradata PT API 세션의 최소 수입니다. 1과 최대 세션 수 값 사이의 양의 정수여야 합니다. 기본값은 1입니다.
중지 기간	Teradata 데이터베이스에서 최대 작업 수가 실행되고 있는 경우 Teradata PT API가 로그인을 다시 시도하기 전에 일시 중지하는 기간(분)입니다. 0이 아닌 양의 정수여야 합니다. 기본값은 6입니다.
TDCH에 메타데이터 JDBC URL 사용	TDCH(Teradata Connector for Hadoop)가 사용자가 메타데이터 액세스 속성에서 연결 문자열에 지정한 JDBC URL을 사용해야 함을 나타냅니다. 기본값은 선택됨입니다. TDCH가 매핑을 실행할 때 사용해야 하는 다른 JDBC URL을 입력하려면 이 옵션을 선택 해제합니다.
TDCH JDBC URL	TDCH가 Teradata 매핑을 실행할 때 사용해야 하는 JDBC URL을 입력합니다. 다음 형식을 사용합니다. <code>jdbc:teradata://<호스트 이름>/database=<데이터베이스 이름>,tmode=ANSI,charset=UTF8</code> 이 필드는 TDCH에 메타데이터 JDBC URL 사용 옵션을 선택 해제할 경우에만 사용할 수 있습니다.
데이터 암호화	Windows에서 SQL 요청, 응답 및 데이터에 대해 전체 보안 암호화를 활성화합니다. 기본값은 비활성화입니다.

속성	설명
추가 Sqoop 인수	<p>이 속성은 Hortonworks 또는 Cloudera 클러스터를 사용하고 Sqoop를 통해 Blaze 또는 Spark 엔진에 Teradata 매핑을 실행하는 경우 적용됩니다.</p> <p>Sqoop가 데이터를 처리할 때 사용해야 하는 인수를 입력합니다. 예를 들어 <code>--method split.by.amp</code>를 입력합니다. 여러 인수는 공백으로 구분합니다.</p> <p>지정할 수 있는 인수 목록은 Hortonworks의 Teradata Connector 및 Cloudera Connector Powered by Teradata 설명서를 참조하십시오.</p> <p>참고: Hortonworks의 Teradata 커넥터를 사용하는 경우 읽기 작업에 둘 이상의 소스 테이블을 추가하려면 <code>--split-by</code> 인수가 필요합니다. Cloudera Connector Powered by Teradata를 사용하는 경우 소스 테이블에 기본 키가 정의되어 있지 않으면 소스 연결에 <code>--split-by</code> 인수가 필요합니다.</p>
인증 유형	<p>사용자를 인증하는 방법입니다.</p> <p>다음 인증 유형 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - 원시. 연결에 지정된 Teradata 데이터베이스에 대해 사용자 이름과 암호를 인증합니다. - LDAP. 외부 LDAP 디렉터리 서비스에 대해 사용자 자격 증명을 인증합니다. <p>기본값은 원시입니다.</p>

Tableau 연결 속성

Tableau에 연결하려면 Tableau 연결을 사용합니다. Tableau 연결을 생성할 때 Tableau 액세스에 필요한 정보를 입력합니다.

다음 테이블에는 Tableau 연결 속성이 설명되어 있습니다.

속성	설명
이름	Tableau 연결의 이름입니다.
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 생성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결에 대한 설명입니다. 설명은 765자를 초과할 수 없습니다.
위치	연결을 생성하려는 Informatica 도메인입니다.
유형	연결 유형입니다. Tableau를 선택합니다.

다음 테이블에는 Tableau에 연결하는 속성이 설명되어 있습니다.

연결 속성	설명
Tableau 제품	연결하려는 Tableau 제품의 이름입니다. 다음 Tableau 제품 중 하나를 선택하여 TDE 또는 TWBX 파일을 게시할 수 있습니다. - Tableau Desktop. 데이터 통합 서비스 시스템에 TDE 파일을 생성합니다. 그런 다음 수동으로 TDE 파일을 Tableau Desktop으로 가져올 수 있습니다. - Tableau Server. 생성된 TDE 또는 TWBX 파일을 Tableau Server에 게시합니다. - Tableau Online. 생성된 TDE 또는 TWBX 파일을 Tableau Online에 게시합니다.
연결 URL	TDE 또는 TWBX 파일을 게시하려는 Tableau Server 또는 Tableau Online의 URL입니다. URL의 형식은 다음과 같습니다. <code>http://<Tableau Server 또는 Tableau Online의 호스트 이름>:<포트></code>
사용자 이름	Tableau Server 또는 Tableau Online 계정의 사용자 이름입니다.
암호	Tableau Server 또는 Tableau Online 계정의 암호입니다.
콘텐츠 URL	TDE 또는 TWBX 파일을 게시하려는 Tableau Server 또는 Tableau Online 사이트의 이름입니다. 사이트 이름을 입력하려면 Tableau 관리자에게 문의하십시오.

Tableau V3 연결 속성

Tableau V3 연결을 설정할 때 연결 속성을 구성해야 합니다.

다음 테이블에는 Tableau V3 연결 속성이 설명되어 있습니다.

속성	설명
이름	Tableau V3 연결의 이름입니다.
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 생성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결에 대한 설명입니다. 설명은 765자를 초과할 수 없습니다.
위치	연결을 작성하려는 Informatica 도메인입니다.
유형	연결 유형입니다. Tableau V3을 선택합니다.

다음 테이블에는 Tableau 연결을 위한 속성이 설명되어 있습니다.

연결 속성	설명
Tableau 제품	<p>연결하려는 Tableau 제품의 이름입니다.</p> <p>다음 Tableau 제품 중 하나를 선택하여 .hyper 또는 TWBX 파일을 게시할 수 있습니다.</p> <p>Tableau Desktop</p> <p>데이터 통합 서비스 시스템에 .hyper 파일을 생성합니다. 그러면 .hyper 파일을 Tableau Desktop으로 수동으로 가져올 수 있습니다.</p> <p>Tableau Server</p> <p>생성된 .hyper 또는 TWBX 파일을 Tableau Server에 게시합니다.</p> <p>Tableau Online</p> <p>생성된 .hyper 또는 TWBX 파일을 Tableau Online에 게시합니다.</p>
연결 URL	.hyper 또는 TWBX 파일을 게시하려는 Tableau Server 또는 Tableau Online의 URL입니다. 다음과 같은 형식의 URL을 입력합니다. http://<Tableau Server 또는 Tableau Online의 호스트 이름>:<포트>
사용자 이름	Tableau Server 또는 Tableau Online 계정의 사용자 이름입니다.
암호	Tableau Server 또는 Tableau Online 계정의 암호입니다.
사이트 ID	<p>TWBX 파일을 게시하려는 Tableau Server 또는 Tableau Online 사이트의 ID입니다.</p> <p>참고: 사이트 ID를 입력하려면 Tableau 관리자에게 문의하십시오.</p>
스키마 파일 경로	<p>데이터 통합 서비스가 가져오는 Tableau 메타데이터가 있는 샘플 .hyper 파일의 경로입니다.</p> <p>스키마 파일 경로에 대해 다음 옵션 중 하나를 입력합니다.</p> <ul style="list-style-type: none"> - .hyper 파일의 절대 경로. - .hyper 파일의 디렉터리 경로. - 빈 디렉터리 경로. <p>스키마 파일에 대해 지정하는 경로는 대상 .hyper 파일의 기본 경로가 됩니다. 파일 경로를 지정하지 않는 경우 데이터 통합 서비스는 다음과 같은 기본 파일 경로를 대상 .hyper 파일에 사용합니다.</p> <p><데이터 통합 서비스 설치 디렉터리>/apps/Data_Integration_Server/<최신 버전>/bin/rtdm</p>

Twitter 연결 속성

Twitter 웹 사이트에서 데이터를 추출하려면 Twitter 연결을 사용합니다. Twitter 연결은 소셜 미디어 연결입니다. Administrator 도구 또는 Developer tool에서 Twitter 연결을 생성하고 관리할 수 있습니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 표에는 **Twitter** 연결 속성이 설명되어 있습니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~ ` ! \$ % ^ & * () - + = { [] } \ ; " ' < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결의 설명입니다. 설명은 765자를 초과할 수 없습니다.
위치	연결을 생성하려는 도메인입니다.
유형	연결 유형입니다. Twitter를 선택합니다.
OAuth 세부 정보가 있으십니까?	OAuth를 구성할지 여부를 나타냅니다. 다음 값 중 하나를 선택합니다. - 예. 액세스 토큰 및 암호가 있음을 나타냅니다. - 아니요. OAuth 유틸리티를 시작합니다.
소비자 키	Twitter에서 응용 프로그램을 생성할 때 얻게 되는 소비자 키입니다. Twitter에서는 이 키를 사용하여 응용 프로그램을 식별합니다.
소비자 암호	Twitter 응용 프로그램을 생성할 때 얻게 되는 소비자 암호입니다. Twitter에서는 이 암호를 사용하여 소비자 키의 소유권을 설정합니다.
액세스 토큰	OAuth 유틸리티가 반환하는 액세스 토큰입니다. Twitter에서는 사용자 자격 증명 대신 이 토큰을 사용하여 보호된 리소스에 액세스합니다.
액세스 암호	OAuth 유틸리티가 반환하는 액세스 암호입니다. 이 암호는 토큰의 소유권을 설정합니다.

Twitter 스트리밍 연결 속성

Twitter 스트리밍 연결을 사용하여 **Twitter** 웹 사이트의 거의 실시간 데이터에 액세스합니다. **Twitter** 스트리밍 연결은 소셜 미디어 회사의 스트리밍 API에 대한 연결입니다. **Administrator** 도구 또는 **Developer tool**에서 **Twitter** 스트리밍 연결을 생성하고 관리할 수 있습니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 표에는 **Twitter** 스트리밍 연결의 일반 속성이 설명되어 있습니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~ ` ! \$ % ^ & * () - + = { [] } \ ; " ' < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.

속성	설명
설명	연결의 설명입니다. 설명은 765자를 초과할 수 없습니다.
위치	연결을 생성하려는 도메인입니다.
유형	연결 유형입니다. Twitter 스트리밍을 선택합니다.

다음 표에는 호스 유형 및 OAuth 인증에 대한 속성이 설명되어 있습니다.

속성	설명
호스 유형	스트리밍 API 메서드. 다음 메서드 중 하나를 지정할 수 있습니다. - 필터. Twitter 상태/필터 메서드는 검색 조건과 일치하는 공개 상태를 반환합니다. - 샘플. Twitter 상태/샘플 메서드는 모든 공개 상태에서 무작위 샘플을 반환합니다.
소비자 키	Twitter에서 응용 프로그램을 생성할 때 얻게 되는 소비자 키입니다. Twitter에서는 이 키를 사용하여 응용 프로그램을 식별합니다.
소비자 암호	Twitter 응용 프로그램을 생성할 때 얻게 되는 소비자 암호입니다. Twitter에서는 이 암호를 사용하여 소비자 키의 소유권을 설정합니다.
OAuth 세부 정보가 있으십니까?	OAuth를 구성할지 여부를 나타냅니다. 다음 값 중 하나를 선택합니다. - 예. 액세스 토큰 및 암호가 있음을 나타냅니다. - 아니요. OAuth 유틸리티를 시작합니다.
액세스 토큰	OAuth 유틸리티가 반환하는 액세스 토큰입니다. Twitter에서는 사용자 자격 증명 대신 이 토큰을 사용하여 보호된 리소스에 액세스합니다.
액세스 암호	OAuth 유틸리티가 반환하는 액세스 암호입니다. 이 암호는 토큰의 소유권을 설정합니다.

VSAM 연결 속성

VSAM 연결을 사용하여 VSAM 데이터 테이블에 액세스합니다. VSAM 연결은 플랫폼 파일 연결 유형입니다. Developer 도구에서 VSAM 연결을 작성합니다. Administrator 도구 또는 Developer 도구에서 VSAM 연결을 관리할 수 있습니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 표에는 VSAM 연결 속성이 설명되어 있습니다.

옵션	설명
위치	VSAM 데이터 집합에 연결하는 PowerExchange Listener의 위치에 대한 노드 이름입니다. 노드 이름은 PowerExchange dbmover.cfg 구성 파일에서 NODE 문의 첫 번째 매개 변수에 정의되어 있습니다.
사용자 이름	VSAM 데이터 집합에 연결할 수 있는 권한이 있는 사용자 이름입니다.

옵션	설명
암호	<p>지정된 사용자에게 대한 암호 또는 유효한 PowerExchange 암호입니다.</p> <p>PowerExchange 암호는 9자에서 128자 사이일 수 있으며 다음 문자를 포함할 수 있습니다.</p> <ul style="list-style-type: none"> - 대문자 및 소문자 - 숫자 0~9 - 공백 - 다음 특수 문자: ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>참고: 첫 문자는 아포스트로피입니다.</p> <p>암호는 작은따옴표('), 큰따옴표(") 또는 통화 기호를 포함할 수 없습니다.</p> <p>암호를 사용하려면 PowerExchange 수신기가 DBMOVER 멤버에서 SECURITY=(1,N) 이상의 보안 설정으로 실행되고 있는지 확인하십시오. 자세한 내용은 <i>PowerExchange Reference Manual</i>에서 "SECURITY Statement"를 참조하십시오.</p> <p>IBM IRRPHREX 종료에서 허용 가능한 문자는 PowerExchange 암호에서 허용 가능한 문자에 영향을 미치지 않습니다.</p> <p>참고: 유효한 RACF 암호는 최대 100자일 수 있습니다. PowerExchange는 유효성 검사를 위해 암호를 RACF로 전달할 때 100자를 넘는 암호를 자릅니다.</p>
코드 페이지	<p>필수. VSAM 데이터 집합에서 읽거나 해당 집합에 기록할 때 사용하는 코드 페이지의 이름입니다. 대개 이 값은 ISO 코드 페이지 이름입니다(예: ISO-8859-6).</p>
통과 보안 활성화됨	<p>연결에 대한 통과 보안을 활성화합니다.</p>
암호화 유형	<p>선택 사항입니다. 데이터 통합 서비스가 사용하는 암호화 유형입니다. 다음 옵션 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - 없음 - AES <p>기본값은 없음입니다.</p> <p>참고: Informatica에서는 암호화 유형 및 수준 연결 속성을 구성하는 대신 SSL(Secure Sockets Layer) 인증을 사용하는 것을 권장합니다. SSL 인증은 더 엄격한 보안을 제공하며 다양한 Informatica 제품에서 사용됩니다.</p> <p>PowerExchange 네트워크에서 SSL 인증을 구현하는 것에 대한 자세한 내용은 <i>PowerExchange 참조 설명서</i>를 참조하십시오.</p>
[암호화] 수준	<p>AES를 암호화 유형으로 선택하는 경우 다음 옵션 중 하나를 선택하여 데이터 통합 서비스가 사용하는 암호화 수준을 나타냅니다.</p> <ul style="list-style-type: none"> - 1. 128비트 암호화 키를 사용합니다. - 2. 192비트 암호화 키를 사용합니다. - 3. 256비트 암호화 키를 사용합니다. <p>AES를 암호화 유형으로 선택하지 않는 경우 이 옵션은 무시됩니다.</p> <p>기본값은 1입니다.</p>
간격 크기	<p>선택 사항입니다. 소스 시스템이 PowerExchange 수신기로 전달할 수 있는 데이터의 양입니다. 외부 응용 프로그램, 데이터베이스 또는 데이터 통합 서비스 노드가 병목 지점인 경우 간격 크기를 설정하십시오. 낮은 값을 사용할수록 성능이 빨라집니다.</p> <p>최소값과 기본값은 0입니다. 0 값이 최상의 성능을 제공합니다.</p>
행으로 해석	<p>선택 사항입니다. 간격 크기를 행 수로 표현하려면 이 옵션을 선택합니다. 간격 크기를 킬로바이트로 표현하려면 이 옵션을 선택 취소합니다. 기본적으로 이 옵션은 선택되지 않으며 간격 크기는 킬로바이트 단위입니다.</p>

옵션	설명
압축	선택 사항입니다. 소스 데이터 압축을 활성화하려면 이 옵션을 선택합니다. 데이터를 압축하면 Informatica 응용 프로그램이 네트워크를 통해 전송하는 데이터의 양을 줄일 수 있습니다. 기본적으로 이 옵션은 선택되지 않으며 압축이 비활성화됩니다.
오프로드 처리	선택 사항입니다. 일부 대량 데이터 처리를 소스 시스템에서 데이터 통합 서비스 시스템으로 오프로드할지 여부를 제어합니다. 다음 옵션 중 하나를 선택합니다. <ul style="list-style-type: none"> - AUTO. 데이터 통합 서비스가 오프로드 처리를 사용할지 여부를 결정합니다. - 예. 오프로드 처리를 사용합니다. - 아니요. 오프로드 처리를 사용하지 않습니다. 기본값은 AUTO입니다.
작업자 스레드	선택 사항입니다. 오프로드 처리가 활성화된 경우 데이터 통합 서비스가 대량 데이터를 처리하기 위해 사용하는 스레드 수입니다. 최적의 성능을 위해서는 이 값이 데이터 통합 서비스 시스템에서 사용할 수 있는 프로세서 수를 초과하지 않아야 합니다. 올바른 값은 1~64입니다. 기본값은 다중 스레딩을 비활성화하는 0입니다.
배열 크기	선택 사항입니다. 작업자 스레드를 위한 저장소 배열의 레코드 수입니다. 이 옵션은 작업자 스레드 옵션을 0보다 큰 값으로 설정한 경우 적용됩니다. 유효한 값은 25~5000입니다. 기본값은 25입니다.
쓰기 모드	선택 사항입니다. 데이터 통합 서비스가 PowerExchange 수신기로 데이터를 전송하는 모드입니다. 다음 쓰기 모드 중 하나를 선택합니다. <ul style="list-style-type: none"> - CONFIRMWRITEON. PowerExchange Listener로 데이터를 전송하고 추가 데이터를 전송하기 전에 응답을 대기합니다. 오류 복구가 중요한 경우 이 옵션을 선택합니다. 하지만 이 옵션을 선택하면 성능이 저하될 수 있습니다. - CONFIRMWRITEOFF. 응답을 기다리지 않고 PowerExchange Listener로 데이터를 전송합니다. 오류 발생 시 대상 테이블을 다시 로드할 수 있는 경우 이 옵션을 사용합니다. - ASYNCHRONOUSWITHFAULTTOLERANCE. 응답을 기다리지 않고 PowerExchange Listener로 데이터를 전송합니다. 이 옵션도 오류 검색을 활성화합니다. 이 옵션은 CONFIRMWRITEOFF의 속도와 CONFIRMWRITEON의 데이터 무결성을 결합합니다. 기본값은 CONFIRMWRITEON입니다.

웹 콘텐츠-Kapow Katalyst 연결 속성

웹 콘텐츠-Kapow Katalyst 연결을 사용하여 Kapow Katalyst의 로봇에 액세스합니다. 이 연결은 소셜 미디어 유형 연결입니다. Administrator 도구 또는 Developer tool에서 웹 콘텐츠-Kapow Katalyst 연결을 생성하고 관리할 수 있습니다.

참고: 연결 속성의 순서는 사용자가 이러한 속성을 보는 도구에 따라 다를 수 있습니다.

다음 표에는 웹 콘텐츠-Kapow Katalyst 연결 속성이 설명되어 있습니다.

속성	설명
이름	연결 이름입니다. 이름은 대/소문자를 구분하지 않으며 도메인 내에서 고유해야 합니다. 이름은 128자를 초과할 수 없고, 공백이나 다음 특수 문자를 포함할 수 없습니다. ~ ` ! \$ % ^ & * () - + = { [] } \ ; " ' < , > . ? /
ID	데이터 통합 서비스에서 연결을 식별하기 위해 사용하는 문자열입니다. ID는 대/소문자를 구분하지 않습니다. ID는 255자 이하여야 하고 도메인 내에서 고유해야 합니다. 연결을 작성한 후 이 속성을 변경할 수 없습니다. 기본값은 연결 이름입니다.
설명	연결의 설명입니다. 설명은 765자를 초과할 수 없습니다.
위치	연결을 생성하려는 Informatica 도메인입니다.
유형	연결 유형입니다. 웹 콘텐츠-Kapow Katalyst를 선택합니다.
관리 콘솔 URL	로봇이 업로드된 관리 콘솔의 URL입니다. URL은 http 또는 https로 시작해야 합니다. 예를 들어 http://localhost:50080입니다.
RQL 서비스 포트	소켓 서비스가 RQL 서비스를 수신하는 포트 번호입니다. 1에서 65535 사이의 값을 입력합니다. 기본값은 50000입니다.
사용자 이름	로컬 관리 콘솔에 액세스하는 데 필요한 사용자 이름입니다.
암호	로컬 관리 콘솔에 액세스하는 데 필요한 암호입니다.

웹 서비스 연결 속성

웹 서비스 연결을 사용하여 웹 서비스 소비자 변환을 웹 서비스에 연결합니다.

다음 테이블에는 웹 서비스 연결 속성이 설명되어 있습니다.

속성	설명
사용자 이름	웹 서비스에 연결하는 사용자 이름입니다. HTTP 인증 또는 WS-Security를 활성화한 경우 사용자 이름을 입력합니다. 웹 서비스 소비자 변환에 WS-Security 포트가 포함된 경우 변환이 입력 포트를 통해 동적 사용자 이름을 받습니다. 데이터 통합 서비스가 연결에 정의된 사용자 이름을 재정의합니다.
암호	사용자 이름의 암호입니다. HTTP 인증 또는 WS-Security를 활성화한 경우 암호를 입력합니다. 웹 서비스 소비자 변환에 WS-Security 포트가 포함된 경우 변환이 입력 포트를 통해 동적 암호를 받습니다. 데이터 통합 서비스가 연결에 정의된 암호를 재정의합니다.
URL 끝점	액세스하려는 웹 서비스의 URL입니다. 데이터 통합 서비스가 WSDL 파일에 정의된 URL을 재정의합니다. 웹 서비스 소비자 변환에 URL 끝점 포트가 포함된 경우 변환이 입력 포트를 통해 동적으로 URL을 받습니다. 데이터 통합 서비스가 연결에 정의된 URL을 재정의합니다.

속성	설명
제한 시간	데이터 통합 서비스가 연결을 닫기 전에 웹 서비스 공급자의 응답을 대기하는 시간(초)입니다. 1초에서 10,000초 사이의 제한 시간 값을 지정합니다.
HTTP 인증 유형	<p>HTTP를 통한 사용자 인증 유형입니다. 다음 값 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - 없음. 인증하지 않습니다. - 자동. 데이터 통합 서비스가 웹 서비스 공급자의 인증 유형을 선택합니다. - 기본. 웹 서비스 공급자의 도메인에 대한 사용자 이름과 암호를 제공해야 합니다. 데이터 통합 서비스가 인증을 위해 웹 서비스 공급자에게 사용자 이름과 암호를 전송합니다. - 다이제스트. 웹 서비스 공급자의 도메인에 대한 사용자 이름과 암호를 제공해야 합니다. 데이터 통합 서비스가 사용자 이름과 암호에서 암호화된 메시지 다이제스트를 생성하여 웹 서비스 공급자에게 전송합니다. 공급자가 사용자 이름과 암호에 대한 임시 값을 생성하여 도메인 컨트롤러의 Active Directory에 저장합니다. 공급자는 이 값을 메시지 다이제스트와 비교합니다. 두 값이 일치하면 웹 서비스 공급자가 사용자를 인증합니다. - NTLM. 도메인 이름, 서버 이름 또는 기본 사용자 이름과 암호를 제공해야 합니다. 웹 서비스 공급자는 사용자가 연결된 도메인을 기반으로 사용자를 인증합니다. 웹 서비스 공급자는 Windows 도메인 컨트롤러에서 사용자 이름 및 암호를 가져와 사용자가 제공한 사용자 이름 및 암호와 비교합니다. 두 값이 일치하면 웹 서비스 공급자가 사용자를 인증합니다. NTLM 인증에서는 도메인 컨트롤러의 Active Directory에 암호화된 암호를 저장하지 않습니다.
WS Security 유형	<p>사용하려는 WS-Security 유형입니다. 다음 값 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - 없음. 데이터 통합 서비스가 생성된 SOAP 요청에 웹 서비스 보안 헤더를 추가하지 않습니다. - PasswordText. 데이터 통합 서비스가 생성된 SOAP 요청에 웹 서비스 보안 헤더를 추가합니다. 암호는 일반 텍스트 형식으로 저장됩니다. - PasswordDigest. 데이터 통합 서비스가 생성된 SOAP 요청에 웹 서비스 보안 헤더를 추가합니다. 네트워크를 통한 반복 공격을 효과적으로 방지할 수 있는 다이제스트 형태로 암호를 저장합니다. 데이터 통합 서비스가 암호에 임시 값과 타임스탬프를 결합시킵니다. 데이터 통합 서비스가 암호에 SHA 해시를 적용하고 base64 인코딩 형식으로 인코딩한 후 SOAP 헤더에서 해당 인코딩된 암호를 사용합니다.
트러스트 인증서 파일	<p>웹 서비스의 SSL 인증서를 인증할 때 데이터 통합 서비스가 사용하는 트러스트된 인증서의 번들이 포함된 파일입니다. 파일 이름 및 전체 디렉터리 경로를 입력합니다.</p> <p>기본값은 <Informatica 설치 디렉터리>/services/shared/bin/ca-bundle.crt입니다.</p>
클라이언트 인증서 파일 이름	클라이언트를 인증할 때 웹 서비스가 사용하는 클라이언트 인증서입니다. 웹 서비스에서 데이터 통합 서비스를 인증해야 하는 경우 클라이언트 인증서 파일을 지정합니다.
클라이언트 인증서 암호	클라이언트 인증서에 대한 암호입니다. 웹 서비스에서 데이터 통합 서비스를 인증해야 하는 경우 클라이언트 인증서 암호를 지정합니다.
클라이언트 인증서 유형	<p>클라이언트 인증서 파일의 형식입니다. 다음 값 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> - PEM. 확장명이 .pem인 파일입니다. - DER. 확장명이 .cer 또는 .der인 파일입니다. <p>웹 서비스에서 데이터 통합 서비스를 인증해야 하는 경우 클라이언트 인증서 유형을 지정합니다.</p>
개인 키 파일 이름	클라이언트 인증서에 대한 개인 키 파일입니다. 웹 서비스에서 데이터 통합 서비스를 인증해야 하는 경우 개인 키 파일을 지정합니다.
개인 키 암호	클라이언트 인증서의 개인 키에 대한 암호입니다. 웹 서비스에서 데이터 통합 서비스를 인증해야 하는 경우 개인 키 암호를 지정합니다.
개인 키 유형	개인 키의 유형입니다. PEM이 지원되는 유형입니다.

데이터베이스 연결의 식별자 속성

대부분의 관계형 데이터베이스 연결을 작성할 때에는 데이터베이스 식별자 속성을 구성해야 합니다. 식별자 속성은 데이터 통합 서비스가 SQL 쿼리를 생성하여 데이터베이스에 액세스할 때 식별자를 구분 문자 안에 묶는 여부를 결정합니다.

데이터베이스 식별자는 데이터베이스 개체 이름입니다. 테이블, 보기, 열, 인덱스, 트리거, 프로시저, 제약 조건 및 규칙에 식별자가 있을 수 있습니다. 식별자를 사용하여 SQL 쿼리에서 개체를 참조할 수 있습니다. 데이터베이스에는 구분 문자 안에 묶여야 하는 정규 식별자 또는 구분자로 분리된 식별자가 있을 수 있습니다.

정규 식별자

정규 식별자는 식별자의 형식 규칙을 준수합니다. 정규 식별자는 SQL 쿼리에서 사용될 때 구분 문자가 필요하지 않습니다.

예를 들어 다음 SQL 문은 정규 식별자 *MYTABLE* 및 *MYCOLUMN*을 사용합니다.

```
SELECT * FROM MYTABLE
WHERE MYCOLUMN = 10
```

구분자로 분리된 식별자

구분자로 분리된 식별자는 식별자의 형식 규칙을 준수하지 않으므로 구분 문자로 묶어야 합니다.

데이터베이스는 다음 구분자로 분리된 식별자 유형을 사용할 수 있습니다.

예약된 키워드를 사용하는 식별자

식별자가 예약된 키워드를 사용하는 경우 해당 식별자를 SQL 쿼리에서 구분자 문자 안에 묶어야 합니다. 예를 들어 다음 SQL 문은 *ORDER*라는 이름의 테이블에 액세스합니다.

```
SELECT * FROM "ORDER"
WHERE MYCOLUMN = 10
```

특수 문자를 사용하는 식별자

식별자가 특수 문자를 사용하는 경우 SQL 쿼리에서 해당 식별자를 구분자 문자 안에 묶어야 합니다. 예를 들어 다음 SQL 문은 *MYTABLE\$@*라는 이름의 테이블에 액세스합니다.

```
SELECT * FROM "MYTABLE$@"
WHERE MYCOLUMN = 10
```

대/소문자 구분 식별자

기본적으로 IBM DB2, Microsoft SQL Server 및 Oracle 데이터베이스의 식별자는 대/소문자를 구분하지 않습니다. 데이터베이스 개체 이름을 대문자로 저장되지만 SQL 쿼리는 대/소문자를 사용하여 참조하도록 할 수 있습니다. 예를 들어 다음 SQL 문은 *MYTABLE*이라는 이름의 테이블에 액세스합니다.

```
SELECT * FROM mytable
SELECT * FROM MyTable
SELECT * FROM MYTABLE
```

대/소문자 구분 식별자를 사용하려면 SQL 쿼리에서 해당 식별자를 구분자 문자 안에 묶어야 합니다. 예를 들어 다음 SQL 문은 *MyTable*이라는 이름의 테이블에 액세스합니다.

```
SELECT * FROM "MyTable"
WHERE MYCOLUMN = 10
```

식별자 속성

대부분의 데이터베이스 연결을 작성할 때에는 데이터베이스 식별자 속성을 구성해야 합니다. 구성하는 식별자 속성은 데이터베이스가 정규 식별자를 사용하는지 식별자에서 키워드 또는 특수 문자를 사용하는지 또는 대/소문자가 구분된 식별자를 사용하는지 여부에 따라 달라집니다.

데이터베이스 연결에서 다음 식별자 속성을 구성합니다.

SQL 식별자 문자

데이터베이스에서 SQL 쿼리의 구분자로 분리된 식별자를 묶기 위해 사용하는 문자 유형입니다. 사용 가능한 문자는 데이터베이스 유형에 따라 다릅니다.

데이터베이스에서 일반 식별자를 사용할 경우 (없음)을 선택합니다. 데이터 통합 서비스에서 SQL 쿼리를 생성할 때 식별자 주위에 구분자로 분리된 문자를 배치하지 않습니다.

데이터베이스에서 구분자로 분리된 식별자를 사용할 경우 문자를 선택합니다. 데이터 통합 서비스가 SQL 쿼리를 생성할 때 구분자로 분리된 식별자를 이 문자로 묶습니다.

대/소문자가 혼합된 식별자 지원

데이터베이스에서 대/소문자를 구분하는 식별자를 사용할 경우 활성화합니다. 이 속성을 활성화하면 데이터 통합 서비스가 SQL 식별자 문자 속성에 대해 선택한 문자로 모든 식별자를 묶습니다.

Informatica 클라이언트 도구에서는 올바른 대/소문자를 가진 식별자를 참조해야 합니다. 예를 들어, 데이터베이스 연결을 작성할 때 올바른 대/소문자를 가진 데이터베이스 사용자 이름을 입력해야 합니다.

SQL 식별자 문자 속성이 없음으로 설정된 경우 대/소문자가 혼합된 식별자 지원 속성이 비활성화됩니다.

예제: 데이터베이스가 정규 식별자를 사용함

이 예제에서는 데이터베이스가 정규 식별자를 사용합니다. 예약된 키워드 또는 특수 문자를 포함하는 식별자가 없습니다. 데이터베이스가 대/소문자를 구분하지 않는 식별자를 사용합니다.

데이터베이스 연결에서 SQL 식별자 문자 속성을 (없음)으로 설정합니다. SQL 식별자 문자가 없음으로 설정된 경우 대/소문자가 혼합된 식별자 지원 속성이 비활성화됩니다.

데이터 통합 서비스에서 SQL 쿼리를 생성할 때 식별자 주위에 구분자로 분리된 문자를 배치하지 않습니다.

예제: 데이터베이스가 식별자에서 키워드 또는 특수 문자를 사용함

이 예제에서는 데이터베이스가 일부 식별자에서 키워드 또는 특수 문자를 사용합니다. 데이터베이스가 대/소문자를 구분하지 않는 식별자를 사용합니다.

데이터베이스 연결에서 다음과 같이 식별자 속성을 구성합니다.

1. SQL 식별자 문자 속성을 데이터베이스가 구분자로 분리된 식별자로 사용하는 문자로 설정합니다.

이 예제에서는 속성을 ""(따옴표)로 설정합니다.

2. 대/소문자가 혼합된 식별자 지원 속성을 선택 해제합니다.

데이터 통합 서비스에서 SQL 쿼리를 생성하면 서비스가 서비스가 선택한 문자를 예약된 키워드를 사용하거나 특수 문자를 사용하는 식별자 주위에 배치합니다. 예를 들어 데이터 통합 서비스가 다음 쿼리를 생성합니다.

```
SELECT * FROM "MYTABLE$@" /* identifier with special characters enclosed within delimited
character */
WHERE MYCOLUMN = 10 /* regular identifier not enclosed within delimited character */
```


예제: 데이터베이스가 대/소문자 구분 식별자를 사용함

이 예제에서는 데이터베이스가 대/소문자 구분 식별자를 사용합니다. 데이터베이스는 일부 식별자에서 키워드 또는 특수 문자를 사용할 수도 있고 사용하지 않을 수도 있습니다.

데이터베이스 연결에서 다음과 같이 식별자 속성을 구성합니다.

1. **SQL 식별자 문자** 속성을 데이터베이스가 구분자로 분리된 식별자로 사용하는 문자로 설정합니다.

이 예제에서는 속성을 `""(따옴표)`로 설정합니다.

2. **대/소문자가 혼합된 식별자 지원** 속성을 선택합니다.

데이터 통합 서비스에서 SQL 쿼리를 생성하면 서비스가 선택한 문자를 모든 식별자 주위에 배치합니다. 예를 들어 데이터 통합 서비스가 다음 쿼리를 생성합니다.

```
SELECT * FROM "MyTable" /* case-sensitive identifier enclosed within delimited character */
WHERE "MYCOLUMN" = 10 /* regular identifier enclosed within delimited character */
```

인덱스

10진수
 데이터 유형 변환 [221](#)
 많은 전체 자릿수 처리 [187](#), [191](#)
 전체 자릿수 손실 [192](#)
10진수 데이터 유형
 개요 [191](#)

A

Adabas 연결
 속성 [230](#)
Amazon Redshift 연결
 속성 [232](#)
Amazon S3 연결
 속성 [234](#)
application patch
 dependency
 direct [149](#)
 indirect [149](#)
 remote [149](#)
 inherit [149](#)
Array
 복합 데이터 유형 [194](#)
 1차원 [194](#)
 다중 차원 [194](#)
 예 [194](#)
 인덱스 [194](#)
 차원 [194](#)
 형식 [194](#)
Avro 데이터 유형
 변환 데이터 유형 [197](#)

B

bigint
 계산에서 사용 [187](#)
 많은 전체 자릿수 처리 [187](#)
 식의 상수 [188](#)
 플랫 파일에 쓰기 [189](#)
Blaze 엔진
 연결 속성 [247](#)
편집
 응용 프로그램 [146](#)
 응용 프로그램 개체 [146](#)
폴더
 설명 [28](#)
 작성 [28](#)
프로젝트
 공유 [24](#)
 사용 권한 [25](#)
 사용 권한 할당 [27](#)
 설명 [24](#)
 작성 [25](#)

프로젝트 (계속)
 필터링 [25](#)
프로젝트 사용 권한
 보안 세부 정보 표시 [27](#)
 부여 권한 [25](#)
 상위 개체 액세스 허용 [27](#)
 쓰기 권한 [25](#)
 외부 개체 사용 권한 [26](#)
 읽기 권한 [25](#)
 종속 개체 인스턴스 [26](#)
 할당 [27](#)
플랫 파일 데이터 개체
 Excel에서 메타데이터 구성 [99](#)
 메타데이터 복사 [100](#)
 Excel에서 편집 [100](#)
 고급 속성 [89](#)
 고정 너비 속성 [91](#)
 구분자로 분리된 속성 [90](#)
 기존 플랫 파일에서 작성 [102](#)
 런타임 쓰기 속성 [94](#)
 런타임 읽기 속성 [92](#)
 비어 있는 상태로 작성 [101](#)
 설명 [87](#)
 열 속성 [88](#)
 일반 속성 [88](#)
 제어 파일에서 작성 [103](#)
 형식 속성 [90](#)
형식 속성
 플랫 파일 데이터 개체 [90](#)
환경 SQL
 구성 [54](#)
 입력 지침 [55](#)
힌트
 쿼리 보기 [70](#)

C

Cassandra 연결
 속성 [237](#)
CI/CD
 infacmd [170](#), [174](#)
 REST API [170](#), [174](#)
 개발자 협업 [171](#)
 개요 [169](#)
 개체 배포 [170](#), [172](#), [174](#)
 개체 테스트 [173](#)
 개체 할당 [171](#), [173](#)
 응용 프로그램 배포 [177](#)
 응용 프로그램 재배포 [177](#)
 지속적인 전달 [171](#)
 지속적인 전달 예 [174](#)
 지속적인 통합 [170](#)
 지속적인 통합 예 [172](#)
 충돌 검색 [171](#)

CI/CD (계속)

충돌 해결 [171](#)

팀 기반 개발 [171](#)

Cosmos DB 연결

작성 [279](#)

D

Databricks 연결 속성 [239](#)

DataSift 연결

속성 [238](#)

DDL 생성

DDL 생성 및 실행 [65](#)

DDL 생성 오류 [67](#)

Developer 도구

작업 공간 디렉터리 [18](#)

E

Excel

플랫 파일 구성 [99](#)

플랫 파일 데이터 개체 편집 [100](#)

플랫 파일에 복사 [100](#)

F

Facebook 연결

속성 [241](#)

G

Google Analytics 연결

속성 [243](#)

Google BigQuery 연결

속성 [244](#)

연결 모드 [245](#)

Google Cloud Spanner 연결

속성 [245](#)

Google Cloud Storage 연결

속성 [246](#)

Greenplum 연결

속성 [242](#)

H

HBase 연결

MapR-DB 속성 [255](#)

속성 [253](#)

HDFS 연결

속성 [253](#)

Hive 연결

속성 [255](#)

Hive 푸시다운

연결 속성 [247](#)

HTTP 연결

속성 [259](#)

I

IBM DB2 for i5/OS 연결

속성 [263](#)

IBM DB2 for z/OS 연결

속성 [266](#)

IBM DB2 연결

속성 [260](#)

IMS 연결

속성 [269](#)

Informatica Data Services

개요 [17](#)

Informatica Developer

개요 [15](#)

검색 [41](#)

설정 [22](#)

시작 [18](#)

Informatica Marketplace

설명 [22](#)

J

JD Edwards EnterpriseOne 연결

속성 [275](#)

JDBC V2 연결

속성 [273](#)

JDBC 연결

속성 [271](#)

JSON 데이터 유형

변환 데이터 유형 [199](#)

L

LDAP 연결

속성 [276](#)

LinkedIn 연결

속성 [277](#)

M

map

복합 데이터 유형 [195](#)

예 [195](#)

키-값 쌍 [195](#)

형식 [195](#)

Microsoft Azure Data Lake Storage Gen1 연결

속성 [280](#)

Microsoft Azure Data Lake Storage Gen2 연결

속성 [281](#)

Microsoft Azure SQL 데이터 웨어하우스 연결

속성 [282](#)

MS SQL Server 연결

속성 [283](#)

N

NaN

설명됨 [188](#)

Netezza 연결

속성 [287](#)

날짜/시간 데이터 유형

개요 [189](#)

내보내기

XML 파일 [180](#)

개요 [178](#)

개체 [180](#)

응용 프로그램 보관 파일에 [143](#), [144](#)

내보내기 (계속)

종속 개체 [179](#)
논리적 데이터 개체
설명 [114](#)
속성 [114](#)
예 [106](#)
작성 [116](#)
특성 관계 [115](#)
논리적 데이터 개체 매핑
쓰기 매핑 [119](#)
유형 [118](#)
읽기 매핑 [119](#)
작성 [119](#)
논리적 데이터 개체 모델
가져오기 [108](#)
생성 [107](#)
설명 [107](#)
예 [106](#)
데이터 뷰어
구성 [122](#), [125](#)
구성 문제 해결 [128](#)
구성 속성 [122](#)
구성 작성 [126](#)
데이터 유형
DB2 for i5/OS [206](#)
10진수 [191](#)
Bigint [187](#)
DB2 for z/OS [206](#)
IBM DB2 [205](#)
JDBC [207](#)
Microsoft SQL Server [209](#)
ODBC [213](#)
Oracle [215](#)
SAP HANA [217](#)
XML [219](#)
개요 [185](#)
날짜/시간 [189](#)
문자열 [193](#)
배정밀도 [191](#)
변환 [186](#)
복합 파일 [197](#)
비관계형 [211](#)
암시적 변환 [221](#)
이진 [189](#)
정수 [187](#)
포트 간 데이터 변환 [221](#)
데이터 유형 참조
플랫 파일 [204](#)
데이터 통합 서비스
선택 [23](#)
데이터베이스
환경 [SQL 54](#)
데이터베이스 연결
식별자 속성 [311](#)
데이터베이스 힌트
Developer 도구에서 입력 [70](#)
데이터의 논리적 보기
개발 [106](#)
개요 [105](#)
도메인
개체 및 속성 검색 [42](#)
검색 [41](#)
설명 [24](#)
추가 [22](#)
동기화
사용자 지정된 데이터 개체 [84](#)
실제 데이터 개체 [84](#)

런타임 쓰기 속성
플랫 파일 데이터 개체 [94](#)
런타임 응용 프로그램
개체 및 속성 검색 [42](#)
런타임 읽기 속성
플랫 파일 데이터 개체 [92](#)
로그
설명 [130](#)
로컬 작업 공간 디렉터리
구성 [18](#)
리소스 매개 변수
매핑에 배포 [139](#)
많은 전체 자릿수
10진수 데이터 유형 [187](#)
Bigint 데이터 유형 [187](#)
매핑
구성 [122](#), [126](#)
구성 문제 해결 [128](#)
구성 속성 [122](#)
배포 속성 [135](#)
생성 구성 [127](#)
메타데이터 액세스 서비스
선택 [23](#)
모니터링
설명 [132](#)
모델 리포지토리
개체 [33](#)
개체 및 속성 검색 [42](#)
개체 체크 아웃 및 체크 인 [39](#)
버전 지정 개체 [37](#)
버전이 없음 [36](#)
버전이 지정됨 [36](#)
설명 [32](#)
연결 [40](#)
추가 [23](#)
템 기반 개발 [36](#)
모델 리포지토리 서비스
새로 고침 [40](#)
문자열 데이터 유형
개요 [193](#)
문제 해결
버전 관리 [40](#)
방법
응용 프로그램 작성, 배포 및 업데이트 [141](#)
배정밀도
많은 전체 자릿수 처리 [191](#)
배정밀도 데이터 유형
개요 [191](#)
배포
개요 [133](#)
데이터 통합 서비스에 [143](#), [144](#)
매핑 속성 [135](#)
응용 프로그램 대체 [140](#)
응용 프로그램 보관 파일에 [143](#), [144](#)
응용 프로그램 업데이트 [140](#)
버전 관리
문제 해결 [40](#)
변환 데이터 유형
목록 [186](#)
보기
설명 [19](#)
연결 탐색기 보기 [48](#)
복사
개체 [29](#)
설명 [29](#)
복합 데이터 유형
Array [194](#)

복합 데이터 유형 (계속)

map [195](#)
struct [196](#)

개요 [193](#)

복합 파일 형식

개요 [197](#)

블록체인 연결 속성 [236](#)

비관계형 데이터 개체

가져오기 [80](#)

설명 [79](#)

비관계형 데이터 작업

읽기, 쓰기 및 조회 변환 작성 [80](#)

비즈니스 용어

바로 가기 키 사용자 지정 [43](#)

조회 [43](#)

사용자 정의 조인

Informatica 조인 구문 [74](#)

사용자 지정된 데이터 개체 [73](#)

오른쪽 외부 조인 구문 [78](#)

외부 조인 지원 [74](#)

왼쪽 외부 조인 구문 [76](#)

일반 조인 구문 [75](#)

입력 [73](#)

사용자 지정 SQL 쿼리

사용자 지정된 데이터 개체 [68](#)

작성 [79](#)

사용자 지정 쿼리

Informatica 조인 구문 [74](#)

오른쪽 외부 조인 구문 [78](#)

외부 조인 지원 [74](#)

왼쪽 외부 조인 구문 [76](#)

일반 조인 구문 [75](#)

사용자 지정된 데이터 개체

고급 쿼리 [68](#)

고유 항목 선택 [71](#)

고유 항목 선택 사용 [71](#)

관계형 데이터 개체 추가 [64](#)

관계형 리소스 추가 [64](#)

기본 쿼리 [68](#)

단순 쿼리 [68](#)

문제 해결 [86](#)

사용자 정의 조인 [73](#)

사용자 정의 조인 입력 [73](#)

사용자 지정 SQL 쿼리 [68](#)

사용자 지정 쿼리 작성 [79](#)

사전 및 사후 매핑 SQL 명령 [78](#)

사전 및 사후 매핑 SQL 명령 추가 [79](#)

설명 [60](#)

소스 필터 입력 [71](#)

쓰기 속성 [62](#)

예약어 파일 [68](#)

작성 [63](#)

정렬된 포트 [72](#)

정렬된 포트 사용 [72](#)

키 관계 [61](#)

키 관계 작성 [65](#)

키 작성 [64](#)

사전 및 사후 매핑 SQL 명령

관계형 데이터 개체에 추가 [79](#)

사용자 지정된 데이터 개체 [78](#)

삭제 [158](#)

생성

Cosmos DB 연결 [279](#)

성능 조정

데이터 뷰어 구성 [125](#)

데이터 뷰어 구성 작성 [126](#)

매핑 구성 [126](#)

성능 조정 (계속)

매핑 구성 작성 [127](#)

웹 서비스 구성 [127](#)

셀프 조인

사용자 지정 SQL 쿼리 [68](#)

소스 필터

입력 [71](#)

순차적 연결

속성 [297](#)

시작 페이지

설명 [21](#)

식별 관계

설명 [115](#)

식별되지 않는 관계

설명 [115](#)

식별자

구분자로 분리됨 [311](#)

정규 [311](#)

실제 데이터 개체

관계형 데이터 개체 [57](#)

동기화 [84](#)

문제 해결 [86](#)

비관계형 데이터 개체 [79](#)

사용자 지정된 데이터 개체 [60](#)

설명 [56](#)

플랫 파일 데이터 개체 [87](#)

연결

속성 [247](#), [278](#)

Salesforce Marketing Cloud [293](#)

개요 [46](#)

데이터베이스 식별자 속성 [311](#)

삭제 [46](#)

세부 정보 [278](#)

속성 [247](#), [278](#)

연결 탐색기 보기 [48](#)

웹 서비스 속성 [309](#)

이름 바꾸기 [46](#)

작성 [49](#)

전환 [46](#)

편집 [46](#)

연결 속성

Databricks [239](#)

블록체인 [236](#)

연결 전환

구성 [52](#)

데이터 유형 매핑 [53](#)

선행 조건 [51](#)

설명 [51](#)

조회 동작 [53](#)

후행 조건 [53](#)

연결 환경 SQL

구성 [54](#), [55](#)

예

Array [194](#)

Map [195](#)

struct [196](#)

예약어 파일

사용자 지정된 데이터 개체 [68](#)

작성 [69](#)

오류 메시지

그룹화 [131](#)

제한 [131](#)

외부 조인 지원

사용자 지정된 데이터 개체 [74](#)

웹 서비스

구성 [127](#)

구성 속성 [122](#)

- 웹 서비스 (계속)
 - 구성 작성 [127](#)
- 웹 연결
 - 속성 [259](#)
- 웹 콘텐츠-Kapow Katalyst 연결
 - 속성 [308](#)
- 유효성 검사
 - 기본 설정 구성 [131](#)
 - 오류 메시지 그룹화 [131](#)
 - 오류 메시지 제한 [131](#)
- 응용 프로그램
 - 개요 [133](#)
 - 개체 [140](#)
 - 데이터 통합 서비스에 배포 [137](#)
 - 런타임 개체 보기 [140](#)
 - 런타임 응용 프로그램에 연결 [146](#)
 - 매핑 배포 속성 [135](#)
 - 속성 [135](#)
 - 업데이트 [140](#)
 - 응용 프로그램 유효성 검사 [134](#)
 - 응용 프로그램 작성 [134](#)
 - 작성 [134](#), [142](#)
 - 작성, 배포 및 업데이트 방법 [141](#)
 - 재배포 [146](#)
 - 전체 [134](#)
 - 중분 [134](#), [148](#)
 - 파일로 내보내기 [138](#)
 - 파일에 배포 [138](#)
- 응용 프로그램 보관 파일 [138](#)
- 응용 프로그램 업데이트
 - 개체 [161](#)
- 응용 프로그램 패치
 - 개체 영향 [156](#), [157](#), [159](#)
 - 관련 개체 [153](#)
 - 규칙 [162](#)
 - 디자인 타임 [149](#)
 - 런타임 [149](#)
 - 배포 [160](#), [161](#)
 - 배포 전략 [160](#)
 - 범위 [159](#)
 - 삭제 [158](#)
 - 상속 [153](#)
 - 생성 [161](#)
 - 선택한 개체 [153](#)
 - 수정 [157](#)
 - 숫자 [159](#)
 - 영향 [156](#), [159](#)
 - 영향받음 [157](#)
 - 영향을 받는 개체
 - 찾기 [158](#)
 - 유형 [153](#)
 - 지침 [162](#)
 - 직접 및 간접 종속성 [155](#)
 - 직접 종속성 [154](#)
 - 직접, 간접 및 원격 종속성 [155](#)
 - 첫 번째 수준의 관련 개체 [153](#)
 - 추가 [157](#)
- 이진 데이터 유형
 - 개요 [189](#)
- 인증서
 - 인증서 관리 [83](#)
 - 인증서 속성 [84](#)
 - 트러스트되지 않은 인증서 [83](#)
 - 트러스트되지 않은 인증서 추가 [84](#)
- 작업 공간 디렉터리
 - 구성 [18](#)

- 작업 영역
 - 설명 [19](#)
- 전체 자릿수 손실
 - 작은 전체 자릿수 사용 [192](#)
- 정규 식별자
 - 데이터베이스 연결 [311](#)
- 정렬된 포트
 - 사용자 지정된 데이터 개체 [72](#)
 - 사용자 지정된 데이터 개체에서 사용 [72](#)
- 정수
 - 계산에서 사용 [187](#)
 - 식의 상수 [188](#)
 - 플랫 파일에 쓰기 [189](#)
- 조인 구문
 - Informatica 구문 [74](#)
 - 사용자 지정된 데이터 개체 [74](#)
 - 오른쪽 외부 조인 구문 [78](#)
 - 왼쪽 외부 조인 구문 [76](#)
 - 일반 조인 구문 [75](#)
- 조회
 - Business Glossary Desktop [43](#)
- 중간 계산
 - 10진수 사용 [192](#)
- 중분 응용 프로그램
 - 규칙 [162](#)
 - 업데이트 [161](#)
 - 지침 [162](#)
 - 패치 [148](#)
- 치트 시트
 - 설명 [21](#)
- 쿼리 보기
 - 힌트 구성 [70](#)
- 키 관계
 - 관계형 데이터 개체 [59](#)
 - 관계형 데이터 개체 간에 작성 [60](#)
 - 사용자 지정된 데이터 개체 [61](#)
 - 사용자 지정된 데이터 개체에서 작성 [65](#)
- 트랜잭션 환경 SQL
 - 구성 [54](#), [55](#)
- 특성
 - 관계 [115](#)
- 팀 기반 개발
 - 문제 해결 [40](#)

O

- OData 연결
 - 속성 [287](#)
- ODBC 연결
 - 속성 [288](#)
- Oracle 연결
 - 속성 [290](#)
- ORC 파일 데이터 유형
 - 변환 데이터 유형 [200](#)

P

- Parquet 데이터 유형
 - 변환 데이터 유형 [201](#)

Q

- QNaN
 - 1.#QNaN으로 변환 [188](#)

S

Salesforce Marketing Cloud
연결 속성 [293](#)
SAP 연결
속성 [294](#)
Snowflake 연결
속성 [299](#)
Spark HDFS 준비 디렉터리
Hadoop 연결 속성 [247](#)
Spark 배포 모드
Hadoop 연결 속성 [247](#)
Spark 실행 매개 변수
Hadoop 연결 속성 [247](#)
Spark 엔진
연결 속성 [247](#)
Spark 이벤트 로그 디렉터리
Hadoop 연결 속성 [247](#)
SQL
환경 SQL 구성 [54](#)
환경 SQL 입력 지침 [55](#)
SQL 힌트
Developer 도구에서 입력 [70](#)
struct
복합 데이터 유형 [196](#)
schema [196](#)
예 [196](#)
이름-유형 쌍 [196](#)
형식 [196](#)

T

Tableau V3 연결
속성 [303](#)
Teradata Parallel Transporter 연결
속성 [300](#)
Twitter 스트리밍 연결
속성 [305](#)
Twitter 연결
속성 [304](#)

V

VSAM 연결
속성 [306](#)

W

WSDL 데이터 개체
가져오기 [81](#)
개요 보기 [81](#)
고급 보기 [82](#)
동기화 [82](#)
스키마 보기 [81](#)
작성 [82](#)

ㄱ

가져오기
XML 파일 [180](#)
개요 [178](#)
개체 [183](#)
응용 프로그램 보관 [145](#)
종속 개체 [179](#)
개체
버전 기록 [39](#)
복사 [29](#)
잠긴 개체 [35](#)
개체 종속성
개체 종속성 보기 [129](#)
개체 종속성 보기
개체 종속성 [129](#)
개체 종속성 필터링 [130](#)
데이터 보기 [129](#)
개체 종속성 필터링
개체 종속성 [130](#)
개체 체크 아웃 및 체크 인 [39](#)
개체 체크 인/체크 아웃 [39](#)
검색
개요 [41](#)
도메인 [41](#)
런타임 응용 프로그램 [41](#)
모델 리포지토리 [41](#)
비즈니스 용어집 [43](#)
편집기 [44](#)
고유 항목 선택
사용자 지정된 데이터 개체 [71](#)
사용자 지정된 데이터 개체에서 사용 [71](#)
고정 너비 속성
플랫 파일 데이터 개체 [91](#)
관계형 데이터 개체
가져오기 [58](#)
문제 해결 [86](#)
사용자 지정된 데이터 개체에 추가 [64](#)
설명 [57](#)
와일드카드 검색 [58](#)
키 관계 [59](#)
키 관계 작성 [60](#)
키 작성 [59](#)
관계형 연결
사용자 지정된 데이터 개체에 추가 [64](#)
구분자로 분리된 속성
플랫 파일 데이터 개체 [90](#)
구분자로 분리된 식별자
데이터베이스 연결 [311](#)
구성
문제 해결 [128](#)
기본 SQL 쿼리
보기 [79](#)
기본 키
관계형 데이터 개체에서 작성 [59](#)
사용자 지정된 데이터 개체에서 작성 [64](#)