

HP Asset Manager

소프트웨어 버전: 9.30

마이그레이션

문서 릴리스 날짜: 2011년 3월 31일
소프트웨어 릴리스 날짜: 2011년 3월



법적 고지

저작권 고지

(c) Copyright 1994-2011 Hewlett-Packard Development Company, L.P.

제한된 권리 범례

기밀 컴퓨터 소프트웨어.

소유, 사용 또는 복사하기 위해서는 HP로부터 유효한 라이선스를 확보해야 합니다.

FAR 12.211 및 12.212에 의거하여 상용 컴퓨터 소프트웨어, 컴퓨터 소프트웨어 문서 및 상용 품목에 대한 기술 데이터는 공급업체의 표준 상용 라이선스 아래에서 미국 정부에 사용이 허가되었습니다.

보증

HP 제품 및 서비스에 대한 모든 보증 사항은 해당 제품 및 서비스와 함께 제공된 익스프레스 보증서에 규정되어 있습니다. 여기에 수록된 어떤 내용도 추가 보증을 구성하는 것으로 해석될 수 없습니다.

HP는 여기에 수록된 기술적 또는 편집상의 오류나 누락에 대해 책임지지 않습니다.

본 정보는 예고 없이 변경될 수 있습니다.

타사 및/또는 개방형 소스 라이선스 계약에 대한 자세한 내용이나 개방형 소스 코드를 보려면 다음 방법 중 하나를 사용하십시오.

- 소프트웨어 설치 CD-ROM의 ThirdParty 디렉터리
- 타사의 이진 파일 및/또는 개방형 소스 도구
- **개방형 소스 및 타사 소프트웨어 라이선스 계약 가이드**에 나오는 구성 요소의 URL

상표 고지

- Adobe®, Adobe logo®, Acrobat® 및 Acrobat Logo®는 Adobe Systems Inc.의 상표입니다.
- Corel® 및 Corel logo®는 Corel Corporation 또는 Corel Corporation Ltd.의 상표 또는 등록 상표입니다.
- Java는 Oracle 및/또는 계열사의 등록 상표입니다.
- Microsoft®, Windows®, Windows NT®, Windows® XP, Windows Mobile® 및 Windows Vista®는 Microsoft Corporation의 미국 등록 상표입니다.
- Oracle®은 Oracle Corporation 및/또는 계열사의 등록 상표입니다.
- UNIX®는 Open Group의 등록 상표입니다.

알림

이 제품은 Apache Software Foundation(Copyright © The Apache Software Foundation, 웹 주소 <http://www.apache.org/> [<http://www.apache.org/>])에서 개발한 소프트웨어를 포함합니다. All rights reserved.

이 제품은 OpenLDAP Foundation(Copyright ©, The OpenLDAP Foundation, Redwood City, California, USA)에서 개발한 소프트웨어를 포함합니다. All Rights Reserved. OpenLDAP®는 OpenLDAP Foundation의 등록 상표입니다.

이 제품은 OpenSSL Project(Copyright © The OpenSSL Project, 웹 주소 <http://www.openssl.org/>)에서 OpenSSL 툴킷에 사용하기 위해 개발한 소프트웨어를 포함합니다. All rights reserved.

이 제품은 OpenSymphony Group(Copyright © The OpenSymphony Group, 웹 주소 <http://www.opensymphony.com/>)에서 개발한 소프트웨어를 포함합니다. All rights reserved.

이 제품은 RSA Data Security로부터 사용 허가를 받은 코드를 포함합니다.

이 제품은 JDOM Project (Copyright © Jason Hunter & Brett McLaughlin, 웹 주소 <http://www.jdom.org/>)에서 개발한 소프트웨어를 포함합니다. All rights reserved.

목차

소개	11
마이그레이션을 수행해야 하는 이유	12
마이그레이션에 포함되는 작업은 무엇입니까?	12
마이그레이션 수행자	12
필수 역량	13
이 가이드를 사용하는 방법	13
1. 지원되는 환경	15
운영 체제 및 DBMS	15
DBMS 서버에 필요한 디스크 공간	15
2. 마이그레이션 프로세스	17
HP Connect-It 사용 시 주의 사항	17
마이그레이션에 필요한 작업은 무엇입니까?	17
변환에 필요한 작업은 무엇입니까?	18
도구를 통해 변환되는 항목은 무엇입니까?	18
Asset Manager 버전 4.2.x, 4.3.x, 4.4.x 또는 5.0x에서 마이그레이션하기 위한 단순 프로세스	18
변환 도구는 어떻게 작동합니까?	21
변환 프로세스가 이전 버전과 어떻게 다른지입니까?	21
버전 9.30에서 DBMS가 지원되지 않는 데이터베이스 마이그레이션	22
조달 모듈의 제한 사항	22

마이그레이션의 복잡성	22
3. 단계별 마이그레이션 - 준비 단계(프로덕션 데이터베이스)	23
예비 분석	23
마이그레이션 프로젝트 시작	24
사용자 및 지원 기술자 교육	25
변환 컴퓨터 준비	26
DBMS 서버 준비	28
4. 단계별 마이그레이션 - 시뮬레이션(시뮬레이션 데이터베이스)	31
1 1단계 - 이전 형식 프로덕션 데이터베이스의 무결성 검증	31
2 2단계 - 이전 형식 프로덕션 데이터베이스의 수동 조정	33
3 3단계 - 이전 형식 프로덕션 데이터베이스에 대한 구조 변경 전파	37
4 4단계 - 이전 형식 프로덕션 데이터베이스 복사	41
5 5단계 - 이전 형식 시뮬레이션 데이터베이스 변환	43
6 6단계 - 9.30 형식 시뮬레이션 데이터베이스 의 무결성 검증	51
7 7단계 - 9.30 형식 시뮬레이션 데이터베이스 확인	52
8 8단계 - 이전 형식 프로덕션 데이터베이스에 대한 특정 권한 제한	52
9 9단계 - 수동으로 변환할 응용 프로그램 데이터 내보내기	53
수동으로 변환할 응용 프로그램 데이터 처리	56
5. 단계별 마이그레이션 - 최종 변환(마이그레이션 데이터베이스)	71
16 16단계 - 이전 형식 프로덕션 데이터베이스 차단 및 복사	71
20 20단계 - 9.30 형식 마이그레이션 데이터베이스 완료	72
21 21단계 - Asset Manager 데이터베이스에 액세스하는 외부 소프트웨어 구성 요소 업그레이드	81
6. 단계별 마이그레이션 - 최종 단계	83
22 22단계 - Asset Manager 프로그램 업그레이드	83
23 23단계 - 9.30 형식 마이그레이션 데이터베이스 를 프로덕션에 배치	85
24 24단계 - 더 이상 사용되지 않는 프로그램 설치 제거	85

7. 용어집	87
마이그레이션	87
Asset Manager 프로그램 업데이트	87
이전 형식 프로덕션 데이터베이스 변환	88
변환 파일	88
변환 시스템	88
프로덕션 데이터베이스	88
트리거	89
데이터	89
응용 프로그램 데이터	89
데이터베이스 구조	89
8. 참조	91
migration.xml 변환 파일 조정	91
이전 버전과 비교되는 표준 데이터베이스에 대한 구조적 수정	106
수동으로 변환할 응용 프로그램 데이터	108
전파된 이전 형식 프로덕션 데이터베이스 의 구조적 매개 변수	111
기타 설명서	112
색인	115

그림 목록

2.1. 변환 - Asset Manager 버전 4.2.x, 4.3.x 또는 4.4.x에서 마이그레이션하기 위한 단순 프로세스	20
4.1. 구조 변경 전파 - 프로세스	38
4.2. 샘플 데이터 처리 - 절차	55
4.3. Asset Manager Script Analyzer - *.xml 파일 분석 창	62
4.4. Asset Manager Script Analyzer - 스크립트 분석 창	64

표 목 록

1. 이전 버전 번호별로 수행할 운영	12
4.1. ^ 문자를 사용할 수 없는 필드 - 목록	35
4.2. / 문자를 사용할 수 없는 필드 - 목록	37
4.3. Asset Manager Script Analyzer - 메뉴	60
8.1. 수동으로 변환할 응용 프로그램 데이터 - 목록	109
8.2. 이전 형식 프로덕션 데이터베이스 의 구조적 매개 변수 - 목록	112
8.3. 기타 설명서 - 목록	112

소개

Asset Manager의 이전 버전에서 버전 9.30으로 업그레이드하기 위해 수행할 운영은 이전 버전 번호에 따라 달라집니다.

이 운영이 간단한 경우 **간단한 업그레이드**라고 하며, **설치 및 업그레이드** 가이드의 **이전 버전 업그레이드** 장에서 설명합니다.

 [중요]:

간단한 업그레이드에 해당하는 경우에는 **마이그레이션** 가이드를 참조할 필요가 없습니다.

다소 복잡한 경우를 **마이그레이션**(이전 버전 번호에 따라 전체 또는 간소화된 마이그레이션)이라고 하며, 이 가이드에서 설명합니다.

다음 표를 사용하면 어느 경우에 해당하는지 알 수 있습니다.

1. 표 이전 버전 번호별로 수행할 운영

업그레이드할 버전 번호	수행할 운영	관련 설명서
버전 4.2.x, 4.3.x, 4.4.x 또는 5.0x	표준 환경에서는 간단한 업그레이드 로 충분합니다.	설치 및 업그레이드 가이드의 이전 버전 업그레이드 장, Asset Manager 버전 4.2.x, 4.3.x, 4.4.x 또는 5.0x 업그레이드 - 개요 섹션
	간단한 업그레이드가 실패할 경우 단순화된 마이그레이션 을 수행해야 합니다.	이 가이드, Asset Manager 버전 4.2.x, 4.3.x, 4.4.x 또는 5.0x에서 마이그레이션하기 위한 단순 프로세스 [18 페이지] 섹션

마이그레이션을 수행해야 하는 이유

Asset Manager 버전 4는 새로운 구조적 수정을 비롯하여 상당히 많은 내용이 변경되었습니다.

- 표준 데이터베이스 구조(테이블, 필드, 링크 및 인덱스)가 대폭 수정되었습니다.
- 새 기능이 추가되었습니다.

이러한 모든 변경으로 인해 규칙적으로 **Asset Manager**의 이전 버전을 버전 **9.30**으로 마이그레이션해야 합니다.

마이그레이션에 포함되는 작업은 무엇입니까?

마이그레이션에서는 다음과 같은 작업을 수행합니다.

- 이전 형식 프로덕션 데이터베이스에서 **9.30** 형식(구조 및 내용)으로 변환
- **Asset Manager** 프로그램을 버전 **9.30**으로 업그레이드

마이그레이션 수행자

마이그레이션은 다음 작업을 담당하는 엔지니어에 의해 수행됩니다.

- **Asset Manager** 데이터베이스 관리
- **Asset Manager** 설치
- **Asset Manager** 배포

필수 역량

마이그레이션은 다음과 같은 역량이 필요한 복잡한 프로세스입니다.

- **Asset Manager**의 이전 버전과 버전 **9.30**에 대한 전반적인 이해(설치, 매개 변수 구성, 데이터베이스 구조, 기능, 관리, 외부 응용 프로그램과의 상호 작용)
- 준비
- 기술 역량: **SQL**, 데이터베이스 관리
- 방법
- 시간
- 리소스

이 가이드를 사용하는 방법

팁:

이 가이드를 읽기 전에 다른 **Asset Manager 9.30** 가이드를 읽어 보는 것이 좋습니다.

- **설치 및 업그레이드**
- **릴리스 노트**
- **Migrating customized compact SI from AC 4.4.x to AM 5.1x**(<Asset Manager 9.30 설치 폴더>\doc\white_papers\Software Asset Management에 있음)

팁:

이 가이드의 전체 내용을 제공된 순서대로 읽어 보는 것도 좋습니다.

지원되는 환경 장

이 장에는 마이그레이션을 통해 지원되는 환경 목록이 포함되어 있습니다. 사용 중인 구성이 지원되는지 이 장에서 확인합니다.

마이그레이션 프로세스 장

이 장에서는 마이그레이션 프로세스에 대해 간략하게 소개합니다.

이 프로세스는 마이그레이션하려는 **Asset Manager**의 버전에 따라 다릅니다. 마이그레이션 프로세스 단계에 대해 알아 보려면 이전에 설치한 버전에 해당하는 장의 섹션을 읽으십시오.

단계별 마이그레이션 - 준비 단계(프로덕션 데이터베이스) 장

단계별 마이그레이션 - 시뮬레이션(시뮬레이션 데이터베이스) 장

이 장의 내용을 전체적으로 읽어보는 것을 시작으로 하여 마이그레이션 프로세스 전체에서 수행해야 하는 모든 단계에 익숙해질 수 있습니다.

그런 다음 각각의 세부 내용에 주의하여 이 가이드에 제공된 순서대로 한 단계씩 계속 진행해야 합니다.

용어집 장

이 장에서는 마이그레이션에 사용된 주요 용어를 정의합니다.

이 가이드에 사용된 용어에 대해 알아 보려면 이 장을 확인합니다.

참조 장

이 장에는 완벽하고 체계적인 참조 정보가 있습니다.

고급 정보나 보충 정보를 얻으려면 이 장을 확인합니다.

1 지원되는 환경

운영 체제 및 DBMS

이 마이그레이션은 Asset Manager에서 지원하는 모든 운영 체제와 DBMS에서 작동합니다.

지원되는 운영 체제 및 DBMS를 알아보려면 웹 사이트에서 Support Matrix를 참조하십시오: www.hp.com/go/hpsupport.

DBMS 서버에 필요한 디스크 공간

경고:

Asset Manager 버전 4.1.0 이상을 마이그레이션하려는 경우에는 이 섹션을 건너뛰어도 좋습니다.

DBMS 서버에서 이전 형식 프로덕션 데이터베이스에 할당된 디스크 공간은 이전 형식 프로덕션 데이터베이스 크기의 두 배 이상이어야 합니다.

2 마이그레이션 프로세스

HP Connect-It 사용 시 주의 사항

이전 형식 프로덕션 데이터베이스를 변환하기 위해 HP Connect-It을 사용해서는 안 됩니다.

마이그레이션에 필요한 작업은 무엇입니까?

이 마이그레이션은 다음과 같이 **Asset Manager**의 이전 버전을 버전 9.30으로 변환하는 데 필요한 작업 집합입니다.

- **Asset Manager**의 9.30 버전과 호환되도록 **이전 형식 프로덕션 데이터베이스** 변환(구조 및 내용)
- 모든 관리 시스템 및 사용자 시스템에서 **Asset Manager** 프로그램을 버전 9.30으로 업데이트

데이터베이스 변환은 복잡한 프로세스이므로 이 장에서는 일반적인 원리부터 설명합니다.

반면에 프로그램 업데이트는 기존에도 많이 수행되었던 작업이므로 이 가이드에서는 업데이트의 일반적인 원리에 대해서는 설명하지 않겠습니다.

변환에 필요한 작업은 무엇입니까?

데이터베이스를 변환하려면 다음 작업을 수행해야 합니다.

- 현재 데이터베이스 구조를 **Asset Manager 9.30** 버전의 데이터베이스 구조에 맞게 변경
- 가능한 한 데이터는 원래대로 보존
- 데이터베이스 구조 변경으로 인해 원래 상태로 보존할 수 없는 데이터 수정. 이러한 수정 작업은 가능한 한 자동으로 수행되지만, 그렇지 않을 경우에는 수동으로 수행해야 합니다.

도구를 통해 변환되는 항목은 무엇입니까?

- 데이터베이스 구조 전체
- 대부분의 데이터
데이터는 데이터베이스의 테이블, 필드 및 링크를 참조하지만, 사용자가 검증하고 가능한 한 수동으로 수정해야 합니다.
이러한 데이터 항목 목록은 다음을 참조하십시오: ▶ 수동으로 변환할 응용 프로그램 데이터 [108 페이지].



경고:

변환 도구는 마이그레이션에만 사용할 수 있습니다.

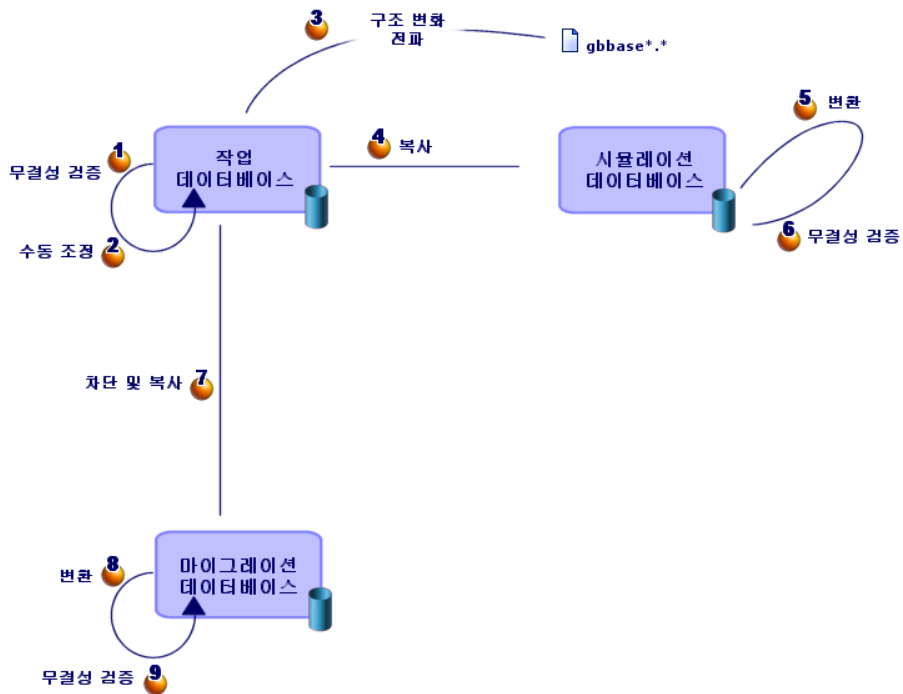
Asset Manager 버전 4.2.x, 4.3.x, 4.4.x 또는 5.0x에서 마이그레이션하기 위한 단순 프로세스

- 1 다음과 같이 **시뮬레이션 데이터베이스**에서 변환을 시뮬레이션합니다.
 - 1 **Asset Manager Application Designer**를 사용하여 **이전 형식 프로덕션 데이터베이스**의 무결성을 검증합니다.
 - ▶ 1단계 - 이전 형식 프로덕션 데이터베이스의 무결성 검증 [31 페이지]
 - 2 필요한 경우 **Asset Manager**를 사용하여 **이전 형식 프로덕션 데이터베이스**를 수동으로 조정합니다.
 - ▶ 2단계 - 이전 형식 프로덕션 데이터베이스의 수동 조정 [33 페이지]
 - 3 **이전 형식 프로덕션 데이터베이스**에 대한 구조적 변경 사항을 표준 **9.30 gbbase*. * 데이터베이스 설명 파일**로 전파합니다.

- ▶ 3단계 - 이전 형식 프로덕션 데이터베이스에 대한 구조 변경 전파 [37 페이지]
- 4 이전 형식 프로덕션 데이터베이스의 백업을 만듭니다. 이 백업을 **시뮬레이션 데이터베이스**라고 합니다. 사용자는 이 **시뮬레이션 데이터베이스**에서 변환을 시뮬레이션하는 동안 **이전 형식 프로덕션 데이터베이스**에서 계속 작업을 수행할 수 있습니다.
- ▶ 4단계 - 이전 형식 프로덕션 데이터베이스 복사 [41 페이지]
- 5 Asset Manager Application Designer를 사용하여 **이전 형식 시뮬레이션 데이터베이스**를 변환합니다.
이렇게 하면 구조적 변경 사항을 유지하면서 **이전 형식 시뮬레이션 데이터베이스**의 구조적 매개 변수를 변환할 수 있습니다.
- ▶ 이전 형식 시뮬레이션 데이터베이스 변환 [45 페이지]
- 6 Asset Manager Application Designer를 사용하여 **9.30 이전 형식 시뮬레이션 데이터베이스**의 무결성을 검증합니다.
이렇게 하면 변환으로 인해 **시뮬레이션 데이터베이스**가 손상되지 않았는지 확인할 수 있습니다.
- ▶ 6단계 - 9.30 형식 시뮬레이션 데이터베이스의 무결성 검증 [51 페이지]
- 2 다음과 같이 **마이그레이션 데이터베이스**라고 하는 이전 형식 프로덕션 데이터베이스의 두 번째 백업을 변환합니다.
- 7 이전 형식 프로덕션 데이터베이스를 차단하고 **마이그레이션 데이터베이스**라는 백업을 만듭니다.
- ▶ 16단계 - 이전 형식 프로덕션 데이터베이스 차단 및 복사 [71 페이지]
- 8 Asset Manager Application Designer를 사용하여 **이전 형식 마이그레이션 데이터베이스**를 변환합니다.
- ▶ 이전 형식 시뮬레이션 데이터베이스 변환 [45 페이지]
- 이전 형식 시뮬레이션 데이터베이스**에 연결하는 대신에 **이전 형식 마이그레이션 데이터베이스**에 연결해야 합니다.
- 9 Asset Manager Application Designer를 사용하여 **9.30 이전 형식 마이그레이션 데이터베이스**의 무결성을 검증합니다.
- ▶ 6단계 - 9.30 형식 시뮬레이션 데이터베이스의 무결성 검증 [51 페이지]
- 이전 형식 프로덕션 데이터베이스**에 연결하는 대신에 **9.30 형식 마이그레이션 데이터베이스**에 연결합니다.
- 3 Asset Manager를 통해 **9.30 형식 마이그레이션 데이터베이스**를 완료하여 변환 프로세스를 마칩니다. 이 **9.30 형식 마이그레이션 데이터베이스**가 프로그램을 업그레이드한 후 프로덕션에 배치할 데이터베이스입니다.
- ▶ 20단계 - 9.30 형식 마이그레이션 데이터베이스 완료 [72 페이지]
- 4 10 Asset Manager 데이터베이스에 액세스하는 외부 소프트웨어 구성 요소를 업그레이드합니다.

- ▶ 21단계 - Asset Manager 데이터베이스에 액세스하는 외부 소프트웨어 구성 요소 업그레이드 [81 페이지]
 - 5 **11** Asset Manager 프로그램을 업데이트합니다.
 - ▶ 22단계 - Asset Manager 프로그램 업그레이드 [83 페이지]
 - 6 **12** 9.30 형식 마이그레이션 데이터베이스를 프로덕션에 배치합니다.
 - ▶ 23단계 - 9.30 형식 마이그레이션 데이터베이스를 프로덕션에 배치 [85 페이지]
 - 7 **13** 더 이상 사용되지 않는 프로그램을 모두 제거합니다.
 - ▶ 24단계 - 더 이상 사용되지 않는 프로그램 설치 제거 [85 페이지]
- 다음은 변환 프로세스의 기본 단계입니다.

2.1. 그림 변환 - Asset Manager 버전 4.2.x, 4.3.x 또는 4.4.x 에서 마이그레이션하기 위한 단순 프로세스



변환 도구는 어떻게 작동합니까?

변환 도구는 다음 제품에 통합되어 있습니다.

- Asset Manager Application Designer 9.30을 시작합니다.
- Asset Manager Script Analyzer 9.30.

이러한 프로그램은 Asset Manager 프로그램 그룹에서 실행됩니다.

Asset Manager Application Designer에 통합된 도구는 다음 메뉴를 통해 액세스할 수 있습니다.

- **작업/진단/데이터베이스 복구**

이 도구는 현재 데이터베이스를 검증하고 복원합니다.

- **마이그레이션/사용자 지정 구조 전파**

이 도구는 이전 형식 프로덕션 데이터베이스에 대한 사용자 지정 사항을 표준 9.30 gbbase*. * 데이터베이스 설명 파일로 전파합니다.

- **마이그레이션/응용 프로그램 데이터 내보내기**

이 도구는 사용자가 수동으로 조작할 수 있도록 XML 형식으로 수동 변환할 응용 프로그램 데이터의 복사본을 내보냅니다.

- **마이그레이션/데이터베이스 변환**

이 도구는 migration.xml 변환 파일의 사양에 따라 현재 데이터베이스의 구조를 변환합니다.

- **마이그레이션/응용 프로그램 데이터 복원**

이 도구는 분석 및 수정한 응용 프로그램 데이터를 가져옵니다.

변환 프로세스가 이전 버전과 어떻게 다른니까?

이전 형식 프로덕션 데이터베이스를 변환하는 작업에 빈 데이터베이스로 이전 데이터를 가져오는 작업(4.0.0 이전 버전에서는 이 작업이 포함됨) 더 이상 포함되지 않습니다.

변환 도구는 이전 형식 프로덕션 데이터베이스에서 필요한 수정 작업을 직접 수행합니다.

이 새로운 기술의 이점은 다음과 같이 다양합니다.

- 변환 기간이 크게 단축됩니다.
- 새 구조에도 계속 존재할 필드에 저장된 데이터의 경우에는 수정이 이루어지지 않습니다. 따라서 이 데이터를 가져올 필요가 없으므로 변환 기간이 다시 단축됩니다.
- migration.xml 변환 파일을 사용자 지정할 수 있습니다.
 - 이 파일은 XML 형식입니다.

- 이 파일은 단순 텍스트 또는 XML 텍스트 편집기를 사용하여 편집할 수 있습니다.
- 이 파일은 대체로 DBMS와 독립적이며, DBMS를 위해 SQL 명령으로 변환됩니다.

Asset Manager Script Analyzer에서는 내보낸 응용 프로그램 데이터를 복원하기 전에 **마이그레이션/응용 프로그램 데이터 내보내기** 메뉴를 사용하여 해당 데이터를 수동으로 변환할 수 있습니다.

버전 9.30에서 DBMS가 지원되지 않는 데이터베이스 마이그레이션

이전 형식 프로덕션 데이터베이스의 DBMS가 9.30 버전에서 지원되지 않는 경우:

- 1 이전 형식 프로덕션 데이터베이스를 Asset Manager 9.30에서 지원되는 DBMS로 전송합니다.
이 작업의 수행 방법은 **관리 가이드의 Asset Manager 데이터베이스 생성, 수정 및 삭제** 장, **DBMS 변경** 섹션을 참조하십시오.
- 2 이 가이드의 설명에 따라 마이그레이션을 진행합니다.

조달 모듈의 제한 사항

이전 형식 프로덕션 데이터베이스를 변환한 후에는 다음 작업을 더 이상 수행할 수 없습니다.

- 변환 전에 부분적으로 받은 주문 수령
- 변환 전에 받은 자산 반환

따라서 이전 형식 프로덕션 데이터베이스를 변환하기 전에 이러한 작업을 수행하는 것이 좋습니다.

마이그레이션의 복잡성

이 가이드에 제공된 방법을 통해 다양한 문제를 예측하고 방지할 수 있습니다.

단, 이 방법을 회사 고유의 Asset Manager 사용 방식에 맞게 조정해야 합니다.

변환의 복잡성은 이전 형식 프로덕션 데이터베이스에 대한 사용자 지정 정도에 따라 다릅니다.

3 단계별 마이그레이션 - 준비 단계(프로덕션 데이터베이스)

이 장에서는 변환하기 전에 수행할 작업을 단계별로 설명합니다.

예비 분석

마이그레이션 프로세스를 구현하기 전에 요구와 제약 조건을 완벽하게 분석하는 작업부터 시작해야 합니다.

- 1 이 가이드에 설명된 대로 마이그레이션의 모든 측면을 처리할 수 있는지 확인합니다.
- 2 **Asset Manager 9.30**에 대한 수정 사항에 대해 알아보십시오.
▶ 기타 설명서 [112 페이지]
- 3 이러한 수정 사항(새 기능, 기능 수정 사항 등)이 **Asset Manager**의 사용에 어떠한 영향을 미치는지 확인합니다.
- 4 이러한 새 기능을 구현할 시기(마이그레이션과 동시에 또는 마이그레이션 이후)를 결정합니다.
- 5 이러한 영향에 따라 프로젝트 사양(작업 구성, 데이터 구성, 매개 변수 구성 등)을 업데이트합니다.
- 6 사용자 및 사용자 교육과 관련된 설명서를 업데이트합니다.

마이그레이션 프로젝트 시작

Asset Manager의 버전 9.30에 대한 향상 및 변경 정도를 고려하여 마이그레이션 프로세스에 다음 업무를 담당하는 사람을 포함시켜야 합니다.

- 명칭
- 다음 기본 모듈 배포:
 - 조달
 - 계약
 - 자금 조달
 - 헬프데스크
 - 케이블 및 회선
- 인벤토리
- 데이터베이스 사용자 지정
- 보고서, 쿼리, 워크플로우 스키마, 작업 등 생성
- **Asset Manager**와 외부 응용 프로그램 통합
- 사용자 교육
- 사용자 지원

프로젝트 착수 시 이러한 사용자를 식별하고 이들에게 정보를 전달해야 합니다.

팁:

이전 버전을 구현하는 데 사용된 프로젝트 사양을 찾는 것이 좋습니다.

마이그레이션 목적을 나타내고, 작업을 분담하고, 계획을 정의하기 위해 앞에서 언급한 모든 사람들을 참가시켜 프로젝트 초기화 회의를 개최해야 합니다.

Asset Manager의 사용이 다소 고급화된 경우(다양한 무결성 규칙, 자동 메커니즘, 매개 변수 사용자 지정) 프로젝트 관리자 조정 하에 각 기능 또는 기술 도메인에 팀을 배정할 수 있습니다.

경고:

이 마이그레이션에서는 여러 가지 기술적 측면을 다룹니다. 따라서 각 팀에 자격을 갖춘 엔지니어가 한 명 이상 있어야 합니다. 특히 기본적으로 제공된 **migration.xml** 변환 파일을 수정할 수 있다고 생각하는 경우 해박한 **SQL** 지식을 보유한 사람이 필요합니다.

이러한 새로운 기능을 즉시 활용하려는 경우에는 프로젝트 사양을 수정하고 매개 변수를 다시 구성해야 합니다.



팁:

특정 기업에서는 타산에 맞게 마이그레이션 프로세스를 여러 개의 개별 단계로 분할하는 경향이 있습니다.

- 1 Asset Manager 이전 버전 수준의 기능을 얻고 이를 안정화하는 것부터 시작합니다.
- 2 Asset Manager 9.30의 새 기능을 살펴봅니다.
이렇게 하면 전환을 보다 원활하게 수행할 수 있습니다.



팁:

의문 사항이 있으면 HP 또는 파트너 업체에 즉시 문의하십시오. 마이그레이션 프로젝트의 어떤 단계에서든 경험이 풍부한 전문 컨설턴트의 도움을 받으실 수 있습니다.

사용자 및 지원 기술자 교육

프로그램을 마이그레이션하고 이전 형식 프로덕션 데이터베이스를 변환하는 경우 **Asset Manager**를 사용하고 이 소프트웨어의 사용을 지원하는 인력의 교육에 대해 생각해 볼 수 있습니다.

이렇게 하려면 다음을 수행합니다.

- 1 교육 요구 사항을 정의합니다.
- 2 교육 일정을 정의합니다.
- 3 교육 자료를 준비합니다.
- 4 사용자 노트를 업데이트합니다.



경고:

9.30 형식 프로덕션 데이터베이스를 프로덕션에 배치하기 전에 Asset Manager 사용자를 교육해야 합니다.



팁:

의문 사항이 있으면 HP 또는 파트너 업체에 즉시 문의하십시오. 교육에 대해 전문적이고 경험이 풍부한 컨설턴트의 도움을 받으실 수 있습니다.

변환 컴퓨터 준비

이전 형식 프로덕션 데이터베이스를 변환하기 전에 이 변환에 맞는 컴퓨터를 준비해야 합니다.

이 장에서는 변환 컴퓨터에 설치해야 하는 모든 항목에 대해 나열합니다.

이전 형식 프로덕션 데이터베이스에 해당하는 **Asset Manager 버전 설치**

프로덕션 데이터베이스에 액세스하려면 이 버전을 설치해야 합니다.

- 프로덕션 데이터베이스
- 시뮬레이션 데이터베이스
- 마이그레이션 데이터베이스

최소한 기본 모듈을 설치해야 합니다.

이전 형식 프로덕션 데이터베이스에 대한 액세스 권한이 있는지 검증

이 작업을 수행하면 다음 작업이 가능합니다.

- 변환을 위해 이전 형식 프로덕션 데이터베이스를 준비합니다.
- 시뮬레이션할 이전 형식 프로덕션 데이터베이스의 백업을 만든 다음 변환을 수행합니다.

Asset Manager 9.30 설치

최소한 다음 구성 요소를 설치해야 합니다.

- Asset Manager 클라이언트
- Asset Manager Application Designer
- 설명서
- 마이그레이션
- 데이터 키트
- Asset Manager Export Tool
- HP AutoPass License Management Tool

데이터베이스에 적용되는 모든 라이선스 키는 이 **HP AutoPass License Management Tool** 인스턴스에 설치되어야 합니다.

▶ **관리 가이드, 라이선스 키 설치 장.**



팁:

HP AutoPass License Management Tool는 설치하는 모든 Asset Manager 9.30 구성 요소와 함께 자동으로 설치됩니다.

그러나 라이선스 키는 수동으로 설치해야 합니다.



참고:

이전 형식 프로덕션 데이터베이스가 다국어 버전(▶ **관리 가이드의 Asset Manager 데이터베이스 생성, 수정 및 삭제** 장, **Asset Manager 클라이언트 언어 수정** 섹션)이며, 특정 다국어 항목을 사용자 지정했고, 다국어 항목(▶ **구조 변경 전파 [39 페이지]**)을 자동으로 변환하려는 경우에는 9.30 버전을 다른 추가 언어에서 사용할 수 있어야 하며, 변환에 사용되는 컴퓨터에 이러한 언어로 Asset Manager를 설치해야 합니다.

필드 및 링크에 대한 컨텍스트 도움말을 제외하고 모든 다중 언어 요소가 전파됩니다(▶ **필드 도움말 [73 페이지]**).

자동으로 언어 X에 대한 사용자 지정을 전파하려면 이 언어로 Asset Manager를 사용할 수 있을 때까지 기다려야 합니다.

기다리는 대신 이미 사용 가능한 언어 버전에서 업그레이드를 수행할 수 있지만 그럴 경우 언어 X에 적용한 사용자 지정을 전파할 수 없게 됩니다. Asset Manager 9.30을 해당 언어로 사용할 수 있게 되면 언어 X를 **9.30 형식 프로덕션 데이터베이스**에 삽입합니다. **수동으로 이전 형식 프로덕션 데이터베이스**에 적용된 사용자 지정을 전파해야 합니다.

HP Connect-It 설치(Asset Manager 9.30에서 제공되는 버전)

수정 후 수동으로 변환할 응용 프로그램 데이터를 복원하려면 이 소프트웨어가 필요합니다.



참고:

HP Connect-It을 사용하려면 인증 키가 필요합니다. 이 키를 보유하고 있는지 가능한 빨리 확인하십시오. 필요한 경우, HP Connect-It를 사용하기 전에 HP에 문의하여 키를 구합니다.



경고:

이전 형식 프로덕션 데이터베이스를 변환하기 위해 HP Connect-It을 사용해서는 안 됩니다.

XML 파일 편집기 설치

XML 파일 편집기 설치에 선택 사항(표준 텍스트 편집기로 충분함)이지만 migration.xml 변환 파일을 편집하고 XML 구조가 올바른지 검증하려는 경우에는 XML 파일 편집기가 상당히 간편합니다.

Java Runtime 환경 설치(Asset Manager 9.30에서 제공되는 버전)

이전 형식 프로덕션 데이터베이스 구조에 적용된 사용자 지정을 변환하려면 이 도구가 필요합니다.

메모리 문제를 방지하기 위해 Java 힙 크기를 증가시키려면 다음과 같이 수행합니다.

- 1 amdba.ini 파일을 찾습니다: ▶ **설치 및 업그레이드 가이드**, **.ini 및 .cfg 파일** 장, **사용 가능한 .ini 및 .cfg 파일** 섹션
- 2 텍스트 편집기에서 파일을 엽니다.
- 3 **[Option]** 섹션에서, **/Advanced/SduJavaCmd** 매개 변수를 추가 또는 수정하고 해당 값을 **java -Xmx500M: /Advanced/SduJavaCmd=java -Xmx500M**로 설정합니다.
- 4 amdba.ini를 저장합니다.

변환 비율에 영향을 주는 요소

- DBMS 성능
- Asset Manager Application Designer 시스템 및 이전 형식 데이터베이스 시스템 간의 처리량
- Asset Manager Application Designer 및 이전 형식 데이터베이스가 설치된 시스템의 성능(최소한으로만 설치)

팁:

이전 형식 프로덕션 데이터베이스에 많은 데이터가 저장되어 있는 경우 Asset Manager Application Designer가 설치된 컴퓨터를 데이터베이스에 가능한 가까이(예: WAN을 거치지 않고) 두어야 합니다. 이는 길이가 아주 긴 필드와 이진 데이터를 포함하는 테이블인 경우 특히 그렇습니다(예: **amComment** 및 **amImage**).

DBMS 서버 준비

이전 형식 데이터베이스에 충분한 공간 할당

마이그레이션하는 동안 **이전 형식 시뮬레이션 데이터베이스**와 **이전 형식 마이그레이션 데이터베이스**를 변환해야 합니다.

이러한 각 데이터베이스에 충분한 공간을 할당했는지 확인해야 합니다. 충분한 공간이 할당되지 않은 경우 변환에 실패할 위험이 있습니다.

▶ DBMS 서버에 필요한 디스크 공간 [15 페이지]

블랙 세그먼트

참고:

블랙 세그먼트는 Oracle에서 사용되는 용어입니다.

Microsoft SQL Server에서는 **트랜잭션 로그**와 같은 의미입니다.

변환 중에 필요한 최대 트랜잭션을 지원하려면 **전체** 롤백 세그먼트를 정의해야 합니다.

이 트랜잭션은 가장 많은 공간을 차지하는 테이블 전체에서 단일 운영으로 **INSERT**를 수행하는 작업으로 이루어집니다.

4 단계별 마이그레이션 - 시뮬레이션(시뮬레이션 데이터베이스)

이전 형식 프로덕션 데이터베이스를 변환하기 전에 이 변환의 시뮬레이션을 수행해야 합니다.

하지만 이러한 시뮬레이션을 **프로덕션 데이터베이스**에서는 수행할 수 없습니다. 시뮬레이션은 **시뮬레이션 데이터베이스**에서만 수행할 수 있습니다.

이와 동시에 사용자는 이전 형식 프로덕션 데이터베이스를 계속 정상적으로 사용할 수 있습니다.

시뮬레이션이 완료되고 나면 **마이그레이션 데이터베이스**라고 하는 이전 형식 프로덕션 데이터베이스의 또 다른 백업을 변환할 수 있습니다.

이 **9.30 형식 마이그레이션 데이터베이스**가 프로덕션에 배치할 데이터베이스입니다.

이 장에서는 **시뮬레이션 데이터베이스**에서 수행할 작업을 단계별로 설명합니다.

1단계 - 이전 형식 프로덕션 데이터베이스의 무결성 검증

-  **[중요]:**
이전 형식 프로덕션 데이터베이스를 백업합니다.
- 이전 버전 Asset Manager Application Designer에서 선택 사항인 초기 검증을 수행하는 경우

 **경고:**

이 확인 절차는 선택 사항입니다.

유효성 스크립트가 있고 많은 수의 레코드를 포함하는 테이블에서 **레코드의 유효성 확인** 옵션을 사용하면 하루 이상 걸릴 수도 있습니다.

이런 테이블의 경우 테이블의 각 레코드에 대해 유효성 스크립트가 수행됩니다.

경우에 따라 검사가 아예 수행되지 못할 수도 있습니다.

- 1 이전 버전의 **Asset Manager Application Designer**를 시작합니다.
- 2 **이전 형식 프로덕션 데이터베이스**에 연결합니다(**파일/열기** 메뉴, **기존 데이터베이스 열기** 옵션).
- 3 데이터베이스 진단 창을 표시합니다(**작업/진단/데이터베이스 복구** 메뉴).
- 4 테이블 목록에서 (**모든 테이블**)을 선택합니다.
- 5 로그 파일의 이름과 위치를 지정합니다.
- 6 **레코드의 유효성 확인** 옵션만 선택합니다.
- 7 **복구** 옵션을 선택합니다.
- 8 **시작**을 클릭합니다.
- 9 실행 창의 메시지를 참조하십시오.
- 10 필요한 경우 로그 파일을 참조하십시오.

3

 **경고:**

이전 형식 프로덕션 데이터베이스의 DBMS가 DB2인 경우 두 번째 검증을 수행하지 않고 여기서 정지합니다.

9.30 형식 **Asset Manager Application Designer**를 통해 두 번째 검증을 수행합니다.

- 1 **Asset Manager Application Designer** 버전 9.30을 시작합니다.
- 2 **이전 형식 프로덕션 데이터베이스**에 연결합니다(**파일/열기** 메뉴, **기존 데이터베이스 열기** 옵션).

 **참고:**

Asset Manager Application Designer 9.30을 사용하여 이전 형식 데이터베이스에 연결할 수 있습니다.

- 3 데이터베이스 진단 창을 표시합니다(**작업/진단/데이터베이스 복구** 메뉴).
- 4 테이블 목록에서 (**모든 테이블**)을 선택합니다.
- 5 로그 파일의 이름과 위치를 지정합니다.
- 6 **레코드의 유효성 확인** 옵션을 제외한 모든 검증 옵션을 선택합니다.
- 7 **복구** 옵션을 선택합니다.

- 8 시작을 클릭합니다.
- 9 실행 창의 메시지를 참조하십시오.
- 10 필요한 경우 로그 파일을 참조하십시오.

분석 및 복구 프로그램에 대한 자세한 내용은 **관리 가이드의 데이터베이스의 진단 및 복구** 장을 참조하십시오.

3단계 - 이전 형식 프로덕션 데이터베이스의 수동 조정

경고:

이 섹션에 설명된 조정을 수행하기 전에 **이전 형식 프로덕션 데이터베이스**의 백업 복사본을 만드는 것이 좋습니다.

프로세스를 원활하게 수행하려면 **이전 형식 프로덕션 데이터베이스**를 변환하기 전에 특정 데이터를 수정해야 합니다.

이전 형식 프로덕션 데이터베이스에서 준수해야 할 대부분의 제약 조건은 **migration.xml** 변환 파일의 **Mapping** 요소에 의해 추론됩니다.

이 섹션에서는 표준 변환 파일에 의해 추론되는 제약 조건 목록을 제공합니다. 표준 변환 파일을 수정할 경우 사용자 고유 변경 사항에 의해 추론되는 제약 조건을 식별하고 검증해야 합니다.

이전 형식 프로덕션 데이터베이스의 모든 버전과 관련된 조정

amCounter 테이블 업데이트

이 섹션은 저장된 프로시저 **up_GetCounterVal**을 수정한 사용자를 위한 것입니다. 이 프로시저는 다음 기술 노트의 지시문에 따라 **amCounter** 테이블을 관리합니다.

- Microsoft SQL Server: TN317171736
- Oracle Database Server: TN12516652
- DB2 UDB: TN1029175140(Asset Manager용 버전 3.x)

이러한 기술 노트에 설명된 수정 사항을 적용한 경우 **amCounter** 테이블의 특정 레코드는 저장된 프로시저 **up_GetCounterVal**에 의해 더 이상 업데이트되지 않습니다.

따라서 **이전 형식 프로덕션 데이터베이스**를 변환하기 전에 다음을 수행해야 합니다.

- 1 변환 후 동일한 방법으로 수정하려는 경우 **up_GetCounterVal**에서 저장된 프로시저를 복사합니다.

- 2 다른 테이블로 전환된 **amCounter** 테이블의 카운터를 수동으로 업데이트합니다.
- 3 저장된 프로시저 **up_GetCounterVal**을 원래 상태로 복원합니다.

필드 및 링크의 필수 특성

지정된 테이블에서 레코드를 생성하기 전에 특정 필드 및 링크를 채워야 합니다.

필드 및 링크의 필수 특성은 **Asset Manager** 데이터베이스 또는 **gbbase*.*** 데이터베이스 설명 파일에 정의되어 있습니다.

이 필수 특성은 모든 경우에 **true**이거나 스크립트를 사용하여 계산할 수 있습니다. 변환 프로그램을 통해 작성하거나 수정한 레코드에서는 필드 및 링크의 필수 특성을 준수해야 합니다. 이 필수 특성은 사용자 지정된 **9.30 gbbase*.*** 데이터베이스 설명 파일에 명시되어 있습니다.

필드 및 링크에는 **migration.xml** 변환 파일에 설명된 명시적 연관이나, 필드 또는 링크가 동일한 **SQL** 이름을 공유할 때 자동으로 추론되는 암시적 연관이 있어야 합니다.

Asset Manager 9.30과 함께 기본적으로 설치되는 **migration.xml** 변환 파일은 **이전 형식 프로덕션 데이터베이스**와 데이터베이스 설명 파일인 표준 **9.30 gbbase*.***의 형식을 수정하지 않은 경우 제대로 작동할 수 있도록 고안되었습니다.

다음의 경우를 제외하고는 표준 **migration.xml** 변환 파일을 조정할 수 없습니다.

- 이전 형식 프로덕션 데이터베이스 사용 시 필드 또는 링크의 필수 특성을 삭제한 경우
- 표준 **9.30 gbbase*.*** 데이터베이스 설명 파일의 특정 필드 또는 링크에 필수 특성을 추가한 경우

필수 필드 및 링크를 채우기 위해 변환 파일에서 **이전 형식 프로덕션 데이터베이스**의 특정 데이터를 사용할 수 있습니다.

변환하기 전에 사용자 지정된 **9.30 gbbase*.*** 데이터베이스 설명 파일에 필수로 선언된 필드 및 링크가 이전 형식 프로덕션 데이터베이스에 채워져 있는지 확인해야 합니다.

예를 들어, **amAsset** 테이블의 **ICategId** 필드를 사용하는 경우입니다.

채워진 링크가 의심스러운 경우에는 외부 키가 채워져 있는지 검증하십시오.

필드 값 길이

이전 형식 프로덕션 데이터베이스의 특정 필드는 **9.30** 형식 프로덕션 데이터베이스의 다른 필드를 채우는 데 사용됩니다.

이러한 원본 필드 중 일부는 대상 필드보다 더 깁니다.

문제가 되는 경우에는 이러한 원본 필드에 저장된 값의 길이가 대상 필드의 크기를 초과하지 않는지 검증해야 합니다.

이 문제가 발생하면 다음 방법을 통해 해결할 수 있습니다.

- 원본 값의 길이를 줄입니다.
- 사용자 지정된 **9.30 gbbase*.*** 파일에서 대상 필드의 크기를 늘립니다.

값이 너무 길면 변환 중에 잘립니다.

^ 문자

이 문자는 이전 형식 프로덕션 데이터베이스의 필드 값에 사용해서는 안 되며, 특히 다음 필드의 값에는 절대 사용할 수 없습니다. 이러한 필드 중에 이전 형식 프로덕션 데이터베이스의 사용자 버전에서 사용되는 필드를 확인할 수 있습니다.

4.1. 표 ^ 문자를 사용할 수 없는 필드 - 목록

테이블의 SQL 이름	필드의 SQL 이름
amProduct	Model
amProduct	CatalogRef
amSoftware	Name
amCatalog	Code
amCompany	Code
amCompany	Name
amProdSupp	PriceCur
amCatProduct	FullName
amAccessRestr	SQLName
amAssetRent	Code
amBrand	BarCode
amBudgClass	Code
amBudgClass	Name
amBudget	Code
amBudget	Name
amBudget	Type
amBudgetCategory	Code
amCategory	Name
amCategory	BarCode
amCategory	FullName
amCategory	sLvl
amCntrRent	Code
amDateAlarm	Code
amDeprScheme	Code
amEscSchLevel	Code
amFloorPlan	Code
amFuncDomain	SQLName
amFuncDomain	Name
amReservation	ItemNo
amLocation	BarCode
amLocation	FullName
amLocation	Name
amLossValRule	Code
amModel	BarCode
amModel	FullName
amModel	Name

테이블의 SQL 이름	필드의 SQL 이름
amContract	Ref
amNature	Code
amNature	Name
amNews	Topic
amPeriod	Name
amPeriod	Code
amEstimate	PONumber
amEstimate	EstimNumber
amPOrdLine	FullName
amPOrdLine	ItemNo
amEstimLine	FullName
amEstimLine	ItemNo
amPortfolio	Code
amPortfolio	FullName
amConsUse	ItemNo
amAsset	FullName
amAsset	AssetTag
amProdCompo	FullName
amProfile	SQLName
amProject	Code
amReceipt	ReceiptNumber
amRequest	ReqNumber
amSoftLicCounter	Code
amThirdParty	Code
amUserRight	SQLName
amPOrder	PONumber
amTaxFormula	Code

조달 및 워크플로우 모듈

변환하기 전에 진행 중인 실행(부분 수령한 주문, 반환할 항목, 워크플로우 등)을 가능한 한 많이 완료하는 것이 좋습니다.

경고:

변환 중에 문제가 발생할 경우 참조로 사용할 수 있도록 이전 형식 프로덕션 데이터베이스 복사본을 잘 보존하는 것도 좋습니다.

전체 이름 필드

/ 문자를 포함하는 문자열을 사용하여 **전체 이름(FullName)** 필드를 채우는 경우 / 문자는 계층 수준의 구분자로 해석됩니다.

특정 DBMS에서는 / 문자를 중립 문자로 대체하도록 표준 변환 파일을 구성할 수 있으므로 이 문자는 문제가 되지 않습니다.

대신 필드에서 **전체 이름** 필드를 채우는 데 사용된 다른 문자를 선택하여 / 문자를 대체해야 합니다.

다음 필드를 검증해야 합니다.

4.2. 표 / 문자를 사용할 수 없는 필드 - 목록

테이블의 SQL 이름	필드의 SQL 이름	관련 버전:						
		3.0.1	3.0.2	3.1.0	3.5.0	3.5.1	3.6.0	4.0.0
amItemListVal	Value(브랜드의 경우)	예	예	예	예	예	예	
amFamily	브랜드	예	예	예	예	예	예	
amFamily	Name	예	예	예	예	예	예	
amAsset	ComputerName							예
amAsset	AssetTag	예	예	예	예	예	예	
amInvoice	InvoiceNumber	예	예	예	예	예	예	
amProduct	Model	예	예	예	예	예	예	
amSoftware	Publisher	예	예	예	예	예	예	
amSoftware	Name	예	예	예	예	예	예	
amSoftware	VersionLevel	예	예	예	예	예	예	
amContract	Ref	예	예	예	예	예	예	
amAdjustment	ItemNo	예	예	예	예	예	예	
amConsUse	ItemNo	예	예	예	예	예	예	
amComputer	Name							예

3 단계 - 이전 형식 프로덕션 데이터베이스에 대한 구조 변경 전파

⚠ 경고:

이 작업을 수행하려면 여기서 사용하는 표준 9.30 gbbase*.* 데이터베이스 설명 파일이 Asset Manager 9.30과 함께 설치된 표준 파일이어야 합니다. 수정이 완료되면 이러한 파일을 사용할 수 없습니다.

이 작업은

- 이전 형식 프로덕션 데이터베이스의 표준 구조를 수정(필드, 인덱스 및 테이블 추가 또는 수정)했으며 9.30 형식 프로덕션 데이터베이스에서 이러한 변경 사항을 유지하려는 사용자를 대상으로 합니다.
- 표준 9.30 gbbase*.* 파일의 구조 수정을 전파하는 것을 목표로 합니다.

💡 팁:

가져온 표준 9.30 gbbase*.* 파일을 사용하여 변환 중에 **9.30 형식 데이터베이스**를 구조화합니다.

- Asset Manager Application Designer에서 제공하는 이 작업 전용 도구를 사용합니다.

경고:

Asset Manager Application Designer를 사용하여 수행한 이전 형식 프로덕션 데이터베이스에 대한 구조 변경만 고려됩니다.

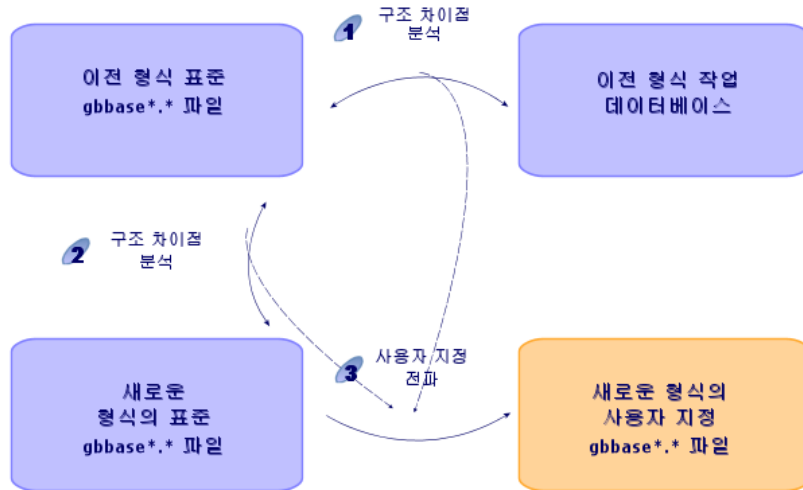
이전 형식 프로덕션 데이터베이스에서 다른 방법을 통해 수행한 구조 변경의 경우에는 모두 수동으로 삭제해야 합니다.

전파된 구조 매개 변수의 목록은 다음을 참조하십시오: ▶ 전파된 이전 형식 프로덕션 데이터베이스의 구조적 매개 변수 [111 페이지].

일반 개요

다음은 구조 변경 전파 프로세스에 대해 설명합니다.

4.1. 그림 구조 변경 전파 - 프로세스



1: 도구에서 이전 형식 프로덕션 데이터베이스와 표준 old-format gbbase*. * 파일의 구조 간 차이를 확인합니다.

2: 도구에서 표준 old-format gbbase*. * 파일과 표준 9.30 gbbase*. * 파일 간 차이를 확인합니다.

3: 이 도구에서 **1** 단계 및 **2** 단계에서 식별된 사항에 따라 표준 9.30 gbbase *. * 파일을 복사하고 수정합니다. 다음 규칙을 준수하면 수행됩니다.

- 버전 9.30에 표시되지 않는 표준 테이블에서 수행된 수정 사항은 손실됩니다.

- 1단계 및 2단계에서 동일한 테이블, 필드 또는 링크에 대한 수정이 감지된 경우 적용되는 수정 사항은 2단계에서 감지된 수정 사항입니다. 그리고 나서 경고 메시지가 나타납니다.

참고:

예외: 1단계 및 2단계에서 동일한 테이블, 필드 또는 링크에 대한 **이름** 또는 **설명** 수정이 감지된 경우 적용되는 수정 사항은 1단계에서 감지된 수정 사항입니다.

- 구조 변경을 한정적으로 전파하기 전에 **이전 형식 프로덕션 데이터베이스**에서 버전 9.30에 표시되는 테이블, 필드 및 인덱스의 **SQL** 이름을 수정해야 합니다. 그렇지 않으면 버전 9.30에 있는 이름이 같은 표준 필드와 충돌합니다.

사용자 지정된 9.30 gbbase*.* 파일을 명확하게 식별해야 합니다. 이렇게 하면 다음 단계에서 작업이 간편해집니다.

- 9단계 - 수동으로 변환할 응용 프로그램 데이터 내보내기 [53 페이지]
- 수동으로 변환할 응용 프로그램 데이터 처리 [56 페이지]
- 5단계 - 이전 형식 시뮬레이션 데이터베이스 변환 [43 페이지]

구조 변경 전파

- 1 Asset Manager Application Designer 버전 9.30을 시작합니다.
- 2 **Admin** 로그인을 사용하여 이전 형식 프로덕션 데이터베이스에 연결합니다(**파일/열기/기존 데이터베이스 열기** 메뉴).
- 3 **마이그레이션/사용자 지정 구조 전파** 메뉴를 선택합니다.

 참고:

이전 형식 프로덕션 데이터베이스가 다중 언어(▶ **관리 가이드의 Asset Manager 데이터베이스 생성, 수정 및 삭제 장, Asset Manager 클라이언트 언어 수정** 섹션)인 경우 마법사의 페이지 중 하나가 **이전 형식 프로덕션 데이터베이스**의 추가 언어에 적용된 사용자 지정을 전파하도록 제안합니다. 이렇게 하려면 각 추가 언어 버전에 사용할 수 있는 **Asset Manager** 버전 9.30이 필요하고 변환에 사용된 컴퓨터에서 이러한 언어로 **Asset Manager**를 설치해야 합니다.

필드 및 링크에 대한 컨텍스트 도움말을 제외하고 모든 다중 언어 요소가 전파됩니다(▶ **필드 도움말 [73 페이지]**).

자동으로 언어 X에 대한 사용자 지정을 전파하려면 이 언어로 **Asset Manager**를 사용할 수 있을 때까지 기다려야 합니다.

기다리는 대신 이미 사용 가능한 언어 버전에서 업그레이드를 수행할 수 있지만 그럴 경우 언어 X에 적용한 사용자 지정을 전파할 수 없게 됩니다.

Asset Manager 9.30을 해당 언어로 사용할 수 있게 되면 언어 X를 **9.30 형식 프로덕션 데이터베이스**에 삽입합니다. **수동으로 이전 형식 프로덕션 데이터베이스**에 적용된 사용자 지정을 전파해야 합니다.

- 4 마법사에서 지정한 지침을 따릅니다.
- 5 newdbb.log 로그 파일을 참조하십시오. 이 파일은 **생성** 폴더 필드에서 정의된 폴더에 있습니다.
- 6 메시지에서 지시할 경우 이전 형식 프로덕션 데이터베이스의 구조를 수정합니다. 그런 다음 4단계 - **이전 형식 프로덕션 데이터베이스 복사 [41 페이지]** 단계부터 마이그레이션을 수행합니다.
문제 메시지가 표시되지 않고 올바른 사용자 지정된 9.30 gbbase*.* 파일을 가져올 때까지 이 작업을 반복해야 합니다.

7

 경고:

이 단계는 **Asset Manager**의 4.3.0 버전 이상을 마이그레이션하려는 경우에는 해당되지 않습니다.

특정 스크립트가 표준 9.30 gbbase*.* 파일로 전파되지 않을 수도 있습니다.

메시지는 전파되지 않은 각 스크립트에 대해 <생성 폴더>\dbbscript 및 <생성 폴더>\bulddb\dbbscripts 폴더에 생성된 .xml 파일과 newdbb.log 로그 파일에 있습니다.

이러한 사용자 지정 내용을 사용자 지정된 9.30 gbbase*.* 파일에 수동으로 전파해야 합니다.

Asset Manager Script Analyzer에서 스크립트를 변환하려는 경우 10단계 - **응용 프로그램 데이터 검증 및 수정 [56 페이지]** 단계로 이동하여 이 작업을 수행할 수 있습니다.

Asset Manager Script Analyzer에서는 수정을 제안합니다. 이 작업은 Asset Manager Application Designer를 사용하여 사용자 지정된 9.30 gbbase*. * 파일에서 수동으로 수행합니다.

- 8 버전이 4.0.0 이상인 이전 형식 프로덕션 데이터베이스를 변환하려는 경우 추가한 각 페이지가 여전히 유효한지 Asset Manager Application Designer를 사용하여 확인합니다. 유효하지 않으면 각 페이지를 수동으로 수정해야 합니다.

 경고:

단, 5단계 - 이전 형식 시뮬레이션 데이터베이스 변환 [43 페이지] 단계를 실행할 때 사용자 지정된 9.30 gbbase*. * 파일을 다시 수정해야 합니다.

 충돌 가능성

구조 변경 전파가 비정상적으로 중단될 경우 Java 설치 폴더의 /jre/lib/ext 하위 폴더에 xerces.jar 파일이 있는지 검증합니다.

이 파일이 있으면 이 폴더를 임시로 옮기고 구조 변경 전파를 다시 실행해 봅니다.

 migration.xml 변환 파일을 분석하고 조정하여 구조 변경을 처리

 경고:

이 작업은 Asset Manager의 4.3.0 버전 이상을 마이그레이션하려는 경우에는 해당되지 않습니다.

전파된 구조 변경에 테이블 추가가 포함되는 경우 이러한 테이블의 변환을 관리하도록 migration.xml 변환 파일을 수정해야 합니다.

4단계 - 이전 형식 프로덕션 데이터베이스 복사

 일반 백업 시 발생할 수 있는 문제

DBMS 도구를 사용하여 이전 형식 프로덕션 데이터베이스를 백업하는 경우 이전 형식 프로덕션 데이터베이스의 백업은 Asset Manager Application Designer 이외의 도구를 사용하는 다음 이벤트의 추가, 수정 또는 삭제와 관련된 모든 항목에서 원본과 동일합니다.

- 색인
- 트리거
- 저장된 프로시저
- 보기

하지만 변환 프로그램에서는 이러한 구조 변경을 관리할 수 없습니다.

이전 형식 프로덕션 데이터베이스를 변환하기 전에 이러한 구조적 수정을 삭제해야 합니다.

변환 요구 사항에 맞게 백업하는 방법으로 두 가지를 제안합니다.

- DBMS 도구를 사용하여 백업하고 이 섹션에 나열된 구조 변경을 취소합니다.
- **Asset Manager Application Designer**를 사용하여 빈 데이터베이스에 이전 형식 프로덕션 데이터베이스를 백업합니다.

참고:

이전 형식 프로덕션 데이터베이스의 백업은 변환 컴퓨터를 통해 액세스할 수 있어야 합니다.

데이터베이스를 백업하는 방법을 알아 보려면 DBMS 설명서를 참조하십시오.

솔루션 1: DBMS 도구를 사용하여 이전 형식 프로덕션 데이터베이스 복사

- 1 DBMS 도구를 사용하여 이전 형식 프로덕션 데이터베이스를 복사합니다. 백업은 원래의 이전 형식 프로덕션 데이터베이스와 동일합니다.
- 2 다음에 대해 적용된 모든 수정 사항을 삭제합니다.
 - 인덱스
 - 트리거
 - 저장된 프로시저
 - 보기
- 3 **이전 형식 시뮬레이션 데이터베이스**에 대한 **Asset Manager** 연결을 생성합니다.

솔루션 2: Asset Manager Application Designer를 사용하여 빈 데이터베이스에 이전 형식 프로덕션 데이터베이스 복사

- 1 비어 있는 이전 형식 **Asset Manager** 데이터베이스를 생성합니다.
- 2 이 빈 데이터베이스에 대한 **Asset Manager** 연결을 생성합니다.
- 3 **Asset Manager Application Designer**에서 **이전 형식 프로덕션 데이터베이스**를 엽니다.
- 4 **이전 형식 프로덕션 데이터베이스**를 위에서 생성한 빈 데이터베이스로 복사합니다(**작업/빈 데이터베이스로 데이터베이스 복사** 메뉴).

이 방법은 위에 나열된 항목에 대한 모든 수정 사항을 삭제하는 경우에 유용합니다.

Asset Manager Application Designer를 사용하여 빈 데이터베이스에 이전 형식 프로덕션 데이터베이스를 백업하는 방법을 알아보려면 **관리 가이드의 테스트 데이터베이스 사용 장, 프로덕션 데이터베이스 복사** 섹션을 참조하십시오.

5 단계 - 이전 형식 시뮬레이션 데이터베이스 변환

경고:

변환 도구는 **9.30 형식 프로덕션 데이터베이스**의 구조를 수정하는 데 사용해서는 안 됩니다(테이블, 필드, 인덱스, 저장 프로시저, 트리거, 화면 등의 추가, 삭제 또는 수정).

이와 같은 수정은 마이그레이션 이후에 계획해야 합니다.

migration.xml 변환 파일 조정

경고:

이 작업은 마이그레이션에 대해 HP의 인증을 받은 기술자가 수행해야 합니다. HP는 이 조건을 엄격히 준수하지 않을 경우 모든 책임을 거부합니다.

Asset Manager 9.30은 기본적으로 변환 파일과 함께 설치됩니다(마이그레이션에서 지원되는 **Asset Manager**의 이전 버전별로 파일 1개).

이러한 파일은 **이전 형식 시뮬레이션 데이터베이스**의 변환 중에 변환할 데이터와 수행할 변환 작업에 대해 설명합니다.

이러한 변환 파일을 **migration.xml**이라고 합니다.

일반적으로 **C:\Program Files\HP\Asset Manager 9.30 xx\migration\fromxxx** 폴더에 있으며, 여기서 **xxx**는 이전 버전의 번호입니다.

표준 방식으로 **Asset Manager**를 사용할 경우 기본적으로 설치된 변환 파일 중 하나를 사용할 것입니다.

기본 기능 이외의 기능을 수행하는 필드, 추가된 테이블 및 필드 등 특별한 요구가 있을 경우에는 변환 파일을 요구에 맞게 조정해야 합니다.

경고:

표준 또는 사용자 지정된 변환 파일은 나중에 **마이그레이션 데이터베이스**에서 실행하기 전에 먼저 **시뮬레이션 데이터베이스**에서 테스트해야 합니다.

변환 파일의 구문 및 사용자 지정 방법: ▶ [migration.xml 변환 파일 조정 \[91 페이지\]](#).

이전 형식 프로덕션 데이터베이스에서 데이터를 수정할 경우 제약 조건: ▶ [2단계 - 이전 형식 프로덕션 데이터베이스의 수동 조정 \[33 페이지\]](#).

 [중요]:

migration.xml 변환 파일을 사용자 지정할 경우 파일의 이름을 변경하거나 대체해서는 안 됩니다. 이 파일을 사용하는 도구가 표준 폴더에서 검색하기 때문입니다. 또한 이 변환 파일을 수정하기 전에 변환 파일을 백업해 두는 것이 좋습니다.

 **이전 형식 프로덕션 데이터베이스가 Asset Manager 5.20 버전보다 하위 버전이고 Oracle을 사용하는 경우 필요 조건**

버전 5.20부터, Oracle을 기반으로 하는 Asset Manager 데이터베이스는 이전에 각각 **LONG** 및 **LONGRAW**를 사용하던 필드에 대해 **CLOB/BLOB** 데이터 유형을 사용합니다. 따라서 데이터베이스 변환을 진행하기 전에 이러한 필드를 식별하고 데이터 유형을 변경해야 합니다.

데이터베이스의 데이터 유형을 변경하려면 다음을 수행합니다.

- 1 Asset Manager Application Designer 버전 9.30을(를) 시작합니다.
- 2 **Admin** 로그인을 사용하여 **이전 형식 시뮬레이션 데이터베이스**에 연결합니다 (파일/열기/기존 데이터베이스 열기 메뉴).

 [중요]:

Asset Manager의 연결 세부사항:

- **소유자** 필드는 비워 두어야 합니다.
- **사용자** 필드는 데이터베이스 테이블의 **소유자**인 사용자를 참조해야 합니다 (모든 데이터베이스 개체에 대한 생성 권한).

- 3 메뉴 항목에서 **작업/템플릿/폴더 선택...**을 선택합니다.
- 4 <Asset Manager 9.30 설치 폴더>\doc\infos 폴더를 선택하고 **확인**을 클릭합니다.
- 5 메뉴 항목에서 **작업/템플릿/목록 새로 고침...**을 선택합니다.
이렇게 하면 migratelob.tpl 템플릿 파일을 기반으로 **ORACLE batch for BLOB migration**이라는 새 옵션이 **템플릿** 메뉴에 추가됩니다.
- 6 메뉴 항목에서 **작업/템플릿/ORACLE batch for BLOB migration...**을 선택합니다.
기본적으로 migratelob.sql이라는 Oracle SQL + 배치 파일이 생성됩니다. 이 파일은 **LONG** 및 **LONGRAW** 필드를 **CLOB** 및 **BLOB**로 각각 변환하는 지침을 포함합니다.
- 7 **ORACLE SQL** 같은 유틸리티와 프롬프트를 사용하여 migratelob.sql 배치 파일을 실행합니다. 예:

```
SQL> @C:\Users\encornet\AppData\Local\Temp\migratelob.sql
```

이 배치 파일은 표준 데이터베이스 업그레이드를 진행할 수 있도록 필드를 새 데이터 유형으로 변경합니다.

 [중요]:

(ODBC 연결을 통해) Asset Manager 데이터베이스에 직접 액세스하는 솔루션을 개발했다면 **이전 형식 프로덕션 데이터베이스 복사본**을 버전 9.30으로 변환한 후에는 그 솔루션이 **LONG** 및 **LONGRAW** 데이터 유형에 액세스했던 통합을 업데이트해야 합니다.

 **이전 형식 시뮬레이션 데이터베이스 변환**

이전 형식 시뮬레이션 데이터베이스를 변환하려면 다음을 수행합니다.

- 1 Asset Manager Application Designer 버전 9.30을(를) 시작합니다.
- 2 **Admin** 로그인을 사용하여 **이전 형식 시뮬레이션 데이터베이스**에 연결합니다 (파일/열기/기존 데이터베이스 열기 메뉴).

 [중요]:

Asset Manager의 연결 세부사항:

- **소유자** 필드는 비워 두어야 합니다.
- **사용자** 필드는 데이터베이스 테이블의 **소유자**인 사용자를 참조해야 합니다 (모든 데이터베이스 개체에 대한 생성 권한).
- Microsoft SQL Server에서 테이블 소유자가 **dbo**인 경우 연결 로그인을 통해 **dbo.<table>** 형식으로 기본 테이블을 생성해야 합니다(일반적으로 로그인: **sa**).

- 3 **마이그레이션/데이터베이스 변환**을 선택합니다.
- 4 마법사에서 지정한 지침을 따릅니다.

 **팁:**

사용자 유형이 **주석**인 필드를 변환하는 데는 시간이 많이 듭니다. 데이터베이스가 클 경우에는 몇 시간이 걸릴 수도 있습니다.

이 단계 동안 메시지가 나타나지 않기 때문에 변환 프로세스가 계속 실행 중인 지 궁금할 수 있습니다.

확인하려면 변환 시스템의 작업이나 데이터베이스 서버의 작업(CPU 또는 I/O)을 확인하십시오.

- 5 sdu.log 로그 파일의 메시지를 확인합니다.

경고:

이 작업은 Asset Manager의 4.3.0 버전 이상을 마이그레이션하려는 경우에는 해당되지 않습니다.

변환 중에 사소한 오류라도 발생하면 다음을 수행해야 합니다.

- 1 문제의 원인을 해결합니다.
- 2 4단계 - 이전 형식 프로덕션 데이터베이스 복사 [41 페이지] 단계부터 변환을 다시 시작합니다.

변환에 대한 정보

다음은 변환 중에 사용되는 몇 가지 규칙입니다.

팁:

다른 동작을 원하는 경우 migration.xml 변환 파일에서 해당되는 연관을 수정하십시오.

이전 형식 시뮬레이션 데이터베이스의 모든 원본 버전에 사용되는 규칙

평면도 위치

amFloorPlanPos 테이블의 레코드가 삭제됩니다.

데이터베이스의 구조 매개 변수

변환 프로그램은 선택한 9.30 gbbase*.* 데이터베이스 설명 파일에 정의된 테이블, 필드 및 인덱스의 모든 매개 변수를 적용합니다.

예를 들어, 필드의 기본값을 계산하는 스크립트가 이에 해당합니다.

필수 필드

대상 필드가

- 필수이거나 고유 값이 필요한 인덱스의 일부인 경우
- 또한 migration.xml 변환 파일에 설명된 명시적 연관이나, 필드가 동일한 SQL 이름을 공유할 때 자동으로 추론되는 암시적 연관에 포함되지 않는 경우

변환 첫 단계에 경고 메시지가 나타납니다.

이 단계는 데이터베이스를 수정하기 전에 선행되는 테스트 단계입니다.

사용자가 직접 중단하는 경우가 아니라면 변환은 중단되지 않습니다.

변환을 중단하려는 경우에는 수정되기 전에 중단해야 합니다. 그렇지 않으면 **이전 형식 시뮬레이션 데이터베이스**를 복원해야 합니다.

필요한 정보를 채워서 필수 필드를 채우고자 할 수도 있습니다. 이 정보는 이전 형식 프로덕션 데이터베이스에 있어야 합니다.

필드 기본값

프로덕션 데이터베이스의 구조에 정의된 기본값은 적용되지 않습니다.

기본값에 해당하는 값을 적용하려면 변환 파일에 이러한 값을 정의해야 합니다.



팁:

표준 migration.xml 변환 파일에는 이와 같은 작업을 수행하는 value 속성이 이미 들어 있습니다.

고유 값 인덱스

변환 파일에서는 고유 값이 준수되었는지 체계적으로 검증하지는 않습니다.

반면에 운영으로 인해 인덱스의 무결성이 손상될 경우 DBMS는 변환을 중단합니다.

value 속성의 SQL 유효성

변환 파일에서는 value 속성의 SQL 유효성도 검증하지 않습니다.

반면에 SQL 용어에 유효하지 않은 value 속성이 발견되면 DBMS는 변환을 중단합니다.

그룹화된 변환 특성

변환 운영은 거의 모든 데이터에 대해 레코드별로 수행되지 않고 **그룹화된** 방식으로 수행됩니다. 따라서 하나의 전역 SQL 명령이 하나의 전체 테이블의 레코드를 수정합니다.

수정된 테이블

수정된 테이블(여기 예에서는 **A** 테이블)의 경우 변환 도구는 다음 순서대로 진행됩니다.

- 1 **A** 테이블의 이름이 변경됩니다(여기 예에서는 **AOld**).
- 2 새 테이블이 생성됩니다(여기 예에서는 **A**).
- 3 데이터가 **AOld**에서 **A**로 전송됩니다.

Mapping 요소는 다른 동작을 정의할 수 있습니다.

- 4 **AOld**가 삭제됩니다.

따라서 지정된 테이블 **A**의 경우

A 테이블이 이전 버전에 존재합니까?	버전 9.30에 A 테이블의 존재 여부	이전 버전과 버전 9.30 간에 필드, 링크 또는 인덱스에 대한 수정이 있는지 여부	그러면 변환 프로그램은 다음을 수행합니다.
예	예	아니요	A 테이블에서 직접 작동합니다.
예	예	예	임시 AOld 테이블을 생성합니다.
아니요	예	적용되지 않음	새 테이블 A 를 생성합니다.
예	아니요	적용되지 않음	A 테이블의 데이터를 다른 테이블로 전송하고, 변환 종료 시 A 테이블을 삭제합니다.



팁:

From 속성은 **AOld** 테이블을 참조하지 않아도 됩니다. **A** 참조만으로 충분하기 때문입니다. 변환 프로그램은 **AOld**에서 정보를 찾을 시기를 알고 있습니다. 반면에 Mapping 요소 외부에서 실행되는 스크립트에서는 **A**와 **AOld**를 구분해야 합니다.



참고:

변경되지 않고 삭제되는 테이블은 변환 중에 이름이 바뀌지 않습니다.

수동으로 변환할 응용 프로그램 데이터를 저장하는 필드

수동으로 변환할 응용 프로그램 데이터를 저장하는 필드는 설명 파일에 정의된 순서대로 비워집니다.

비워진 필드가 내보낸 응용 프로그램 데이터에 해당하도록 기본적으로 설치되는 migration.xml 변환 파일이 작성됩니다.

버전이 5.10보다 낮은 이전 형식 시뮬레이션 데이터베이스에 사용되는 규칙

사용자 역할

버전 5.10에서는 사용자 역할이 도입되었습니다. 사용자가 더 이상 고유 사용자 프로필과 연관되지 않고 사용자 역할과 연관됩니다. 사용자 역할은 하나 이상의 사용자 프로필로 구성됩니다.

변환 중에는

- 사용자 역할이 이전 형식 시뮬레이션 데이터베이스에 있던 각 사용자 프로필에 대해 생성됩니다. 역할의 이름은 프로필과 동일한 이름이며, 프로필은 사용자 역할에 연결되어 있습니다. 초기에 사용자 프로필과 연관된 사용자는 이제 사용자 역할에 직접 연결됩니다.

- 사용자 프로필(amProfile) 테이블의 로그인 시 작업(LoginAction), 화면 집합(ScreenSets) 및 목록의 모든 필드 및 링크 표시 인증(bFullListCfg) 필드가 이제 사용자 역할(amMasterProfile) 테이블로 이동되었습니다.

버전이 5.0.0보다 낮은 이전 형식 시뮬레이션 데이터베이스에 사용되는 규칙

쿼리 마법사

변환 중에는

- 쿼리 마법사가 다른 마법사와 함께 변환됩니다.
- 매개 변수가 수정되지 않은 시스템 화면의 경우, **QBE 필드** 매개 변수 (Asset Manager Application Designer에서 테이블을 선택하고 화면을 표시한 다음 **목록/세부사항** 탭을 선택하여 볼 수 있음)가 **9.30 형식 시뮬레이션 데이터베이스**의 기본 매개 변수를 유지 보수하도록 업데이트됩니다.

Asset Manager 데이터베이스 테이블에 추가된 화면

테이블에 추가된 화면은 테이블을 선택하고, **보기/화면** 메뉴를 사용하고, **새로 만들기** 버튼을 클릭한 후 Asset Manager Application Designer에서 생성된 화면입니다.

이러한 화면은 값이 **아니오인 시스템 개체** 필드를 통해 식별할 수 있습니다.

변환 중에는

- 추가된 화면이 **사용자 지정** 화면 집합과 연관됩니다.
- 사용자 프로필 세부사항에서 **화면 집합(ScreenSets)** 필드의 값은 **사용자 지정, 전체, 단순**입니다.

가능한 충돌 원인

식별자

변환 중에는 테이블에서 생성된 각 레코드마다 새 ID(기본 키)가 생성됩니다.

하지만 ID의 수는 사용 중인 DBMS와 상관 없이 데이터베이스 수준에서 2^{31} 로 제한됩니다.

이 수를 초과할 경우 최종 데이터베이스가 손상됩니다.

변환 시 이 사항에 대해 어떠한 오류 메시지도 표시되지 않습니다.

따라서 변환 전에 이 수를 초과하지 않는지 스스로 검증해야 합니다.

변환 중에 생성된 최대 ID 수는 **이전 형식 시뮬레이션 데이터베이스**의 버전에 따라 다릅니다.

이 수를 초과하지 않았는지 검증하려면 다음을 수행합니다.

- 1 **이전 형식 시뮬레이션 데이터베이스**에서 가장 큰 ID(**MaxId**라고 가정)의 예상 값을 결정합니다.

이를 위해서는 아무 테이블(예: **amLocation**)에 레코드를 생성합니다. 이 새 레코드의 기본 키 값을 기록해 두십시오(**amLocation** 테이블의 **lLocald**).

 **팁:**

이 값을 보려면 이 필드를 목록에 추가하기만 하면 됩니다. 마우스 오른쪽 버튼을 클릭하고 컨텍스트 메뉴에서 **유틸리티/목록 구성**을 선택합니다.

2 MaxId가 (2^31)/8보다 작은지 검증합니다.

 **참고:**

이전 형식 시뮬레이션 데이터베이스가 버전 **4.0.0**인 경우에는 아무런 제약 조건이 없습니다.

주문 라인 브랜드

변환 중에 제품(**제품 링크**)에 연결된 주문 라인에서 **브랜드 필드(Brand)**의 값이 손실됩니다. 그 이유는 제품 자체가 브랜드에 연결되기 때문입니다.

다른 주문 라인의 **브랜드 필드(Brand)** 값은 **설명 필드(LineDesc)**에 추가됩니다.

요청 라인 브랜드

변환 중에 제품(**제품 링크**)에 연결된 요청 라인에서 **브랜드 필드(Brand)**의 값이 손실됩니다. 그 이유는 제품 자체가 브랜드에 연결되기 때문입니다.

다른 요청 라인의 **브랜드 필드(Brand)** 값은 **설명 필드(LineDesc)**에 추가됩니다.

고유한 인덱스

고유한 인덱스가 **Asset Manager**의 버전 **4.3.0**에서 특정 테이블에 추가되었습니다.

 **참고:**

이러한 새 고유 인덱스는 이전에는 사용할 수 없었던 안정된 정합성 유지 키를 제공합니다.

사용 예: **Asset Manager** 외부에서 수정하기 위해 **Asset Manager**에서 데이터를 내보낸 후 **Asset Manager**로 다시 가져오는 경우, 정합성 유지 키를 사용하여 중복 레코드를 생성하지 않고 이전 레코드를 찾아 업데이트할 수 있습니다.

결과: 특정 고유성 제약 조건이 이전 형식 데이터베이스에서 준수되지 않을 수도 있습니다.

이러한 경우가 발생할 때마다 데이터베이스 변환이 중단됩니다.

변환 프로그램에서 이러한 경우에 대한 경고 메시지를 표시하고, 충돌 목록을 제공합니다.

변환 프로그램에서 제공하는 지침을 따릅니다.

브랜드와 모델이 같지만 범주가 다른 제품

이러한 제품은 변환할 수 없습니다.

이러한 경우가 발생할 때마다 데이터베이스 변환이 중단됩니다.

변환 프로그램에서 이러한 경우에 대한 경고 메시지를 표시하고, 충돌 목록을 제공합니다.

변환 프로그램에서 제공하는 지침을 따릅니다.

6 단계 - 9.30 형식 시뮬레이션 데이터베이스의 무결성 검증

1

 **[중요]:**

이전 형식 프로덕션 데이터베이스를 백업합니다.

2 Asset Manager Application Designer 버전 9.30을 시작합니다.

3 **9.30 형식 시뮬레이션 데이터베이스**에 연결합니다(**파일/열기** 메뉴, **기존 데이터베이스 열기** 옵션).

4 데이터베이스 진단 창을 표시합니다(**작업/진단/데이터베이스 복구** 메뉴).

5 테이블 목록에서 (**모든 테이블**)을 선택합니다.

6 로그 파일의 이름과 위치를 지정합니다.

7 **레코드의 유효성 확인** 옵션을 제외한 모든 검증 옵션을 선택합니다.

8 **분석만** 옵션을 선택합니다.

9 **시작**을 클릭합니다.

10 실행 창의 메시지를 참조하십시오.

11 필요한 경우 로그 파일을 참조하십시오.

프로그램에서 문제를 표시하면 다음 운영 중 하나를 수행합니다.

1 **migration.xml** 변환 파일을 수정합니다.

2 **5단계 - 이전 형식 시뮬레이션 데이터베이스 변환 [43 페이지]** 단계부터 다시 시작합니다.

 **경고:**

앞의 두 작업은 Asset Manager 버전 4.1.0 이상에서 마이그레이션하는 사용자에게는 해당되지 않습니다.

또는

1 이전 형식 프로덕션 데이터베이스의 데이터를 수정합니다.

2 **4단계 - 이전 형식 프로덕션 데이터베이스 복사 [41 페이지]** 단계부터 다시 시작합니다.

분석 및 복구 프로그램에 대한 자세한 내용은 **관리 가이드의 데이터베이스의 진단 및 복구** 장을 참조하십시오.

7단계 - 9.30 형식 시뮬레이션 데이터베이스 확인

9.30 형식 시뮬레이션 데이터베이스를 찾아서 변환이 제대로 나타나는지 확인합니다.

특히 다음을 확인합니다.

- **9.30 형식 시뮬레이션 데이터베이스**와 **이전 형식 시뮬레이션 데이터베이스** 간 기본 테이블의 레코드 수를 비교합니다.

레코드 수가 크게 다른 경우 이 결과가 정상인지 검증합니다.

레코드 수가 크게 다르지만 정상적인 경우의 예: 라이선스 계약이 변환 중에 계약 테이블에서 삭제되었습니다. 따라서 계약 테이블의 레코드 수가 크게 줄었지만 이는 정상적인 결과입니다.

- 각 기본 테이블에서 하나 이상의 레코드에 대한 세부사항을 살펴보고 정보의 일관성이 유지되는지 확인합니다.

특히 계약과 관련하여 계약 유형(임대, 유지 보수 등)별로 하나 이상의 레코드를 살펴보아야 합니다.

자산 수준에서는 **모델** 링크와 같은 중요 링크에 각별히 주의해야 합니다.

- **Feature**와 해당 값이 제대로 변환되었으며, **Feature**별로 한 번씩 테스트를 수행하여 필드로의 **Feature** 변환 작업이 원활하게 수행되었는지 확인합니다.

비정상적인 경우가 발견되면 다음 운영 중 하나를 수행합니다.

- 1 **migration.xml** 변환 파일을 수정합니다.
- 2 **5단계 - 이전 형식 시뮬레이션 데이터베이스 변환 [43 페이지]** 단계부터 다시 시작합니다.

또는

- 1 이전 형식 프로덕션 데이터베이스의 데이터를 수정합니다.
- 2 **4단계 - 이전 형식 프로덕션 데이터베이스 복사 [41 페이지]** 단계부터 다시 시작합니다.

8단계 - 이전 형식 프로덕션 데이터베이스에 대한 특정 권한 제한

사용자가 수동으로 변환할 응용 프로그램 데이터가 포함된 테이블을 더 이상 수정할 수 없도록 이전 형식 프로덕션 데이터베이스의 사용자 권한을 수정합니다.

- 1 수동으로 변환할 응용 프로그램 데이터 목록을 결정하려면 다음을 참조하십시오. ▶ **수동으로 변환할 응용 프로그램 데이터 [108 페이지]**.

- 2 **관리/권한/사용자 권한** 메뉴를 통해 사용자 권한 목록을 표시합니다.

- 3 각 사용자 권한을 선택하고, 각 권한마다 다음을 수행합니다.
 - 1 사용자 권한에서 설명하는 개체를 모두 선택합니다.
 - 2 **생성, 삭제 및 생성 중에 입력** 권한의 선택을 취소합니다.
 - 3 **수정**을 클릭합니다.

수동으로 변환할 응용 프로그램 데이터가 이전 형식 프로덕션 데이터베이스에서 추출되므로 이 작업을 수행해야 합니다. 프로덕션 데이터베이스 백업에 대한 수정 사항은 변환 프로세스에서 복구되지 않습니다.

9단계 - 수동으로 변환할 응용 프로그램 데이터 내보내기

미리 알림

- ▶ 수동으로 변환할 응용 프로그램 데이터 [108 페이지]

팁

응용 프로그램 데이터의 양이 많을 수도 있습니다.

응용 프로그램 데이터를 내보내기 전에 **이전 형식 데이터베이스**에서 더 이상 사용되지 않는 응용 프로그램 데이터를 삭제하는 것이 좋습니다.

이렇게 하면 더 이상 사용되지 않는 응용 프로그램 데이터를 테스트하지 않아도 되므로 그만큼 시간이 절약될 수 있습니다.

수동으로 변환할 응용 프로그램 데이터 내보내기

- 1 **Asset Manager Application Designer** 버전 9.30을 시작합니다.
- 2 **Admin** 로그인을 사용하여 이전 형식 프로덕션 데이터베이스에 연결합니다(**파일/열기/기존 데이터베이스 열기** 메뉴).
- 3 **마이그레이션/응용 프로그램 데이터 내보내기** 메뉴를 선택합니다.
- 4 마법사에서 지정한 지침을 따릅니다.
- 5 **sduxprt.log** 로그 파일을 참조하십시오. 이 파일은 **작업 폴더** 필드에서 정의된 폴더에 있습니다.
- 6 생성된 **.xml** 파일(**작업 폴더** 필드에서 정의된 폴더에 있음) 트리 구조의 복사본을 만듭니다.

이 복사본은 나중에 원본 **.xml** 파일을 사용하거나 **.xml** 파일에 대한 수정 사항을 확인하려는 경우에 유용합니다.

! 내보내기 중에 준수해야 할 규칙

내보내기 도구는 다음을 수행합니다.

- 수동 변환할 응용 프로그램 데이터의 복사본을 사용자가 수동으로 조작할 수 있는 형식으로 내보냅니다.
- 변환할 응용 프로그램 데이터뿐만 아니라 이 데이터 컨텍스트에 대한 정보도 내보냅니다. 이를 통해 **Asset Manager Script Analyzer**를 사용하여 이 데이터를 보다 쉽게 업데이트할 수 있습니다.
- 응용 프로그램 데이터의 유형별로 구성된 **.xml** 파일의 트리 구조를 생성합니다. 각 **.xml** 파일은 검증할 응용 프로그램 데이터 유형을 하나 이상 포함하는 레코드에 해당합니다.
- 이전 형식 프로덕션 데이터베이스에 자신이 직접 추가한 응용 프로그램 데이터를 모두 포함시킵니다.
- 시스템 데이터를 제외시킵니다.
이 데이터는 [변환에 대한 정보 \[46 페이지\]](#) 섹션에서 설명하는 특정 방식으로 처리됩니다.
- 응용 프로그램 데이터의 테이블, 링크 및 필드가 **9.30** 데이터베이스의 구조와 일치하지 않으면 검증하지 마십시오.



팁:

이 작업은 **Asset Manager Script Analyzer**에서 수행됩니다.

- **비즈니스 라인 데이터**와 **샘플 데이터**를 포함시킵니다.

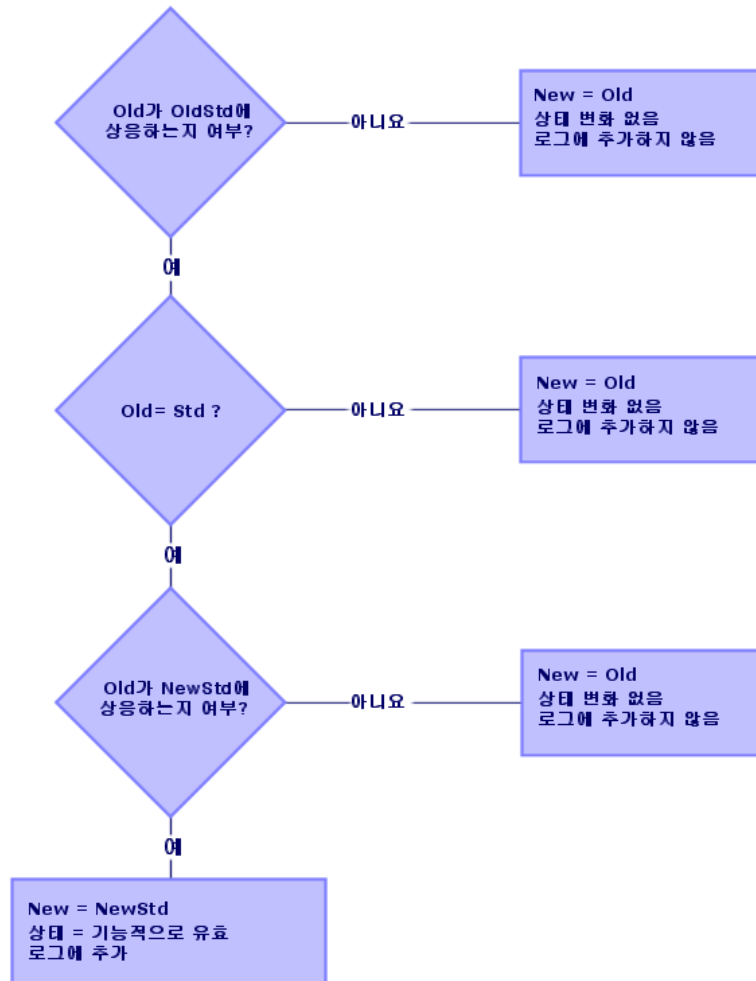
샘플 데이터는 특정 방식으로 처리됩니다.

이 특정 프로세스의 목적은 이전 형식 프로덕션 데이터베이스에서 변경되지 않은 샘플 응용 프로그램 데이터를 자동으로 업데이트하는 것입니다.

도구에서는 이 데이터를 처리하기 위해 내보낸 응용 프로그램 데이터의 각 항목을 하나씩 검사합니다.

절차는 다음과 같습니다.

4.2. 그림 샘플 데이터 처리 - 절차



정의:

- **Old:** 내보낸 응용 프로그램 데이터 항목, 즉 이전 형식 프로덕션 데이터베이스의 데이터 항목입니다.
- **OldStd:** 이전 형식의 표준 **샘플 데이터**가 있는 경우에는 **Old**에 해당합니다. 데이터 유형에 종속된 ID를 사용하여 **Old**와 **OldStd** 간 정합성 유지가 수행됩니다. 예를 들어, 작업에서는 **SQL 이름** 필드가 이에 해당합니다.

도구에서 참조로 사용하는 샘플 응용 프로그램 데이터는 **Asset Manager 9.30** 설치 폴더에 저장됩니다. 그 위치는 `\migration\fromXxx\reference` 하위 폴더이며 여기서 **Xxx**는 **Asset Manager** 이전 버전의 수입입니다.

- **NewStd: 9.30** 형식의 표준 **샘플 데이터**가 있는 경우에는 **Old**에 해당합니다. 데이터 유형에 종속된 **ID**를 사용하여 **Old**와 **NewStd** 간 정합성 유지가 수행됩니다.

도구에서 참조로 사용하는 샘플 응용 프로그램 데이터는 **Asset Manager 9.30** 설치 폴더에 저장됩니다. 그 위치는 `\migration\fromXxx\referencenew` 하위 폴더이며 여기서 **Xxx**는 **Asset Manager** 이전 버전의 수입입니다.

- **New:** 도구에서 처리된 이후의 **Old**입니다(수정되었거나 그대로 유지됨).

수동으로 변환할 응용 프로그램 데이터 처리

수동으로 변환할 응용 프로그램 데이터는 다음과 같이 여러 단계로 처리됩니다.

- 1 10단계 - 응용 프로그램 데이터 검증 및 수정 [56 페이지]
- 2 11단계 - 수정된 응용 프로그램 데이터 복원 [66 페이지]
- 3 12단계 - 9.30 형식 시뮬레이션 데이터베이스의 무결성 검증 [68 페이지]
- 4 13단계 - 복원된 응용 프로그램 데이터 검증 [68 페이지]

이 섹션에서는 이러한 단계에 대해 설명합니다.

참고:

이 섹션에서 **필드**를 검증하고 새 값으로 대체해야 한다고 설명하는 경우 이는 **Asset Manager** 데이터베이스 구조에 포함된 필드와 링크를 의미합니다.

팁:

응용 프로그램 데이터 처리를 여러 명에게 분담하여 진행할 수도 있지만 일관성 있는 프로젝트로 관리해야 합니다.

10단계 - 응용 프로그램 데이터 검증 및 수정

이 단계는 **Asset Manager Script Analyzer**를 사용하여 수행합니다.

응용 프로그램 데이터 검증 및 수정

수행해야 할 단계는 다음과 같습니다. 이러한 각 단계를 더 자세히 알아보려면 (나중에) **Asset Manager Script Analyzer**의 그래픽 레이아웃에 대한 정보를 참조하십시오.

- 1 **Asset Manager Script Analyzer**를 시작합니다.

- 2 **작업 폴더** 필드를 채웁니다.
아래의 **2**를 참조하십시오.
- 3 3단계 - 이전 형식 프로덕션 데이터베이스에 대한 구조 변경 전파 [37 페이지] 단계에서 전파되지 않은 스크립트를 포함하는 .xml 파일의 트리 구조를 생성한 경우에는
 - 1 3단계 - 이전 형식 프로덕션 데이터베이스에 대한 구조 변경 전파 [37 페이지]에서 생성된 <생성 폴더>\dbbscript 및 <생성 폴더>\bulddb\dbbscript 폴더를 복사합니다(폴더가 있는 경우).
 - 2 구조적 변경 전파: ▶ 3단계 - 이전 형식 프로덕션 데이터베이스에 대한 구조 변경 전파 [37 페이지].
 - 3 이 폴더를 **작업 폴더** 필드에 지정된 폴더로 붙여넣습니다.

- 4 **작업/모든 파일 나열** 또는 **작업/저리되지 않은 파일 나열** 메뉴를 통해 검증할 응용 프로그램 데이터 목록을 표시합니다.

메시지 창은 통합 데이터에 대해 검증할 .xml 파일 목록을 표시합니다.

6을 참조하십시오.

응용 프로그램 데이터를 내보낼 때 **Asset Manager Application Designer**는 .xml 파일에 **SQL** 이름을 자동으로 할당합니다. 기본적으로 이 이름은 접두사와 그 뒤에 자동으로 증가하는 번호로 이루어집니다.

경우에 따라 보다 정확한 이름을 사용할 수도 있습니다.

테이블의 SQL 이름	.xml 파일에 이름을 지정하는 데 사용된 필드의 SQL 이름
amAction	SQLName
amQuery	SQLName
amCalcField	SQLName
ItemNo	amFieldAdjustTempl
ItemNo	amFieldAdjust
OptSection	amOption




- 5 이 단계에서 3단계 - 이전 형식 프로덕션 데이터베이스에 대한 구조 변경 전파 [37 페이지] 단계에서 자동으로 전파되지 않은 스크립트를 처리하려는 경우에는 이러한 스크립트에 해당하는 .xml 파일부터 시작합니다.
 - 1 <생성 폴더>\dbbscript 및 <생성 폴더>\bulddb\dbbscript 폴더에서 첫 .xml 파일을 선택합니다.
 - 2 **작업/스크립트의 문제 나열** 메뉴를 통해 파일을 자세히 분석합니다.
 - 3 **메시지** 창을 참조하십시오.
13 및 **14**를 참조하십시오.
 - 4 **Asset Manager Script Analyzer**에서 제공한 수정 제안을 사용하여 3단계 - 이전 형식 프로덕션 데이터베이스에 대한 구조 변경 전파 [37 페이지] 단계에서 얻어진 사용자 지정된 9.30 gbbase*.* 파일에서 해당 스크립트를 수정합니다.

이를 위해 **Asset Manager Application Designer**를 시작하고 사용자 지정된 **9.30 gbbase.xml** 파일을 엽니다. 그런 다음 스크립트 수정을 직접 수행합니다.

- 5 .xml 파일에 대한 처리가 완료되면 **기능적으로 유효** 옵션을 선택합니다.
 - 6 **작업/처리되지 않은 파일 나열** 메뉴를 통해 목록을 표시합니다.
보고서 창에 .xml 파일 목록이 표시됩니다.
 - 7 확인할 다음 .xml 파일을 선택하고 이 파일의 세부 분석을 수행합니다.
- 6 보고서에서 검증할 다른 .xml 파일을 각각 선택합니다.
선택한 각 .xml 파일마다
- 1 파일을 자세히 분석합니다(**작업/스크립트의 문제 나열** 메뉴).

 **팁:**

동일한 .xml 파일에는 수동으로 변환할 여러 유형의 응용 프로그램 데이터가 있을 수 있습니다.

- 2 **메시지** 창을 참조하십시오.
 및 를 참조하십시오.
- 3 편집 영역에서 .xml을 직접 수정합니다: **컨텍스트 필드** 및 **탭**.
나중에 변환 프로세스에서 수정한 .xml 파일을 가져오게 됩니다.
을 참조하십시오.
- 4 해당 컨텍스트에서 스크립트를 테스트합니다(**작업/해당 컨텍스트에서 스크립트 확인** 메뉴).
이 운영의 목적은 버전 **9.30**의 데이터베이스 구조에 따라 스크립트가 유효한지 검증하는 것입니다.

 **[중요]:**

스크립트가 유효하지 않으면 **Asset Manager**에서 그래픽 인터페이스를 사용하여 열 수 없으므로, 이 운영은 스크립트 및 **SQL** 쿼리에 아주 중요합니다. 따라서 .xml 파일을 복원한 후 수정하는 것은 다소 복잡하고, 경우에 따라 불가능할 수도 있습니다.

이 운영은 작업 컨텍스트에 따라 괄호 안에 지정된 필드와 링크가 유효한지 검증합니다.

 **참고:**

이 스크립트는 현재 파일에 대해 **복원 가능** 옵션을 선택할 때 컨텍스트에서 자동으로 테스트됩니다.


 **경고:**

해당 컨텍스트에서 스크립트를 테스트하더라도 **작업/스크립트의 문제 나열** 메뉴를 사용하여 스크립트를 테스트해야 합니다. 이 도구는 스크립트의 다양한 측면을 테스트합니다.

5 **.xml** 파일 분석 및 수정이 완료되면 **복원 가능** 옵션을 선택합니다.


즉 **9.30 형식 시뮬레이션 데이터베이스**의 **.xml** 파일을 복원하여 수동으로 변환된 응용 프로그램 데이터를 테스트할 수 있습니다.

복원 가능 옵션을 선택하려 할 경우 이 스크립트는 컨텍스트에서 자동으로 테스트됩니다.

 을 참조하십시오.

6 검증할 응용 프로그램 데이터의 목록을 표시합니다(**작업/저리되지 않은 파일 나열** 메뉴).

보고서 창에 **.xml** 파일 목록이 표시됩니다.

복원 가능으로 표시된 **.xml** 파일은 **Asset Manager Script Analyzer**에서 더 이상 분석되지 않습니다. 괄호 안의 숫자가 **0**으로 설정됩니다. 파란색  아이콘은 파일이 복원 가능함을 나타냅니다.

7 확인할 다음 **.xml** 파일을 선택하고 이 파일의 세부 분석을 수행합니다.

 **참고:**

일부 응용 프로그램 데이터는 이미 **기능적으로 유효** 상태입니다.

이 데이터는 **내보내기 중에 준수해야 할 규칙 [54 페이지]** 단계의 샘플 데이터에 찾은 응용 프로그램 데이터입니다.

응용 프로그램 데이터의 수정 속도 향상

 **경고:**

여기에 설명된 팁은 다소 어려운 절차입니다.

이 작업에 완전히 익숙해졌으며 작업 전체를 담당하는 경우가 아니면 이 절차를 수행해서는 안 됩니다.

이러한 수정 작업 중 일부는 반복되어 보이기도 합니다.

따라서 **.xml** 파일 집합에 대해 검색 및 바꾸기를 수행할 수 있습니다.

다음은 주의해야 할 몇 가지 사항입니다.

- 각 단계를 수행한 후 검색 및 바꾸기를 수행하기 전에 **.xml** 파일을 백업합니다.
- 검색하는 문자열에 구분 기호를 포함시킵니다.
- 검색 시 **단어 단위로** 옵션을 선택합니다.
- 바꾸기 횟수 계산을 요청하고 숫자가 맞는지 확인합니다.

- 수정된 파일 수준에서 바꾸기 전/후의 차이를 분석합니다.

Asset Manager Script Analyzer 메뉴

4.3. 표 Asset Manager Script Analyzer - 메뉴

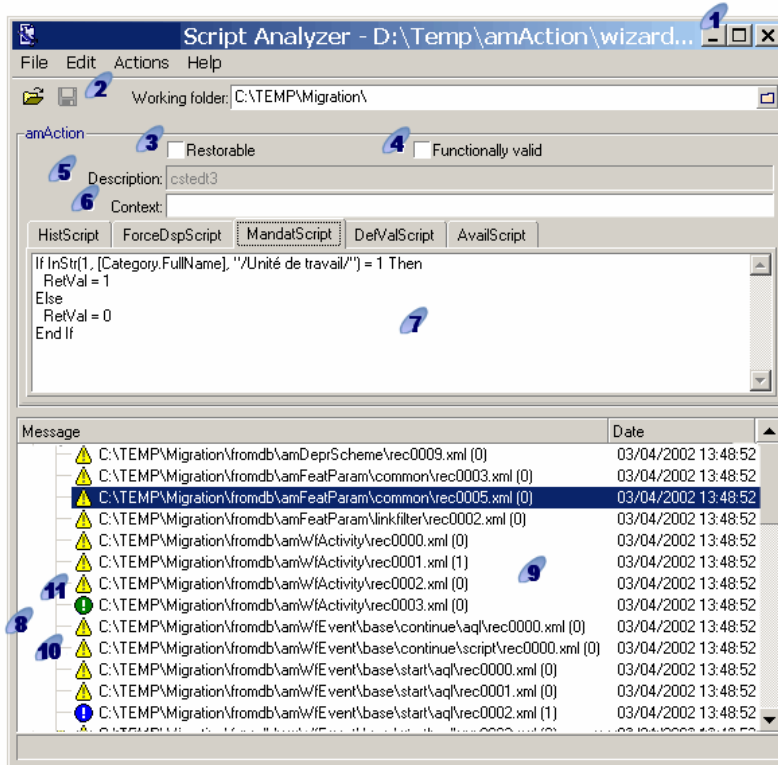
메뉴	사용
파일 메뉴	
새 기능	필요하지 않습니다.
열기	작업 폴더 필드에서 루트가 정의된 트리 구조에서 .xml 파일을 열 수 있습니다.
저장	파일에 대한 수정 사항을 저장합니다(복원 가능 또는 기능적으로 유효 문자, 컨텍스트, 스크립트).
다른 이름으로 저장	필요하지 않습니다.
종료	Asset Manager Script Analyzer를 종료합니다.
편집 메뉴	다른 편집 메뉴와 비슷하게 작동합니다.
작업 메뉴	
다음 파일 열기	메시지 창에 표시된 목록에서 다음 .xml 파일을 엽니다.
이전 파일 열기	메시지 창에 표시된 목록에서 이전 .xml 파일을 엽니다.
스크립트의 문제 나열	선택한 .xml 파일의 잠재적인 문제를 분석하여 그 결과를 메시지 창에 표시합니다.
해당 컨텍스트에서 스크립트 확인	컨텍스트 필드(채워진 경우)에서 테이블에 따라 현재 스크립트의 유효성을 테스트합니다. 그렇지 않을 경우 컨텍스트 외부에서 스크립트의 유효성을 테스트합니다.

메뉴	사용
파일에 대해 복원 가능 특성 강제 적용	<p>작업/해당 컨텍스트에서 스크립트 확인 메뉴를 통해 스크립트를 컨텍스트에서 확인하지 않았더라도 복원 가능 옵션을 선택합니다.</p> <p>경고:</p> <p>이 메뉴는 다음 경우에만 사용합니다.</p> <ul style="list-style-type: none"> ■ 작업/해당 컨텍스트에서 스크립트 확인 메뉴 사용 시 부적절한 오류가 반환됨 ■ 스크립트의 유효성을 확인할 수 있음 <p>사용 예</p> <p>일정 수준 임대 테이블(amCntrRent)에 비례 배분 기준 필드(ProrateField)가 포함되어 있습니다. 이 필드는 필드의 시스템 이름을 저장합니다. 그러나 Asset Manager Script Analyzer는 SQL 이름의 불일치 사항을 감지하는 방법만 알고 있으므로 대부분의 경우 부적절한 오류를 표시합니다.</p> <p>이 필드의 경우 다음을 수행합니다.</p> <ol style="list-style-type: none"> 1 작업/파일에 대해 복원 가능 특성 강제 적용 메뉴를 선택합니다. 2 파일을 복원합니다. 3 Asset Manager 9.30에서 파일을 테스트합니다. 4 Asset Manager Script Analyzer에서 비례 배분 기준 필드의 값을 수정합니다. 5 기능적으로 유효 옵션을 선택합니다.
처리되지 않은 파일 나열	<p>.xml 파일의 목록을 표시합니다.</p> <ul style="list-style-type: none"> ■ 작업 폴더 필드에서 루트가 정의된 트리 구조의 파일 ■ 기능적으로 유효 옵션이 선택되지 않은 파일
모든 파일 나열	<p>작업 폴더 필드에서 루트가 정의된 트리 구조에서 모든 .xml 파일의 목록을 표시합니다.</p>
응용 프로그램 데이터 복원	<p>Asset Manager 데이터베이스와의 연결을 선택하고 복원 가능 옵션이 선택된 .xml 파일을 가져올 수 있습니다. 이 메뉴는 Asset Manager Application Designer의 마이그레이션/응용 프로그램 데이터 복원 메뉴와 동일한 작업을 수행합니다.</p>

Asset Manager Script Analyzer를 통해 표시되는 .xml 파일 목록

작업/모든 파일 나열 또는 **작업/처리되지 않은 파일 나열** 메뉴를 사용하는 경우 Asset Manager Script Analyzer에서 표시하는 창은 다음과 같습니다.

4.3. 그림 Asset Manager Script Analyzer - *.xml 파일 분석 창



1 현재 .xml 파일의 전체 경로입니다.

2 Asset Manager Application Designer를 사용하여 내보낸 응용 프로그램 데이터를 포함하는 폴더입니다(수동으로 변환할 응용 프로그램 데이터를 포함하는 .xml 파일의 트리 구조).

이 파일은 Asset Manager Application Designer 메뉴의 **마이그레이션/응용 프로그램 데이터 내보내기**를 통해 **작업 폴더** 필드에 지정한 파일입니다.

또한 **modifications.xml** 파일이 있는 루트의 폴더이기도 합니다.

이 파일은 **migration.xml** 변환 파일에서 생성됩니다.

원본 데이터베이스 파일에서 순서대로 사용할 수 있는 가능한 마이그레이션을 모두 설명합니다.

modifications.xml 파일은 Asset Manager Script Analyzer에서 필드 이름에 대한 문제를 분석하는 데만 사용됩니다.

3 .xml 파일 분석 및 수정이 완료되면 **복원 가능** 옵션을 선택합니다.

4 Asset Manager 데이터베이스에서 복원된 .xml 파일의 응용 프로그램 데이터에 대한 테스트가 완료되면 **기능적으로 유효** 옵션을 선택합니다.

5 확인할 응용 프로그램 데이터를 식별하도록 도와주는 정보입니다. 이 정보는 다양하며(예: 응용 프로그램 데이터를 저장하는 레코드 이름) Asset Manager Application Designer를 사용하여 응용 프로그램 데이터를 내보내는 동안 추출할 수 있습니다.

6 응용 프로그램 데이터의 컨텍스트 테이블입니다(이 컨텍스트가 있는 경우).

경고:


작업/스크립트에 문제 나열 메뉴는 이 정보를 테스트하지 않습니다. 사용자는 컨텍스트가 항상 유효한지 확인해야 합니다(예: 9.30 버전에서 삭제된 테이블).


7 파일에 스크립트가 여러 개 포함된 경우 각 스크립트는 서로 다른 탭에 들어 있습니다. 스크립트 중 하나(modifications.xml 파일의 필드)에 문제가 발생하면 **작업/스크립트의 문제 나열** 메뉴에 메시지가 표시됩니다.

8 **작업 폴더** 필드에서 루트가 정의된 트리 구조의 .xml 파일을 나열합니다. 사용된 메뉴에 따라 이 목록은 모든 파일을 포함하거나(**작업/모든 파일 나열** 메뉴) **기능적으로 유효** 옵션이 선택되지 않은 파일만 포함할 수도 있습니다(**작업/처리되지 않은 파일 나열** 메뉴).

9 이 목록의 각 라인은 .xml 파일에 해당합니다.


괄호 안의 숫자는 검증할 필드, 테이블 또는 링크를 포함하는 .xml 파일의 라인 수에 해당합니다.

숫자가 0이고 라인이 로 시작되면 (필드의) SQL 이름 수준의 문제라는 신호가 아닙니다. 이는 파일을 정의하는 테이블 컨텍스트에서 잘못된 응용 프로그램 데이터가 파일에 포함되어 있다는 의미입니다.

이 번호가 0이고 라인이 으로 시작하는 경우에는 필드의 SQL 이름 수준에서 문제가 있음을 나타내는 것이 아니며, 데이터를 정의하는 테이블의 컨텍스트에서 잘못된 응용 프로그램 데이터가 파일에 포함되었음을 나타내는 것도 아닙니다. 이 파일은 Asset Manager 데이터베이스에서 복원하고 테스트할 수 있습니다.

참고:

파일을 열려면 파일을 한 번 클릭하십시오.

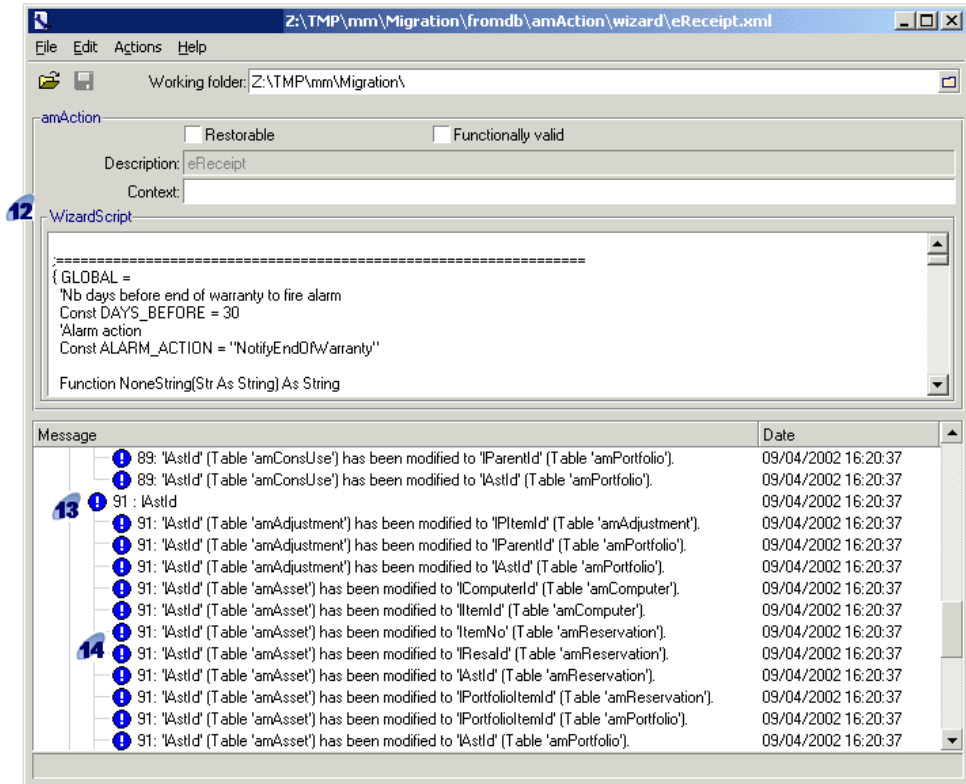
10 녹색  아이콘은 .xml 파일이 **기능적으로 유효**한 상태임을 나타냅니다.

11 파란색 1 아이콘은 .xml 파일이 **복원 가능한** 상태를 나타냅니다. 이 상태는 사용자가 수동으로 선택했거나, **작업/모든 파일 나열**과 **작업/처리되지 않은 파일 나열** 메뉴를 사용할 때 자동으로 선택됩니다. .xml 파일이 modifications.xml파일 내에 전혀 없고, 스크립트가 컨텍스트에서 확인된 경우에만 적용됩니다.

스크립트의 문제 나열

작업/스크립트의 문제 나열 메뉴를 사용하는 경우 Asset Manager Script Analyzer에 표시되는 창은 다음과 같습니다.

4.4. 그림 Asset Manager Script Analyzer - 스크립트 분석 창





참고:

메시지 창은 현재 스크립트만 분석합니다.

12 .xml 파일의 응용 프로그램 데이터가 시작된 테이블의 **SQL** 이름입니다.

13 문제가 있는 스크립트의 라인 번호와 그 뒤에 **modifications.xml** 파일에서 발견된 필드의 **SQL** 이름이 이어집니다.

확인: **modifications.xml** 변환 파일에 **SQL** 이름이 있는 필드 및 테이블

분석 프로그램에서는 필드와 링크가 속한 테이블을 고려하지 않습니다.

modifications.xml 파일에 필드 한 개의 **SQL** 이름이 표시되더라도 파일에 문제가 있는 것으로 간주할 수 있습니다.

SQL 이름이 한 테이블에서는 변경되지 않은 필드이면서 다른 테이블에서는 수정된 필드의 이름일 수도 있습니다. 프로그램에서는 사용자가 확인하여 수정할 수 있도록 도와줍니다.

modifications.xml 파일에서 테이블, 필드 및 링크의 **SQL** 이름을 검색하는 동안 _를 제외한 모든 영숫자가 구분 기호로 고려됩니다.



참고:

마우스를 두 번 클릭하면 문제가 있는 라인에 커서가 놓여집니다.

14 각 하위 라인은 수정 제안에 해당합니다.

이 창은 검증할 필드의 **SQL** 이름에 대해 가능한 수정 하나당 라인 하나를 표시합니다.

라인의 앞부분에 있는 숫자는 .xml 파일에서 검증할 라인 번호에 해당합니다.

각 제안된 수정 사항은 **modifications.xml** 파일에 설명된 연관 중 하나에서 비롯됩니다.

제안은 **modifications.xml** 파일에서 발견된 연관의 결과입니다.

몇 가지 메시지 유형은 다음과 같습니다.

- 'A'(테이블 'B')가 'C'(테이블 'D')에서 수정되었습니다. 스크립트의 A 필드가 원본 데이터베이스의 B 테이블에 포함됩니다. A 필드는 C 필드와 **modifications.xml** 파일에서 연관되었으며, 이는 사용자 지정된 **9.30 gbbase*.*** 대상 데이터베이스 설명 파일에 있는 D 테이블의 일부입니다.

예: 'script'('amAction' 테이블)가 'memScript'('amAction' 테이블)에서 수정되었습니다.

- 'A'(테이블 'B')가 더 이상 존재하지 않습니다. 스크립트의 A 필드가 원본 데이터베이스의 B 테이블에 포함됩니다. A 필드 또는 B 테이블이 사용자 지정된 **9.30 gbbase*.*** 대상 데이터베이스 설명 파일에 더 이상 포함되지 않습니다.

- 'A'(테이블 'B')가 'C'(테이블 'D')에서 수정되었습니다(수식 'E').: 스크립트의 A 필드가 원본 데이터베이스의 B 테이블에 포함됩니다. A 필드는 C 필드와 **modifications.xml** 파일에서 연관되었으며, 이는 사용자 지정된 **9.30 gbbase*.***

대상 데이터베이스 설명 파일에 있는 D 테이블의 일부입니다. C 필드는 수식 E를 사용하여 채워집니다. 수식 E는 `modifications.xml` 파일에서 검색되었습니다. 수식은 Value 속성이 필드의 SQL 이름과 다른 경우 메시지로 표시됩니다. 예(이론상): 'dtEnd'('amTicket' 테이블)이 'duration'('amTicket' 테이블)에서 수정되었습니다. (수식 'dtEnd - dtStart')

 참고:

마우스를 두 번 클릭하면 문제가 있는 라인에 커서가 놓여집니다.

 경고:

문제가 있는 테이블 이름에 대한 수정 제안이 없습니다.

11단계 - 수정된 응용 프로그램 데이터 복원

 참고:

HP Connect-It은 수정된 응용 프로그램 데이터를 복원합니다. HP Connect-It이 설치되어 있으면 이 복원 작업은 사용자에게 아무런 영향을 주지 않고 수행됩니다.

응용 프로그램 데이터의 복원은 Asset Manager Application Designer 또는 Asset Manager Script Analyzer로 수행할 수도 있습니다.

Asset Manager Application Designer를 사용하여 수정한 응용 프로그램 데이터 복원

- 1 Asset Manager Application Designer 버전 9.30을 시작합니다.
- 2 **Admin** 로그인을 사용하여 **9.30 형식 시뮬레이션 데이터베이스**에 연결합니다 (파일/열기/기존 데이터베이스 열기 메뉴).
- 3 **마이그레이션/응용 프로그램 데이터 복원** 메뉴를 선택합니다.
- 4 마법사에서 지정한 지침을 따릅니다.
- 5 `sdurest.log` 로그 파일을 참조하십시오. 이 파일은 **작업 폴더** 필드에서 정의된 폴더에 있습니다.

Asset Manager Script Analyzer를 사용하여 수정한 응용 프로그램 데이터 복원

- 1 Asset Manager Script Analyzer를 시작합니다.
- 2 **작업 폴더** 필드를 채웁니다. 이 폴더는 Asset Manager Script Analyzer를 사용하여 수정한 응용 프로그램 데이터를 포함하는 폴더입니다(스크립트를 포함하는 `.xml` 파일의 트리 구조).

- 3 **작업/응용 프로그램 데이터 복원** 메뉴를 선택합니다.
- 4 **Admin** 로그인을 사용하여 **9.30 형식 시뮬레이션 데이터베이스**에 연결합니다.
- 5 **마이그레이션 폴더** 필드를 채웁니다. 이 폴더는 변환에 필요한 참조 파일을 포함하는 폴더입니다.
데이터베이스 버전별로, 변환할 수 있는 하나의 폴더가 있습니다(C:\Program Files\HP\Asset Manager 9.30 xx\migration\fromxxx, 여기서 xxx는 이전 버전 번호).
- 6 **작업 폴더** 필드를 채웁니다. 이 폴더는 **Asset Manager Application Designer**를 사용하여 내보낸 응용 프로그램 데이터를 포함하는 폴더입니다(스크립트를 포함하는 .xml 파일의 트리 구조).
이 파일은 **Asset Manager Application Designer** 메뉴의 **마이그레이션/응용 프로그램 데이터 내보내기**를 통해 **작업 폴더** 필드에 지정한 파일입니다.
- 7 **확인**을 클릭합니다.
- 8 화면에 나타나는 메시지를 참조하십시오.
- 9 **sdurest.log** 로그 파일을 참조하십시오. 이 파일은 **작업 폴더** 필드에서 정의된 폴더에 있습니다.

거부 원인

- 복원되지 않은 상태로 선언된 .xml 파일에 저장된 응용 프로그램 데이터는 거부됩니다.
- 9.30 버전의 필수 필드는 **modifications.xml** 파일에 **Mapping** 요소가 있거나, 이전 버전 이후에 수정되지 않은 테이블에 속하거나, **modifications.xml** 파일의 **Mapping** 요소와 연관된 두 테이블 사이에 변경되지 않은 **SQL** 이름이 있어야 합니다.

팁:

필드의 필수 특성은 **Asset Manager Application Designer**에 있는 **필수** 매개 변수에 의해 정의됩니다(**Yes** 또는 **Script** 값 포함).

12단계 - 9.30 형식 시뮬레이션 데이터베이스의 무결성 검증

6단계 - 9.30 형식 시뮬레이션 데이터베이스의 무결성 검증 [51 페이지] 섹션의 지시에 따라 **9.30 형식 시뮬레이션 데이터베이스**의 무결성을 검증합니다.

이전 형식 프로덕션 데이터베이스에 연결하는 대신에 **9.30 형식 시뮬레이션 데이터베이스**에 연결해야 합니다.

복구 옵션 대신에 **분석만** 옵션을 선택합니다.

프로그램에서 문제를 표시하면 변환이 올바르게 수행되지 않았을 수도 있습니다. 그러면 변환 매개 변수, 특히 **migration.xml** 변환 파일을 검증해야 합니다.

13 단계 - 복원된 응용 프로그램 데이터 검증

프로세스

복원된 응용 프로그램 데이터는 검증된 데이터로, **Asset Manager Script Analyzer**를 통해 수정되었을 수 있습니다.

이 사실만으로는 이 응용 프로그램 데이터가 **Asset Manager**와 함께 사용될 때 확실히 작동한다고 확신할 수 없습니다.

모든 응용 프로그램 데이터를 수동으로 테스트해야만 올바르게 작동하는지 확인할 수 있습니다.

- 1 복원된 .xml 파일을 하나씩 표시합니다.
- 2 복원된 응용 프로그램 데이터를 포함하는 레코드를 찾습니다.
- 3 **9.30 형식 시뮬레이션 데이터베이스**의 응용 프로그램 데이터를 테스트합니다.

팁:

특히 데이터베이스 구조의 재구성이 수동으로 변환할 응용 프로그램 데이터를 포함하는 레코드에 영향을 주지 않는지 확인해야 합니다. (스크립트를 수정하는 것만으로는 충분하지 않습니다. 예: 자산 테이블을 사용하여 워크플로우 스키마를 재구성해야 포트폴리오 항목 테이블의 추가를 고려할 수 있습니다.)

참고:

단락 끝 표시 문자가 |로 대체됩니다.

이 경우 스크립트 실행 중에는 문제가 생성되지 않습니다.

- 4 복원된 응용 프로그램 데이터를 테스트한 경우 **Asset Manager Script Analyzer**에서 **기능적으로 유효** 옵션을 선택합니다.
즉, **9.30 형식 마이그레이션 데이터베이스**에서 .xml 파일을 복원할 수 있습니다.

주의 사항

연결 연산자

일부 **Asset Manager** 버전에서는 + 문자를 문자열 연결 연산자로 사용하는 것이 허용되었습니다.

이제 **9.30** 버전에서 이 문자는 더하기 연산자로 보다 엄격하게 해석됩니다.

이로 인해 스크립트를 테스트할 때 **Asset Manager**에서 오류가 발생할 수도 있습니다.

이 경우 + 연산자를 &로 대체합니다.

쿼리

쿼리에서 기본 키 값에 의해 연결된 레코드를 식별한 경우, 그리고 이 테이블의 레코드가 변환 중에 다른 인덱스를 사용하여 새 테이블로 이동된 경우 쿼리에서는 더 이상 올바른 링크를 선택하지 않습니다.

다음 수정 절차 중 하나를 수행합니다.

- 쿼리에서 기본 **ID**를 수정합니다.
- 변환을 활용하여 쿼리가 더 안정적인 필드의 값을 가리키게 합니다. 이렇게 하면 회사에서 나중에 다른 변환 중에 동일한 문제가 다시 발생하는 것을 방지할 수 있습니다.

5 단계별 마이그레이션 - 최종 변환(마이그레이션 데이터베이스)

이 단계에는 다음 항목이 준비되어 있습니다.

- 사용자 지정된 **9.30 gbbase*.*** 파일 집합
 - ▶ 3단계 - 이전 형식 프로덕션 데이터베이스에 대한 구조 변경 전파 [37 페이지]
- **시뮬레이션 데이터베이스**에서 테스트한 **migration.xml** 변환 파일
- **9.30 형식 시뮬레이션 데이터베이스**에서 테스트한 수동으로 변환된 응용 프로그램 데이터

이 장에서는 **9.30 형식 프로덕션 데이터베이스**를 변환하기 위해 수행할 운영을 단계별로 설명합니다.

16단계 - 이전 형식 프로덕션 데이터베이스 차단 및 복사

이전 형식 프로덕션 데이터베이스 차단은 작업이 손실될 수도 있으므로 변환 중에 수정 작업을 수행할 수 없도록 사용을 금지하는 작업으로 구성됩니다.

다음 작업을 수행합니다.

- 1 이전 형식 프로덕션 데이터베이스에서 모든 사용자의 연결을 해제합니다.
- 2 다음을 종료합니다.
 - Asset Manager Automated Process Manager
 - Asset Manager APIs
 - 이전 형식 프로덕션 데이터베이스에 액세스하는 외부 프로그램.
- 3 이전 형식 프로덕션 데이터베이스에 대한 액세스를 차단합니다.

4 4단계 - 이전 형식 프로덕션 데이터베이스 복사 [41 페이지] 섹션에 설명된 대로
이전 형식 프로덕션 데이터베이스를 백업합니다.

이 이전 형식 프로덕션 데이터베이스의 백업을 **마이그레이션 데이터베이스**라
고 합니다.

사용자에게 문제가 발생되지 않도록 이전 형식 프로덕션 데이터베이스가 차단되는
시간을 최소화해야 합니다.

이러한 이유로 모든 문제를 해결하기 위해 실제로 변환하기에 앞서 시간을 들여 침
착하게 시뮬레이션해야 합니다.

20단계 - 9.30 형식 마이그레이션 데이터베이스 완료

다음과 같은 여러 가지 이유로 **9.30 형식 마이그레이션 데이터베이스**를 변경할 필
요가 있습니다.

- 일부 데이터는 변환 프로그램을 통해 변환되지 않습니다.

9.30 형식 마이그레이션 데이터베이스에서 특정 데이터를 테스트하고 수동으
로 변경해야 합니다.

- 일부 기능이 추가되거나 향상되었습니다.

이를 충분히 활용하려면 **9.30 형식 마이그레이션 데이터베이스**에서 이러한 기
능의 사용을 준비해야 합니다.

이 작업을 통해 **Asset Manager**에서 수행되는 서비스와 효율성을 개선할 수 있
습니다.

이전 형식 프로덕션 데이터베이스의 모든 버전과 관련된 완료 작업

변환 성공 검증

변환이 제대로 수행되었는지 검증하는 것이 좋습니다.

예를 들어, 다음을 수행할 수 있습니다.

- **9.30 형식 마이그레이션 데이터베이스**에서 비정상적인 사항을 검색합니다.
- 특정 테이블의 변환 전/후 레코드 수를 비교합니다.

차이가 있는 경우 이는 **migration.xml** 변환 파일의 고의적인 사양이거나, 정상
이 아닌 경우입니다.

저장된 프로시저 **up_GetCounterVal**에 적용된 수정 사항

이 섹션은 이전 형식 프로덕션 데이터베이스의 저장된 프로시저
up_GetCounterVal을 수정한 사용자를 위한 것입니다.

이전 형식 프로덕션 데이터베이스를 변환하기 전에 먼저 다음을 수행해야 합니다.

- 1 다른 테이블에 전환된 **amCounter** 테이블의 카운터를 수동으로 업데이트합니
다.

2 저장된 프로시저 **up_GetCounterVal**을 원래 상태로 복원합니다.

다음 기술 노트의 지시문에 따라 저장된 프로시저 **up_GetCounterVal**을 다시 적용할 수 있습니다.

- Microsoft SQL Server: TN317171736
- Oracle Database Server: TN12516652
- DB2 UDB: TN1029175140(Asset Manager용 버전 3.x)

트리거, 인덱스, 저장된 프로시저 및 보기

변환하기 전에 이러한 항목의 수정과 관련된 모든 부분에 대해 이전 형식 프로덕션 데이터베이스를 원래 상태로 돌려놓습니다.

이제 이러한 수정이 여전히 필요할 경우 수동으로 다시 수행할 수 있습니다.

필드 도움말

필드(및 링크) 도움말은 **필드 도움말** 테이블(amHelp)에 저장됩니다.

이전 형식 마이그레이션 데이터베이스를 변환하는 동안 이 테이블의 내용은 수정되지 않습니다.

이전 버전의 필드 도움말에서 수행된 사용자 지정 저장

- 1 필드 도움말을 있는 그대로 내보냅니다.
 - 1 Asset Manager 9.30을 시작합니다.
 - 2 **9.30 형식 마이그레이션 데이터베이스**에 연결합니다(**파일 / 데이터베이스에 연결** 메뉴).
 - 3 **필드 도움말**에서 레코드 목록을 표시합니다(**관리/화면 목록** 메뉴).
 - 4 필드와 링크가 아래 표시된 순서대로 나타나도록 목록을 구성합니다.
 - 테이블(TableName)
 - 필드(FieldName)
 - 설명
 - 예
 - 주의
 - 5 목록의 내용을 내보냅니다(**목록 내보내기** 바로 가기 메뉴).
- 2 이전 버전의 표준 필드 도움말을 내보냅니다.
 - 1 선택한 DBMS를 사용하여 빈 데이터베이스를 생성합니다.
빈 데이터베이스를 생성하는 방법을 알아보려면 **관리 가이드의 Asset Manager 데이터베이스 생성, 수정 및 삭제 장, DBMS를 사용하여 공백 데이터베이스 생성** 섹션을 참조하십시오.
 - 2 이전 버전의 Asset Manager를 시작합니다.
 - 3 빈 데이터베이스에 연결합니다(**파일/데이터베이스에 연결** 메뉴).
 - 4 **필드 도움말**에서 레코드 목록을 표시합니다(**관리/화면 목록** 메뉴).
 - 5 필드와 링크가 아래 표시된 순서대로 나타나도록 목록을 구성합니다.

- 테이블(TableName)
- 필드(FieldName)
- 설명
- 예
- 주의

6 목록의 내용을 내보냅니다(**목록 내보내기** 바로 가기 메뉴).

- 3 내보낸 두 파일을 비교합니다.
두 파일의 차이가 수정 사항에 해당합니다.
이러한 수정 사항의 복사본을 보존합니다.

버전 9.30에서 필드 도움말 업데이트

- 1 Asset Manager Application Designer를 시작합니다.
- 2 **파일/열기** 메뉴를 선택합니다.
- 3 **데이터베이스 설명 파일 열기 - 새 데이터베이스 생성** 옵션을 선택합니다.
- 4 Asset Manager 9.30 설치 폴더의 config 하위 폴더에 있는 표준 9.30gbbase.xml 파일을 선택합니다.
- 5 데이터베이스 생성 마법사를 시작합니다(**작업/데이터베이스 생성** 메뉴).
- 6 다음과 같이 마법사 페이지를 채웁니다(**다음** 및 **이전** 버튼을 사용하여 마법사 페이지 탐색).

SQL 스크립트 생성/데이터베이스 생성 페이지:

필드	값
데이터베이스	9.30 형식 마이그레이션 데이터베이스 에 대한 연결을 선택합니다.
생성	비즈니스 라인 데이터를 가져옵니다.
고급 생성 옵션 사용	이 옵션을 선택합니다.

생성 매개 변수 페이지:

필드	값
암호	관리자 암호입니다.
	참고: Asset Manager 데이터베이스 관리자는 이름(Name) 필드가 Admin 으로 설정된 직원 및 부서(amEmplDept) 테이블의 레코드입니다. 사용자 이름(UserLogin) 필드에 저장된 데이터베이스 연결 로그인입니다. 관리 이름은 Admin 입니다.

시스템 데이터 생성 페이지:

필드	값
표준 시간대 사용	이 옵션은 선택하지 마십시오.
필드 도움말 사용	이 옵션을 선택합니다.

가져올 데이터 페이지:

필드	값
사용 가능한 데이터	데이터를 선택하지 마십시오.
오류가 발생하면 가져오기 중지	이 옵션은 선택하지 마십시오.
로그 파일	이 필드를 채우지 마십시오.

- 7 마법사를 사용하여 정의한 옵션을 실행합니다(**마침** 버튼).
- 8 **데이터베이스 생성** 페이지에서 메시지를 살펴보고 나서 **확인**을 클릭하여 닫습니다.

필드 도움말에 사용자 지정 다시 적용

버전 9.30의 필드 도움말을 업데이트하여 이미 수행된 사용자 지정을 덮어쓰려고 합니다.

따라서 이전 버전의 필드 도움말의 이러한 사용자 지정을 저장한 복사본을 사용하여 이 사용자 지정을 수동으로 다시 실행할 수 있습니다.

예를 들어, **테이블** 및 **필드** 필드(TableName 및 FieldName)를 정합성 유지 키로 사용하여 수정 사항을 가져옵니다.

사용자 권한, 액세스 제한 및 기능 권한

새 테이블, 필드 및 링크가 새 데이터베이스 구조에 추가되었으므로 사용자 프로필의 사용자 권한, 액세스 제한 및 기능 권한을 조정해야 합니다.

새 테이블, 필드 및 링크를 기존 권한과 제한에 추가하고, 필요한 경우 새 권한과 제한을 생성합니다.

특정 Feature를 필드로 전송

Asset Manager 9.30에서는 새 필드가 고유의 표준 9.30 형식 데이터베이스 구조에서 얻어진 것이든 사용자 지정에서 얻어진 것이든 상관 없이 새 필드에 액세스할 수 있습니다.

이전 형식 프로덕션 데이터베이스에 사용된 Feature 대신에 이러한 새 필드 중 하나를 사용할 수도 있습니다.

이는 Feature를 광범위하게 사용하는 경우에만 유용합니다.

장점

- 세부사항 창의 Feature의 경우보다 쉽게 필드를 배치할 수 있습니다.
- 액세스 제한은 Feature에서보다 필드에서 더 많이 수행됩니다.



팁:

액세스 제한은 링크와 Feature에서 똑같이 수행됩니다.

단점

- Feature 매개 변수의 **사용 가능** 필드(seAvailable)에는 필드 수준에 상응하는 항목이 없습니다.
- Feature와 달리 이 필드는 클래스와 연관시킬 수 없습니다.

절차

이 절차에 대한 자세한 내용은 **관리** 가이드를 참조하십시오.

보기

모든 공유 보기에 대한 화면이 생성되었습니다.

더 이상 사용하지 않는 보기는 삭제할 수 있습니다.

이전 형식 프로덕션 데이터베이스의 4.4.x 이하 버전과 관련된 완료 작업

시스템 항목별 목록에 의해 채워진 필드에 대한 변경 이력 유지

Asset Manager 버전 5.00부터는 **이력**(amHistory) 테이블의 **이전 값**(PreviousVal)과 **새 값**(NewVal) 필드가 데이터베이스에 저장된 값이 아니라 시스템 항목별 값 입력 필드에 표시된 값을 저장합니다.

예: **작업 주문**(amWorkOrder) 테이블에서 **상태**(seStatus) 필드는 시스템 항목별 목록을 통해 채워집니다. 이 항목별 목록의 항목 중 하나는 **알림**으로 표시되고 **0**으로 저장됩니다.

이전 값과 **새 값** 필드는 **알림**을 저장하지만 **0**은 저장하지 않습니다.

이전 버전에서는 데이터베이스에 저장된 값이 사용되었습니다.

5.00 이전 버전의 데이터베이스를 버전 9.30으로 변환하면 **이전 값**과 **새 값** 필드에 시스템 항목별 목록의 저장된 값과 표시된 값이 모두 포함됩니다.

이전 값과 **새 값** 필드를 참조하는 쿼리, 마법사 등을 그에 맞게 수정해야 합니다.

수령 대기 중인 포트폴리오 항목을 검색하는 **쿼리 예**입니다. 여기서는 **포트폴리오 항목(amPortfolio)** 테이블의 **할당(seAssignment)** 필드에 대한 이력이 유지된다고 가정합니다.

변환 전 쿼리는 다음과 같습니다.

```
seAssignment=3 or exists (SELECT 'x' FROM amHistory WHERE ((amPortfolio:IPortfolioItemId = IHistObjId) AND (PreviousVal = '3.0000')) AND (Field = 'seAssignment'))
```

데이터베이스를 변환한 후 작동을 위해 수정한 쿼리는 다음과 같습니다.

```
(seAssignment = 3) OR ( exists ((SELECT 'x' FROM amHistory WHERE ((amPortfolio:IPortfolioItemId = IHistObjId) AND ((PreviousVal = '3.0000') OR (PreviousVal = 'Awaiting Receipt')))) AND (Field = 'seAssignment'))))
```

이전 형식 프로덕션 데이터베이스의 4.1.x 이하 버전과 관련된 완료 작업

가져오기 스크립트

생성했으며 유지하려는 각 가져오기 스크립트를 하나씩 테스트해야 합니다.

- 1 Asset Manager 버전 9.30을 시작합니다.
- 2 테스트 데이터베이스(9.30 형식 마이그레이션 데이터베이스의 백업일 수 있음)에 연결합니다.
- 3 가져오기 모듈을 시작합니다(파일/가져오기 메뉴).
- 4 데이터베이스 가져오기 옵션을 선택합니다.
- 5 텍스트 탭을 선택하고 열기를 클릭합니다.
- 6 표시되는 새 창에 스크립트를 엽니다(파일/스크립트 열기 메뉴).
- 7 오른쪽 목록에 있는 (원본, 대상) 쌍을 두 번 클릭하여 한 번에 하나씩 각 연관을 검증합니다.
- 8 수정 사항을 저장합니다(파일/저장).
- 9 가져오기를 테스트합니다(가져오기 버튼).
- 10 필요한 경우 가져오기 스크립트를 다시 수정합니다.

내보내기 스크립트

생성했으며 유지하려는 각 내보내기 스크립트를 테스트해야 합니다.

- 1 Asset Manager Export Tool 9.30을 시작합니다.

- 2 **9.30 형식 마이그레이션 데이터베이스**에 연결합니다. 내보내기에서는 연결 중인 데이터베이스의 데이터를 수정하지 않습니다.
- 3 스크립트를 엽니다(**파일/스크립트 열기** 메뉴).
- 4 한 번에 하나씩 각 쿼리를 검증합니다.
 - 1 상단 목록에서 쿼리를 선택합니다.
 - 2 하단 목록에 있는 **돋보기** 아이콘을 클릭합니다.
 - 3 쿼리가 유효할 경우 경고 메시지가 나타나지 않습니다.
 - 4 쿼리가 유효하지 않은 경우 경고 메시지가 나타납니다.
 - 5 경고 메시지 표시 여부와 관계없이, 쿼리 매개 변수가 여전히 원하는 내용에 해당하는지 확인해야 합니다(데이터베이스 구조가 변경되었음을 고려). 예: 자산 테이블에서 검색 중이던 데이터는 이제 포트폴리오 항목 테이블에 있습니다.
- 5 수정 사항을 저장합니다(**파일/스크립트 저장**).
- 6 내보내기를 테스트합니다(**작업/스크립트 실행**).
- 7 필요한 경우 내보내기 스크립트를 다시 수정합니다.

보기

이전 형식 마이그레이션 데이터베이스를 변환하더라도 보기는 그대로 유지됩니다. 따라서 데이터베이스 구조에 대한 변경 사항은 복구되지 않습니다.

보기가 표시할 열과 적용된 필터를 기억하므로 보기를 하나씩 표시하면서 검증해야 합니다. 각 보기마다 표시하려는 열과 적용된 필터의 선택을 확인합니다.

- 1 **Asset Manager**를 시작합니다.
- 2 한 번에 하나씩 각 보기를 선택합니다(**도구/보기** 메뉴).
- 3 경고가 나타나면 경고 메시지를 읽고 메시지에 따라 보기를 수정합니다.

팁:

필요한 새 보기를 모두 생성합니다.

SAP Crystal Reports

이전 형식 마이그레이션 데이터베이스의 변환 중에는 보고서가 그대로 유지됩니다.

따라서 데이터베이스 구조에 대한 변경 사항은 복구되지 않습니다.

테이블, 필드 및 링크의 **SQL** 이름 여러 개가 더 이상 유효하지 않게 될 수도 있습니다.

이전 보고서 다시 사용

- 1 **Asset Manager**를 시작합니다.
- 2 보고서 목록을 표시합니다(**도구/보고/보고서** 메뉴).

- 3 더 이상 보관하지 않을 보고서를 삭제합니다.
- 4 유지할 각 보고서를 한 번에 하나씩 테스트합니다.
 - 각 보고서의 경우
 - 1 해당 보고서의 컨텍스트에 커서를 놓습니다(예: 자산 목록 또는 세부사항).
 - 2 보고서 인쇄 화면을 표시합니다(**파일/인쇄**).
 - 3 테스트할 보고서 유형에 따라 **유형** 필드를 채웁니다.
 - 4 보고서를 선택합니다.
 - 5 **미리보기**를 클릭합니다.
 - 6 경고가 나타나면 경고 메시지를 읽고 메시지에 따라 SAP Crystal Reports 에서 보고서를 수정합니다.
- 5 Asset Manager 9.30과 함께 제공된 새로운 표준 보고서를 가져오려는 경우: 새 보고서를 가져오기 전에 유지할 이전 보고서의 **SQL** 이름을 수정합니다.

 **경고:**

이 작업을 수행하지 않으면 SQL 이름이 동일한 새 보고서가 이전 보고서를 덮어쓰게 됩니다.

이전 보고서를 사용하지 않기로 결정

- 1 Asset Manager를 시작합니다.
- 2 보고서 목록을 표시합니다(**도구/보고/보고서** 메뉴).
- 3 이전 보고서를 모두 삭제합니다.

Asset Manager 9.30에서 제공하는 표준 보고서 가져오기

9.30 형식 마이그레이션 데이터베이스에서 **샘플 데이터** 보고서를 가져오려면 다음을 수행합니다.

- 1 Asset Manager Application Designer를 시작합니다.
- 2 **파일/열기** 메뉴를 선택합니다.
- 3 **데이터베이스 설명 파일 열기 - 새 데이터베이스 생성** 옵션을 선택합니다.
- 4 Asset Manager 9.30 설치 폴더의 config 하위 폴더에 있는 표준 9.30gbase.xml 파일을 선택합니다.
- 5 데이터베이스 생성 마법사를 시작합니다(**작업/데이터베이스 생성**).
- 6 다음과 같이 마법사 페이지를 채웁니다(**다음** 및 **이전** 버튼을 사용하여 마법사 페이지 탐색).

SQL 스크립트 생성/데이터베이스 생성 페이지:

필드	값
데이터베이스	보고서를 가져오려는 대상 데이터베이스에 대한 연결을 선택합니다.
생성	비즈니스 라인 데이터를 가져옵니다.
고급 생성 옵션 사용	이 옵션은 선택하지 마십시오.

생성 매개 변수 페이지:

필드	값
암호	관리자 암호를 입력합니다. 참고: Asset Manager 데이터베이스 관리자는 이름(Name) 필드가 Admin 으로 설정된 직원 및 부서(amEmplDept) 테이블의 레코드입니다. 사용자 이름(UserLogin) 필드에 저장된 데이터베이스 연결 로그인입니다. 관리 이름은 Admin 입니다. 암호는 암호 필드(LoginPassword)에 저장됩니다.

가져올 데이터 페이지:

필드	값
사용 가능한 데이터	Crystal Reports 옵션을 선택합니다.
오류가 발생하면 가져오기 중지	문제가 발생한 경우 가져오기를 중지하려면 이 옵션을 선택합니다.
로그 파일	오류 및 경고를 포함하여 모든 가져오기 운영이 기록되는 파일의 전체 이름입니다.

7 마법사를 사용하여 정의한 옵션을 실행합니다(**마침** 버튼).

자격 카운터

이전 형식 프로덕션 데이터베이스의 4.1.x 이하 버전에서는 카운터 세부사항의 **권한** 탭에 자격 개수가 정의되었습니다.

Asset Manager 버전 5.00 이상에서는 새로운 **자격** 탭을 사용합니다.

카운터가 이 새로운 기능과 일관성을 유지하도록 하려면 **권한** 탭의 정보를 **자격** 탭으로 전송하는 것이 좋습니다.

21 단계 - Asset Manager 데이터베이스에 액세스하는 외부 소프트웨어 구성 요소 업그레이드

Asset Manager Web

Asset Manager Web의 이전 버전을 제거하고 9.30 버전을 설치해야 합니다.

Asset Manager Web 5.0.0은 완전히 새로 설계되었습니다. 이제 웹 클라이언트 화면은 몇 가지 관리 화면을 제외하고는 Windows 클라이언트의 화면과 동일합니다.

Windows 클라이언트의 경우와 같이 Asset Manager Application Designer는 웹 클라이언트 화면을 사용자 지정하는 데 사용됩니다.

Asset Manager Web의 이전 버전에서 수행된 모든 사용자 지정은 손실됩니다.

Get-It

Asset Manager 9.30 데이터베이스에서 실행되도록 Get-It을 통해 개발된 각 웹 응용 프로그램에 대해 다음을 수행합니다.

- 1 해당 Get-It 버전이 Asset Manager 9.30 Support Matrix(HP 고객 지원 웹 사이트에서 사용 가능)에 있는지 확인합니다.
- 2 필요한 경우 Get-It을 업그레이드합니다.
- 3 사용자 지정 웹 페이지를 테스트하고 서로 적용합니다.

Get-Resources

Asset Manager 9.30 데이터베이스에서 Get-Resources를 실행하려면 다음을 수행합니다.

- 1 해당 Get-Resources 버전이 Asset Manager 9.30 Support Matrix(HP 고객 지원 웹 사이트에서 사용 가능)에 있는지 확인합니다.
- 2 필요한 경우 Get-Resources를 업그레이드합니다.

Get-Resources의 표준 페이지만 사용하는 경우 이 운영으로 충분합니다.

Get-Resources의 새 표준 페이지를 사용할 수 있습니다.

추가 웹 페이지 또는 사용자 지정 표준 웹 페이지를 만든 경우:

- 1 이전의 추가 또는 사용자 지정 페이지를 저장합니다.
- 2 필요한 경우 Get-Resources를 업그레이드합니다.
- 3 사용자 지정 웹 페이지를 테스트하고 서로 적용합니다.

HP Connect-It 시나리오

HP Connect-It을 사용하여 9.30 형식 마이그레이션 데이터베이스에 액세스하려면 Asset Manager 9.30과 함께 제공된 HP Connect-It 버전을 사용해야 합니다.

표준 **HP Connect-It** 시나리오를 사용하는 경우 이제 새 표준 시나리오를 사용해야 합니다.

자체 시나리오를 생성한 경우 다음을 수행합니다.

- 1 이전 비표준 시나리오를 저장합니다.
- 2 **HP Connect-It**을 업그레이드합니다.
- 3 **HP Connect-It**에서 각 시나리오를 하나씩 엽니다.
- 4 각 시나리오에 대해 다음을 수행합니다.
 - 1 시나리오를 열 때 **HP Connect-It**에 의해 표시될 수 있는 경고 메시지를 조사합니다.
 - 2 경고 메시지에 따라 시나리오를 수정합니다.
 - 3 테스트 데이터를 사용하여 시나리오를 실행합니다.
 - 4 이 테스트를 수행하는 동안 나타나는 문제를 수정합니다.

6 단계별 마이그레이션 - 최종 단계

이 장에서는 **9.30 형식 마이그레이션 데이터베이스**를 설정하고 실행하기 위해 수행할 운영을 단계별로 설명합니다.

22단계 - Asset Manager 프로그램 업그레이드

모든 관리자 및 사용자 시스템에서 모든 Asset Manager 프로그램을 업그레이드해야 합니다.

또한 Asset Manager와 상호 작용하는 프로그램 버전이 Asset Manager 9.30과 계속해서 호환 가능한지 확인해야 합니다. 필요한 경우 이러한 프로그램도 업그레이드합니다.

Asset Manager 프로그램 및 Asset Manager와 상호 작용하는 기타 프로그램의 목록을 보려면 **설치 및 업그레이드** 가이드의 **Asset Manager 구성 요소** 장을 참조하십시오.

Asset Manager 9.30과 호환 가능한 프로그램 버전에 대해서는 Support Matrix 사이트(www.hp.com/go/hpsupport)를 참조하십시오..

관리 시스템에 Asset Manager Automated Process Manager 설치

Asset Manager Automated Process Manager는 Asset Manager 데이터베이스에서 여러 자동 작업을 수행합니다. Asset Manager Automated Process Manager가 시작되지 않으면 Asset Manager가 제대로 작동하지 않습니다.

따라서 다음을 수행해야 합니다.

- 1 클라이언트 시스템에 Asset Manager Automated Process Manager를 설치합니다.
- 2 Asset Manager Automated Process Manager를 적절히 구성합니다.
- 3 Asset Manager Automated Process Manager를 영구적으로 실행합니다.

Asset Manager Automated Process Manager의 작동 방법에 대한 자세한 내용은 **관리** 가이드의 **Asset Manager Automated Process Manager** 장을 참조하십시오.

9.30 형식 마이그레이션 데이터베이스에서 Asset Manager 캐시 삭제

9.30 형식 마이그레이션 데이터베이스에 연결된 상태로 캐시를 사용할 경우에는 이 캐시를 삭제하는 것이 좋습니다.

캐시가 작동하는 방법에 대한 자세한 내용은 **사용자 인터페이스** 가이드의 **참조 정보** 장, **연결** 섹션, **Asset Manager 성능** 하위 섹션을 참조하십시오.

Asset Manager 프로그램 업그레이드

프로그램을 업그레이드하려면 다음을 수행합니다.

- 1 이전 버전의 Asset Manager를 제거합니다.

 **팁:**

변환 시스템에 Asset Manager 9.30을 설치하는 경우 당분간은 이전 버전의 Asset Manager를 보존해야 합니다.

설치 제거 절차(보호 조치, 수행할 단계 및 Asset Manager 제거 방법)에 대한 자세한 내용은 제거할 Asset Manager 버전에 해당하는 **설치 및 업그레이드** 가이드를 참조하십시오.

- 2 Asset Manager 9.30을 설치합니다.

설치 절차(보호 조치, 수행할 단계 및 Asset Manager 설치 방법)에 대한 자세한 내용은 \$SoftName; 9.30 **설치 및 업그레이드** 가이드를 참조하십시오.

문제 없이 Asset Manager를 시작할 수 있는지 확인

Asset Manager 9.30을 시작할 때 문제가 발생하면 사용자 지원에 문의하십시오.

데이터베이스에 대한 이전 연결 제거 및 새 연결 생성

목표는 사용자가 **9.30 형식 마이그레이션 데이터베이스**에 연결되는 것입니다.

사용자 인터페이스 가이드의 **참조 정보** 장, **연결** 섹션을 참조하십시오.

원하는 경우 이전 연결을 수정할 수 있습니다.

유용하게 사용할 수 있다고 판단되는 경우 연결에 대한 Asset Manager 캐시를 생성합니다.

클라이언트 시스템 수준에서 Asset Manager 사용자 지정 수정(이 작업이 유용하다고 생각될 경우)

조정(Tailoring) 가이드의 파트 1 **클라이언트 컴퓨터 사용자 지정, 클라이언트 워크스테이션 사용자 지정** 장을 참조하십시오.

23 단계 - 9.30 형식 마이그레이션 데이터베이스를 프로덕션에 배치

이 단계는 마이그레이션 프로세스의 마지막 단계입니다.

현재까지의 작업 상태는 다음과 같습니다.

- **이전 형식 마이그레이션 데이터베이스** 전체를 변환했으며 **9.30 형식 마이그레이션 데이터베이스**를 미세 조정했습니다.
- 모든 사용자 및 관리자 시스템에서 Asset Manager 프로그램을 업그레이드했습니다.

이제 다음 작업을 수행해야 합니다.

- 1 완료된 **9.30 형식 마이그레이션 데이터베이스**에서 Asset Manager Automated Process Manager를 프로덕션에 배치합니다.
- 2 **9.30 형식 마이그레이션 데이터베이스**에 액세스하는 외부 프로그램을 다시 시작합니다.
- 3 사용자에게 데이터베이스를 사용할 수 있음을 알립니다.

24 단계 - 더 이상 사용되지 않는 프로그램 설치 제거

마이그레이션 프로세스를 시작할 때 변환 컴퓨터에 특정 프로그램을 설치했습니다 (▶ **변환 컴퓨터 준비 [112 페이지]** 참조).

변환한 후 일정 기간 동안 변환 컴퓨터에서 다음 소프트웨어를 유지하는 것이 좋습니다.

- **이전 형식 프로덕션 데이터베이스**에 해당하는 Asset Manager 버전: 필요한 경우 변환 전의 데이터 항목을 검증하기 위해 **이전 형식 프로덕션 데이터베이스**를 찾아볼 수 있습니다.

다음 프로그램은 변환 컴퓨터에서 설치 제거해도 됩니다.

- HP Connect-It
- XML 파일 편집기

■ Java Runtime

대체로 변환 프로세스를 완료하고 나면 **Asset Manager 9.30** 및 프로덕션 데이터 베이스의 일상적인 실행에는 변환 컴퓨터와 소프트웨어가 더 이상 필요하지 않습니다.

7 용어집

마이그레이션

마이그레이션은 다음과 같이 **Asset Manager**의 이전 버전을 버전 **9.30**으로 변환하는 데 필요한 작업 집합입니다.

마이그레이션에는 다음 작업이 포함됩니다.

- **Asset Manager**의 **9.30** 버전과 호환되도록 이전 형식 프로덕션 데이터베이스 변환(구조 및 내용)
- 모든 관리 시스템 및 사용자 시스템에서 **Asset Manager** 프로그램을 **9.30** 버전으로 업데이트

Asset Manager 프로그램 업데이트

Asset Manager 마이그레이션에 필요한 운영 중 하나입니다.

프로그램을 업데이트하게 되면 모든 관리 및 사용자 시스템에서 **Asset Manager** 프로그램이 모두 **9.30** 버전이 되도록 다시 설치됩니다.

흔들하기 쉬운 용어

- ▶ 이전 형식 프로덕션 데이터베이스 변환 [88 페이지]

이전 형식 프로덕션 데이터베이스 변환

Asset Manager 마이그레이션에 필요한 운영 중 하나입니다.

이전 형식 프로덕션 데이터베이스 변환에는 **Asset Manager**의 9.30 버전과 호환되도록 구조 및 내용을 수정하는 작업이 포함됩니다.

변환은 여러 단계에 걸쳐 수행됩니다. 일부 단계는 수동으로 수행해야 하며, 나머지 단계는 추가 도구를 사용하여 수행합니다.

흔들하기 쉬운 용어

▶ [Asset Manager 프로그램 업데이트 \[87 페이지\]](#)

변환 파일

변환 파일이란 이전 형식 프로덕션 데이터베이스 변환 중 변환할 데이터와 수행할 변환 작업을 설명하는 파일입니다.

변환 파일의 이름은 **migration.xml**입니다.

이들 파일은 일반적으로 **C:\Program Files\HP\Asset Manager 9.30xx\migration\fromxxx** 폴더에 있습니다.

Asset Manager은 기본적으로 변환 파일과 함께 설치됩니다(마이그레이션에서 지원되는 **Asset Manager**의 버전별로 파일 1개).

이러한 파일을 사용자 지정할 수 있습니다.

변환 시스템

변환 시스템이란 이전 형식 프로덕션 데이터베이스를 9.30 형식으로 변환하는 데 사용되는 컴퓨터를 말합니다.

이 컴퓨터에는 이 가이드에 설명된 특정 구성이 필요합니다.

프로덕션 데이터베이스

프로덕션 데이터베이스란 포트폴리오를 관리하는 데 사용되는 **Asset Manager** 데이터베이스입니다.

혼동하기 쉬운 용어

데모 데이터베이스

트리거

트리거란 데이터베이스 필드 또는 링크가 수정될 때 **Asset Manager**에서 자동으로 "발생"되는 작업을 말합니다.

데이터

데이터란 필드를 사용하는 데이터베이스에 저장되어 있는 레코드의 정보를 말합니다.

응용 프로그램 데이터

응용 프로그램 데이터는 구조가 아니라 프로덕션 데이터베이스에 저장되어 있는 데이터입니다. 변환 중에 검증해야 할 데이터가 바로 이 데이터입니다.

팁:

이 응용 프로그램 데이터는 버전 **9.30**에서 삭제되었거나 수정된 테이블, 필드 또는 링크를 참조하므로 반드시 검증해야 합니다.

다양한 응용 프로그램 데이터가 다음 범주 중 하나에 입력됩니다.

- **Basic** 스크립트
- **AQL** 쿼리
- 테이블 이름을 저장하는 필드
- 필드 이름을 저장하는 필드
- 마법사
- 계산된 문자열(지정된 컨텍스트의 링크 및 필드 문자열)

데이터베이스 구조

데이터베이스 구조는 다음 항목으로 조합됩니다.

- 테이블

- 필드
- 링크
- 색인

또한 다음과 같은 매개 변수도 조합합니다.

- 설명 문자열
- 유효성
- 관련성
- 이력
- 읽기 전용
- 필수
- 비관련성
- 기본값


이러한 매개 변수는 **Asset Manager Application Designer**에서 스크립트 또는 고정 값을 통해 정의됩니다.

이러한 매개 변수는 **gbase*.*** 데이터베이스 설명 파일이나 데이터베이스 자체에 저장됩니다.

8 참조

migration.xml 변환 파일 조정

경고

 **경고:**

변환 파일을 조정하려면 우수한 기술 능력을 갖추고 있어야 하며, Asset Manager 원본 버전은 물론 9.30 버전에 대해 잘 알고 있어야 합니다.

따라서 변환 파일 조정은 HP의 인증을 받은 엔지니어만 수행할 수 있습니다.

인증되지 않은 사용자에게 의한 변환 파일 수정은 HP의 책임이 아니라 수정하는 사용자의 전적인 책임 하에서 수행됩니다.

 **팁:**

HP와 해당 파트너는 이 변환 파일을 조정할 수 있는 전문적이고 경험이 풍부한 컨설턴트를 제공할 수 있습니다.

이 참조 섹션의 내용은 인증된 엔지니어에게만 해당합니다.

 [중요]:

migration.xml 변환 파일을 사용자 지정할 경우 파일의 이름을 변경하거나 대체해서는 안 됩니다. 이 파일을 사용하는 도구가 표준 폴더에서 검색하기 때문입니다. 또한 이 변환 파일을 수정하기 전에 변환 파일을 백업해 두는 것이 좋습니다.

미리 알림

변환 파일의 정의 ▶ [변환 파일 \[88 페이지\]](#).

변환 파일을 적용해야 하는 경우를 알아보려면 다음을 참조하십시오: ▶ [migration.xml 변환 파일 조정 \[43 페이지\]](#).

변환 파일이 수행하는 작업은 무엇입니까?

변환 파일은 다음과 같은 이유로 값을 보존할 수 없는 필드를 변환하기 위한 규칙을 정의합니다.

- 필드가 속해 있는 테이블이 사라졌거나 해당 테이블의 **SQL** 이름이 변경되었습니다.
- 필드가 사라졌거나 해당 필드의 **SQL** 이름이 변경되었습니다.
- 필드가 **9.30 형식 데이터베이스**의 테이블이나 직접 필드로 전송된 **Feature**의 일부입니다.

 [중요]:

표준 매핑에서 변환되지 않는 다른 **Feature**를 필드로 변환하려는 경우 새 매핑을 생성하지 마십시오. 특정 **Feature**를 필드로 전송 [\[76 페이지\]](#) 섹션을 참조하십시오.

 참고:

링크는 외래 키(실제 필드)를 통해 처리됩니다.

변환 파일은 **이전 형식 데이터베이스(DBMS에 사용된 SQL)**를 수정하기 위한 **SQL** 명령을 생성하는 데 사용됩니다.

변환 규칙

특정 변환 규칙은 변환 프로그램을 통해 다음과 같이 자동으로 결정됩니다.

- 테이블의 구조가 **Asset Manager**의 이전 버전과 **9.30** 버전 간에 동일한 경우 (**SQL** 이름, 필드, 링크 및 인덱스가 동일함):
변환 파일에서 필드를 선언하지 않아도 됩니다. 값이 변경되지 않습니다.



팁:

하지만 필요한 경우 구조적으로 변경되지 않은 테이블의 필드 및 링크에 대한 변환을 정의할 수 있습니다.

- 필드의 SQL 이름이 변환 파일의 Mapping 요소에서 연관된 원본과 대상 테이블에 대해 동일한 경우:
이러한 필드는 자동으로 연관됩니다. 값을 수정하지 않으려는 경우에는 변환 파일에 이러한 필드를 명시하지 않아도 됩니다.

변환 파일 구문

전역 구문

```
<?xml version="1.0" encoding="iso-8859-1"?> <!DOCTYPE MigrationFile SYSTEM "acmig.dtd"> <MigrationFile continueonerror=[AA]> <StartScript engine=[G]'> [A] </StartScript> <Translate table="[R]" into "[S]"/> <Mapping to="[C]" from="[B]" where="[K]" orderby="[O]" groupby="[P]" having="[Q]" autofill="[L]"> <PreActions engine="[T]"> [U] </PreActions> <Field sqlname="[E]" value="[F]" translate="[X]" feature="[Y]" featuretable="[Z]"> <Exception engine="[M]" value="[N]"/> </Field> <PostActions engine="[V]"> [W] </PostActions> </Mapping> <Script engine="[O]"> [I] </Script> <!--[J]--> <!--[P]--> </MigrationFile>
```

`<?xml version="1.0" encoding="iso-8859-1"?>` 라인

이 라인은 필수입니다.

이 라인은 파일에 사용된 문자 집합과 XML 버전을 명시합니다.

이 문자 집합을 수정할 수 있지만, .xml 파일에 사용된 문자 집합에 해당하는 경우에만 가능합니다.

`<!DOCTYPE MigrationFile SYSTEM "acmig.dtd">` 라인

이 라인은 .xml 파일에 연관시킬 .dtd를 나타냅니다.

Asset Manager는 migration.xml 변환 파일 다음에 acmig.dtd 파일을 설치합니다.

acmig.dtd는 필수는 아니지만 구조를 확인하는 데 유용하며, 이를 통해 .xml 파일을 보다 쉽게 읽을 수 있습니다.

acmig.dtd를 활성 상태로 설정하려면 XML 편집기를 사용해야 합니다.

MigrationFile 요소

이 요소에는 변환 중에 수행할 운영에 대해 설명하는 네 가지 요소가 들어 있습니다.

- StartScript

- Translate
- 매핑
- 스크립트

continueonerror 속성

이 속성은 선택 사항입니다.

AA가 No로 설정된 경우 변환 오류가 처음 나타날 때 변환이 중단됩니다.

AA가 Yes로 설정된 경우 변환 중에 오류가 발생해도 가능한 한 변환이 계속 진행됩니다.

기본적으로 이 속성은 No로 설정됩니다.

engine 속성

이 선택적 속성은 몇 가지 요소에서 요소가 적용될 DBMS를 정의하는 데 사용됩니다.

가능한 값:

- MSSQL
- Oracle
- DB/2

대/소문자를 구분해야 합니다.

StartScript 요소

이 요소에는 [A] SQL 스크립트가 포함되어 있는데, 이 스크립트는 **이전 형식 데이터베이스**를 변환하기 전과 이전 테이블의 이름을 변경하기 전에 실행됩니다.

가능한 경우에는 PreActions 요소를 사용하는 것이 좋습니다. 이 요소는 변환 파일에 대한 유지 보수를 도와줍니다.

StartScript 요소는 다음의 경우에 유용합니다.

- 동일한 PreActions 요소를 실행하기 위해 Mapping 요소가 여러 개 필요한 경우
- **이전 형식 데이터베이스** 구조에 적용된 사용자 지정을 제거해야 하는 경우
- 트리거를 비활성화해야 하는 경우

이 스크립트는 **이전 형식 데이터베이스**의 DBMS에 사용된 스크립트에 맞게 SQL 언어로 작성되어야 합니다.

팁:

이 제약 조건에는 한 가지 예외가 있습니다. 즉 문자열을 연결하려는 경우 모든 엔진에서 || 연산자를 사용할 수 있습니다(MSSQL의 경우 +로 변환됨).

경고:

Asset Manager의 AQL 언어는 인식되지 않습니다.

각 SQL 명령줄은 GO 행을 사용하여 실행됩니다.

예:

```
UPDATE amPortfolio SET lParentId=0 WHERE lPortfolioItemId IN (SELECT p.lPortfolioItemId FROM amAssetOld a, amPortfolio p WHERE a.lParentId=0 AND p.lAstId=a.lAstId) GO DELETE FROM amItemListVal WHERE lItemlId=(SELECT lItemlId FROM amItemizedList WHERE Identifier='amBrand') GO
```

engine 속성

DBMS가 [G]인 데이터베이스에서 StartScript 요소가 실행되는 경우 engine 속성이 있는 StartScript 요소는 engine 속성이 없는 StartScript 요소를 대체합니다.

Translate 요소

이 요소는 테이블 이름을 저장하는 필드 변환 중에 사용됩니다(예: 작업의 컨텍스트).

원본 테이블 **[R]**이 여러 Mapping 요소 내부에 있는 몇몇 대상 테이블 **[S]**와 연관된 경우 Translate 요소를 정의해야 합니다.

Translate 요소는 이러한 **[S]** 테이블 중 어느 것이 테이블 이름을 저장하는 필드의 자동 변환 대상 테이블인지 나타내는데 사용됩니다.

테이블 이름을 저장하는 필드의 변환에는 매핑 테이블이 사용되며, 이 테이블은 migration.xml 변환 파일에 있는 정보를 사용하여 변환 착수 시 자동으로 생성됩니다.

매핑 테이블은 다음을 매핑합니다.

- 테이블 **[C]**와 **[B]**가 다른 경우 to="**[C]**" 및 from="**[B]**" 속성별로 Mapping 요소에 연관된 테이블
 - table="**[R]**" 및 into "**[S]**" 속성별로 Translate 요소에 연관된 테이블
- Translate 요소에서 수행된 연관은 Mapping 요소에서 수행된 연관보다 우선합니다.

맵 테이블은 UPDATE 명령을 사용하는 변환 파일 스크립트에서 사용됩니다.

이렇게 하면 이전 테이블 이름을 새 테이블 이름으로 교체할 수 있습니다.

예:

```
UPDATE amDocument SET DocObjTable = ( SELECT newsqname FROM s dutrans WHERE oldsqlname = amDocument.DocObjTable ) WHERE amDocument.DocObjTable IN( SELECT oldsqlname FROM s dutrans)
```

Mapping 요소

이 요소를 사용하면 이전 구조의 테이블에 있는 필드를 버전 9.30 구조의 테이블로 전송 및 변환할 수 있습니다.

from 속성

from 속성은 필수입니다. 이 속성은 이전 구조의 [B] 테이블을 식별합니다. 조인 시 다음 구문에 따라 여러 개의 테이블을 사용할 수 있습니다.

```
from="[테이블 1의 SQL 이름] alias1, [테이블 2의 SQL 이름] alias2, ..., [테이블 n의 SQL 이름] aliasn"
```

to 속성

to 구조는 필수입니다. 이 구조는 새 구조의 [C] 테이블을 식별합니다.

where 속성

where 속성은 선택 사항입니다. 이 속성은 Mapping 요소를 통해 처리해야 하는 [B] 테이블의 레코드를 정의하는 [K] SQL 조건을 지정합니다.

기본적으로 where 절은 원본 테이블 [B]에서 null 기본 키 레코드를 제외시킵니다 (내부 조인 - where [기본 키의 SQL 이름] <> 0).

기본적으로 where 절은 [B] 테이블에 연결된 원격 테이블의 null 기본 키 레코드를 포함시킵니다(외부 조인).

예를 들어, 다음과 같은 연관에서,

```
<Mapping to="amCatProduct" from="amProdSoftInfo s, amSoftware soft" where="s.lSoftId = soft.lSoftId">
```

s.lSoftId와 soft.lSoftId가 동일한 레코드는 유지됩니다.

null 기본 키 레코드의 역할에 대해 알아보려면 **고급 사용** 가이드의 **AQL 쿼리** 장, **AQL 쿼리 작성**을 위한 권장 사항 섹션에서 하위 섹션인 **기본 키(PK) 0 레코드의 이유 및 유용성**을 참조하십시오.

orderby 속성

orderby 속성은 선택 사항입니다. 이 속성은 SQL 정렬 [O]의 순서를 지정합니다.

groupby 속성

groupby 속성은 선택 사항입니다. 이 속성은 [P] SQL 하위 집합을 지정합니다.

having 속성

having 속성은 선택 사항입니다. 이 속성은 [Q] SQL 검색 조건을 지정합니다.

autofill 속성

autofill 속성은 선택 사항입니다. 이 속성에는 값으로 yes 또는 no가 허용됩니다. 기본적으로 이 속성의 값은 yes입니다.

이 속성의 값이 no이면 Field 요소를 통해 처리되는 [C] 테이블의 필드만 채워집니다.

변환 프로그램을 통해 자동으로 연관된 필드는 채워지지 않습니다. 이러한 필드는 SQL 이름이 **[B]** 및 **[C]** 테이블에서 동일한 필드입니다.

PreActions 요소

이 요소는 다음에 나오는 **Field** 요소를 실행하기 전에 실행할 **SQL** 스크립트 **[U]**를 포함합니다.

PreActions 요소는 다음의 경우에 유용합니다.

- 변환할 데이터베이스의 내용과 관계없는 특성을 생성하는 경우
- **Feature**를 생성하는 경우

[중요]:

표준 매핑에서 변환되지 않는 다른 **Feature**를 필드로 변환하려는 경우 새 매핑을 생성하지 마십시오. 특정 **Feature**를 필드로 전송 [\[76 페이지\]](#) 섹션을 참조하십시오.

이 요소의 구문은 **StartScript** 요소의 구문과 동일합니다.

고급 사용자는 **Mapping** 요소를 사용하여 수행할 수 없는 작업을 수행하기 위해 이와 같은 스크립트를 실행합니다.

PreActions 요소를 실행할 때는 이전 테이블이 아직 삭제되지 않는 상태입니다. 따라서 여전히 이전 데이터를 사용할 수 있습니다.

PreActions 요소는 이전 형식 프로덕션 데이터베이스의 표준 구조를 수정한 사용자를 대상으로 합니다.

Field 요소

이 요소를 사용하면 새 **SQL** 이름 필드 **[E]**를 **SQL** 표현식 **[F]**를 통해 계산된 값으로 채울 수 있습니다.

SQL 표현식 **[F]**는 **SQL** 이름에 의해 식별된 **[B]** 테이블의 필드에 의존해야 합니다.

SQL 표현식 **[F]**가 지정된 **DBMS**에서 유효하지 않으면 **Field** 요소 라인 바로 뒤에 오는 **Exception** 요소를 채워야 합니다.

feature 속성

이 속성은 원본 **Feature** 값을 **9.30 형식 데이터베이스**의 필드로 변환하는 데 사용됩니다.

[중요]:

표준 매핑에서 변환되지 않는 다른 **Feature**를 필드로 변환하려는 경우 새 매핑을 생성하지 마십시오. 특정 **Feature**를 필드로 전송 [\[76 페이지\]](#) 섹션을 참조하십시오.

이 속성의 **[Y]** 값은 값을 변환할 **Feature**의 **SQL** 이름에 해당합니다.

featuretable 속성

이 속성은 원본 Feature 값을 **9.30 형식 데이터베이스**의 필드로 변환하는 데 사용됩니다.

[중요]:

표준 매핑에서 변환되지 않는 다른 Feature를 필드로 변환하려는 경우 새 매핑을 생성하지 마십시오. 특정 Feature를 필드로 전송 [76 페이지] 섹션을 참조하십시오.

이 속성의 [Z] 값은 값을 변환해야 할 Feature 값을 저장하는 테이블의 SQL 이름에 해당합니다.

경고:

[Z] 테이블에서 연관되어 있는 Feature 값을 저장하는 테이블은 Mapping 요소의 from 속성 수준에서 선언됩니다.

예: **amFVAsset** 테이블은 **amAsset** 테이블의 레코드와 연관된 Feature 값을 저장합니다. [Y] Feature 값을 필드로 변환하려면 **amAsset** 테이블이 from 속성 수준에서 선언되어야 합니다. 그리고 **amFVAsset** 테이블은 **featuretable** 속성 수준에서 선언됩니다.

Exception 요소

이 요소를 사용하면 이 요소의 앞에 오는 Field 요소에 대해 지정된 DBMS 관련 예외 사항을 생성할 수 있습니다.

engine 속성

engine 속성을 사용하면 예외 사항이 적용될 [O] DBMS를 정의할 수 있습니다. Exception 요소는 [O] DBMS의 Field 요소를 대체합니다.

value 속성

value 속성을 사용하면 [O] DBMS에 유효한 SQL 표현식을 정의할 수 있습니다. 조인 시 다음 구문에 따라 별칭을 사용해야 합니다.

```
value="[테이블 별칭].[필드의 SQL 이름]"
```

SDU_NEWID 변수

이 변수는 경우에 따라 기본 키의 새 값을 정의하는 value 속성에서 사용됩니다. SDU_NEWID는 1씩 증가되는 이전 형식 데이터베이스에서 최대 숫자 값을 가진 기본 키 ID의 값입니다.

SDU_NEWID는 변환 프로그램을 통해 자동으로 계산됩니다.

PostActions 요소

이 요소는 이 요소 앞에 오는 **Field** 요소를 실행한 후 실행할 **SQL** 스크립트 [W]를 포함합니다.

PostActions 요소는 다음의 경우에 유용합니다.

- **전체 이름** 필드의 값을 계산합니다.
- 필드로 전송되는 **Feature**와 **Feature** 값을 삭제합니다.

[중요]:

표준 매핑에서 변환되지 않는 다른 **Feature**를 필드로 변환하려는 경우 새 매핑을 생성하지 마십시오. 특정 **Feature**를 필드로 전송 [76 페이지] 섹션을 참조하십시오.

이 요소의 구문은 **StartScript** 요소의 구문과 동일합니다.

고급 사용자는 **Mapping** 요소를 사용하여 수행할 수 없는 작업을 수행하기 위해 이와 같은 스크립트를 실행합니다.

PostActions 요소를 실행할 때는 이전 테이블이 아직 삭제되지 않는 상태입니다.

따라서 여전히 이전 데이터를 사용할 수 있습니다.

PostActions 요소는 이전 형식 프로덕션 데이터베이스의 표준 구조를 수정한 사용자를 대상으로 합니다.

Script 요소

이 요소는 **Mapping** 요소를 실행한 후 더 이상 사용되지 않는 이전 테이블을 삭제하기 전에 실행할 **SQL** 스크립트 [I]를 포함합니다.

가능한 경우에는 **PostActions** 요소를 사용하는 것이 좋습니다. 이 요소는 변환 파일에 대한 유지 보수를 도와줍니다.

Script 요소는 다음의 경우에 유용합니다.

- 동일한 **PostActions** 요소를 실행하기 위해 **Mapping** 요소가 여러 개 필요한 경우
- **Mapping** 요소를 사용하여 수행할 수 없는 정리 작업을 수행하려는 경우
- 더 이상 사용되지 않는 열거 항목을 삭제하는 경우

이 요소의 구문은 **StartScript** 요소의 구문과 동일합니다.

Script 요소를 실행할 때는 이전 테이블이 아직 삭제되지 않는 상태입니다.

따라서 여전히 이전 데이터를 사용할 수 있습니다.

Script 요소는 이전 형식 프로덕션 데이터베이스의 표준 구조를 수정한 사용자를 대상으로 합니다.

!- 요소

이 태그를 사용하면 코드에 [J] 주석을 삽입할 수 있습니다. 변환 프로그램에서 이 주석은 고려되지 않습니다.

!- 요소

이 태그를 사용하면 변환 파일 사용자를 위해 [J] 주석을 삽입할 수 있습니다. 변환 프로그램에서 이 주석은 고려되지 않습니다.

특수 문자 사용

다음은 특수한 방식으로 해석할 수 있는 특정 문자 사용을 나타내는 지침입니다.

이러한 지침은 전체를 철저히 설명하지는 않습니다. 자세한 내용은 **SQL** 및 **XML** 설명서를 참조하십시오.

일반적으로 변환 파일의 일반 구조는 **XML** 제약 조건을, 속성 값은 **SQL** 제약 조건을 준수해야 합니다.

특수하게 해석되는 일부 문자는 다음과 같습니다.

특수 문자	해석	예	문자를 텍스트로 해석해야 하는 경우에 해당	예
"	XML 속성 값을 구분합니다.	value="!AssetRentId"	\	value="\ ""
'	속성 값 내부에서 SQL 텍스트 문자열을 구분합니다.	value='!ofPublisher/!ofName'	"	value="" ""
<	XML 태그를 엽니다.	</Mapping>	<	value=""\$!t;"
>	XML 태그를 닫습니다.	</Mapping>	\$gt;	value="">"
&	엔티티 시작을 표시합니다.	<	&	value=""&"
;	엔티티 끝을 표시합니다.	<	; without & before	value="";"
\	SQL 이스케이프 문자입니다.		\\	value=""\ ""
	SQL 문자열 연결 문자 (모든 DBMS에 유효함)입니다.	value=""A 'B'"	' ' ' '	value=""A ' ' B'"

여러 개의 새 테이블 간 이전 테이블 필드 나누기

예를 들어, **Asset Manager**의 이전 버전에서는 자산 테이블을 사용했습니다. 이 버전에서는 포트폴리오 항목 테이블과 자산 테이블이 있습니다. 따라서 이전 자산 테이블의 필드를 이러한 두 개의 새 테이블로 나누어야 합니다. 그리고 이전 자산 테이블에서 레코드 하나는 두 개의 레코드가 됩니다(각 새 테이블에 하나씩).

이러한 이유로 지금 포트폴리오 항목 테이블에서 기본 ID를 생성해야 합니다. 이러한 레코드가 하나의 테이블뿐 아니라 전체 **Asset Manager** 데이터베이스에서 고유해야 하기 때문입니다.

다음 유형의 **Mapping** 요소를 생성해야 합니다.

```
<Mapping to="amPortfolio" from="amAsset"> <Field sqlname="lPortfolioItem  
Id" value="SDU_NEWID+lAstId"/> </Mapping>
```

Feature를 필드로 전송

Asset Manager 9.30는 새 필드를 포함합니다.

경우에 따라 이러한 새 필드는 이전 형식 프로덕션 데이터베이스에 사용된 Feature 대신에 사용됩니다.

[중요]:

이 섹션의 내용은 기존 매핑의 구문을 이해하는 데 유용합니다.

반면에 기타 Feature를 필드로 변환하지 않으려는 경우에는 새 매핑을 생성하지 마십시오. 특정 Feature를 필드로 전송 [76 페이지] 섹션을 참조하십시오.

구문

```
<Mapping to="[대상 테이블의 SQL 이름]" from="[Feature 값을 저장하는 원본  
테이블의 SQL 이름]"> <Field sqlname="[대상 필드의 SQL 이름]" value="[Fea  
ture 값을 저장하는 필드의 SQL 이름]" feature="[원본 Feature의 SQL 이름]" fe  
aturetable="[Feature 값을 저장하는 테이블의 SQL 이름]"/> </Mapping>
```

별칭은 모든 테이블에 사용됩니다. 이러한 별칭은 Feature 값을 저장하는 필드를 참조하는 value 속성의 경우를 제외하고는 속성 수준에서 사용됩니다.

Value 속성에서는 다음 값을 가져올 수 있습니다.

- Feature에서 텍스트를 저장하는 경우 **ValString**
- Feature에서 숫자를 저장하는 경우 **fVal**
- Feature에서 날짜를 저장하는 경우 **dtVal**

예

```
<Mapping to="amComputer A" from="amAsset"> <Field sqlname="VideoCar  
d" value="ValString" feature="Video Card" featuretable="amFVAsset"/> </M  
apping>
```

제한 사항

Feature를 필드로 전송하는 이 방법에는 몇 가지 제한이 있습니다.

- 다양한 조인을 사용해야 합니다.
- 변환 성능이 느려질 위험이 있습니다.
- Feature 상속을 관리할 수 없습니다.
- 전송된 Feature 값의 삭제 또는 Feature 자체 삭제를 관리할 수 없습니다.
Field 요소 뒤에 PostActions 요소를 추가하여 이 작업을 수행할 수 있습니다.

그렇지 않으면 삭제가 변환 후에 수동으로 수행됩니다.

여러 개의 **Feature**를 변환하기 위해 다음 예에서처럼 <Script> 요소를 사용하도록 선택했습니다.

```
UPDATE amComputer SET ComputerDesc = (SELECT F.ValString FROM amFVAsset F, amFeature V, amAsset A WHERE lComputerId = SDU_NEWID * 2 + A.lAstId AND F.lFeatId = V.lFeatId AND V.SQLName='fv_BiosMachine') GO DELETE FROM amFVAsset WHERE lFeatValId IN ( SELECT lFeatValId FROM amFVAsset F, amFeature V WHERE F.lFeatId = V.lFeatId AND V.SQLName='fv_BiosMachine' ) GO
```

링크 유형 **Feature**와 관련된 잠재적 문제

링크 유형 **Feature**의 값을 9.30 데이터베이스의 링크로 전송할 때 대상 테이블이 변환 중에 변경되면 링크가 채워지지 않습니다.

예: 변환 전, **Feature**는 자산 테이블을 가리킵니다. 변환 후, **Feature**를 대체한 링크는 **Portfolios** 테이블을 가리킵니다.

이 예에서는 자산의 ID가 사라지고 새 포트폴리오 항목의 ID가 생성됩니다.

변환 후 링크(**Feature**를 대체한 항목)가 채워지지 않는 레코드를 식별하기 위해 쿼리를 실행해야 합니다.

수동으로 변환할 응용 프로그램 데이터를 저장하는 필드 변환

수동으로 변환할 응용 프로그램 데이터를 저장하는 필드는 **Mapping** 요소를 사용하여 변환 중에 의도적으로 비워집니다. 다음은 이러한 동작을 보여주는 예입니다.

```
<Mapping to="amAccessRestr" from="amAccessRestr"> <Field sqlname="ReadCond" value=""/> </Mapping>
```

하지만 빈 응용 프로그램 데이터를 포함하는 레코드는 마이그레이션 중에도 여전히 보존됩니다.

수동으로 변환할 응용 프로그램 데이터는 손실되지 않습니다. 이는 변환하기 전에 **Asset Manager Application Designer**를 사용하여 해당 데이터를 내보냈기 때문이며, 이러한 데이터는 변환 프로세스를 수행하는 동안 나중에 복원됩니다.

수동으로 변환할 요소의 컨텍스트로 사용되지 않는 테이블 이름을 저장하는 필드는 자동으로 변환됩니다.

변환 메커니즘은 **Translate** 요소를 사용하여 구성할 수 있습니다.

조인 사용

조인은 다음 규칙을 준수해야 합니다.

- 조인의 각 테이블마다 별칭을 정의해야 합니다.

- Field 요소에 대한 where, orderby, groupby, having 및 value 속성의 표현식에 서는 별칭별로 테이블을 식별해야 합니다.



경고:

정수(32비트) 또는 가변 길이 바이너리 필드에 관한 조인은 지원되지 않습니다.

예

```
<Mapping from="amProdSoftInfo s, amSoftware soft" to="amCatProduct" where="s.lSoftId = soft.lSoftId"> <Field sqlname="lCatProductId" value="s.lProdSoftId"/> <Field sqlname="InternalRef" value="soft.Publisher+'/' +soft.Name+'/' +soft.VersionLevel"/> <Field sqlname="FullName" value="/' +soft.Publisher+'.' +soft.Name+'.' +soft.VersionLevel+'/'"/> <Field sqlname="dtLastModified" value="s.dtLastModif"/> </Mapping>
```



참고:

from 속성을 통해 지정된 첫 번째 테이블은 특수한 상태를 갖습니다.

이 테이블 필드는 변환 파일에 없는 경우 동일한 SQL 이름을 가진 대상 테이블의 필드와 자동으로 연관됩니다.

외래 키 채우기

외래 키는 다양한 테이블의 레코드 사이에 링크를 생성하는 데 사용됩니다.

예

```
<Mapping from="amAsset" to="amPortfolio" <Field sqlname="lParentId" value="SDU_NEWID+lParentId"/> </Mapping>
```

두 개 이상의 대상 테이블 간 원본 테이블 나누기

하나의 원본 테이블을 두 개 이상의 대상 테이블 간에 나누어야 하는 경우 대상 테이블에서 생성된 기본 ID가 Asset Manager 데이터베이스 전체에서 고유해야 합니다.

이 기술은 다음 유형의 Field 요소 생성을 포함합니다.

```
<Mapping to="amPortfolio" from="amAsset"> <Field sqlname="lPortfolioItemId" value="SDU_NEWID * 2 + lAstId"/> </Mapping>
```

숫자 문자열을 텍스트 문자열로 변환

데이터 변환 시 숫자 문자열을 텍스트 문자열로 변환해야 하는 경우가 종종 있습니다.

숫자 필드에 따라 텍스트 필드의 값을 계산해야 하는 경우가 이에 해당합니다.

이러한 변환은 SQL 언어를 사용하여 수행하는 복잡한 변환이며, 엔진별로 다릅니다.

모든 엔진 및 모든 숫자 필드 유형에 대해 이 변환을 쉽게 처리할 수 있는 SDUSTR 매크로를 작성했습니다.

예:

```
<Mapping to="amPortfolio" from="amSoftInstall"> <Field sqlname="Code" value="'^' || SDUSTR IInstId"/> </Mapping>
```

이 예에서는 다음과 같습니다.

- IInstId 필드는 정수(32비트) 유형 필드입니다.
- Code 필드는 텍스트 유형 필드입니다.
- IInstId는 SDUSTR 매크로를 통해 텍스트 문자열로 변환됩니다.
- 변환된 문자열은 ^ 문자를 사용하여 연결됩니다.
- 연결된 문자열은 Code 필드에 삽입됩니다.

응용 프로그램 데이터 수동 변환

특정 Mapping 요소의 역할은 수동으로 변환할 응용 프로그램 데이터를 비우는 것입니다.

다음은 이러한 동작을 보여주는 예입니다.

```
<Mapping to="amAccessRestr" from="amAccessRestr"> <Field sqlname="TableName" value=""/> </Mapping>
```

비워진 필드는 수동으로 변환된 응용 프로그램 데이터의 복원 중에 다시 채워집니다.

변환 파일에서 생성된 SQL 명령

변환 파일은 이전 형식 데이터베이스(구조 및 데이터)를 수정하기 위해 DBMS에서 사용하는 SQL 명령을 생성하는 데 사용됩니다.

예

다음 Mapping 요소의

```
<Mapping from=[F] to=[T] where=[W]> <Field sqlname=[F1] value=[V1]/> <Field sqlname=[F2] value=[V2]/> ... <Field sqlname=[Fn] value=[Vn]/> </Mapping>
```


SQL 문은 다음과 같습니다.

```
Insert Into to T(F1; F2, ..., Fn) Select V1 as F1, V2 as F2, ..., Vn as Fn From A  
Where W
```

사용하기 전에 변환 파일 검증

경고:

변환에 사용하기 전에 변환 파일이 `acmig.dtd` 파일을 준수하는 방법을 확인해야 합니다.

준수를 확인하려면 **Internet Explorer** 또는 텍스트 편집기를 사용해야 합니다.

다음은 몇 가지 권장 테스트 항목입니다.

- 변환 파일에 조합이 발생하면 안 됩니다(**from, to, where, groupby**).
- **Mapping** 요소가 데이터베이스 사용 방식과 맞습니다.
- 동일한 원본 기본 키에서 생성된 여러 기본 키가 다양합니다(**SDU_NEWID** 변수에 적합한 사용).
- 변환 중에 생성된 기본 키를 저장하는 외래 키가 올바른 기본 키에 해당합니다.
- 변환 파일에서 수동으로 또는 변환 도구를 통해 자동으로 연관되지 않은 원본 및 대상 필드가 의도적으로 연관되어 있지 않습니다.
이러한 사항을 검증하려면 다음을 수행합니다.
 - 1 변환 로그 폴더에 있는 `sdu.xml` 파일을 표시합니다.
 - 2 **NotMappedSrc** 및 **NotMappedDst**를 검색합니다.
- **where** 속성에서 정의된 레코드 하위 집합은 복구되지 않습니다.
이러한 레코드가 모든 레코드에 적용됩니다.
- 여러 번 연관된 테이블은 링크가 한 번만 존재할 때 동일한 레코드에 대한 여러 링크의 생성을 트리거하지 않습니다(예: **IParentId** 또는 **ICommentId** 링크).

특정 테이블이 없는 자산 전송

경우에 따라 특정 유형의 특정 자산(예: 컴퓨터)을 설명할 수 있게 하는 **오버플로 테이블**이 있습니다.

이 경우 **Mapping** 요소는 변환 파일에 추가되었습니다.

특정 **오버플로** 테이블(예: 차량 테이블)이 없는 경우에는 이전 형식 프로덕션 데이터베이스에서처럼 데이터 조직을 변경하지 않는 것이 좋습니다.

이전 버전과 비교되는 표준 데이터베이스에 대한 구조적 수정

Asset Manager 9.30은 다음 버전 간 데이터베이스 구조 차이를 설명하는 파일 (diff*.*)과 함께 설치됩니다.

- 버전 9.30.
- 지정된 이전 버전

경고:

diff*. * 파일에서는 **이전 형식 프로덕션 데이터베이스**에 적용할 수 있었던 사용자 지정을 고려하지 않습니다.

diff*. * 파일은 다음과 같은 형식으로 제공됩니다.

- 텍스트(diff*.txt)
- XML(diff*.xml).
- HTML(diff*.htm).

이들 파일은 일반적으로 C:\Program Files\HP\Asset Manager 9.30 xx\doc\infos 폴더에 있습니다.

설치 중에 **설명서** 패키지를 선택할 경우 설치됩니다.

이러한 파일의 이름의 형식은 다음과 같습니다.

diff<Asset Manager 이전 버전>.*

팁:

Asset Manager의 이전 버전을 시작하고 **도움말/Asset Manager 정보** 메뉴를 열면 버전 번호를 찾을 수 있습니다.

diff*.txt 파일 사용

DOS 또는 ASCII 형식의 텍스트 파일로 지정하는 Excel이나 다른 도구로 이 파일을 엽니다.

팁:

Excel에서는 표시하려는 변경 사항에 따라 정보를 필터링할 수 있도록 첫 번째 행에 자동 필터를 적용하는 것이 좋습니다.

제목은 각 열의 내용을 설명합니다.

제목 다음에 오는 각 라인은 표준 데이터베이스의 구조적 수정에 해당합니다.

다음은 사용 가능한 특정 열에 대한 정보입니다.

- 개체를 포함하는 테이블의 이름
 - 테이블 생성

<9.30 형식 표준 데이터베이스에서의 SQL 이름>

- 테이블 삭제

<SQL 이름, 또는 이 이름이 없는 경우에는 이전 형식 표준 데이터베이스의 코딩 이름>

- 필드, 인덱스 또는 링크의 생성, 삭제 또는 수정(테이블 수정)

<SQL 이름, 또는 이 이름이 없는 경우에는 이전 형식 표준 데이터베이스의 코딩 이름>(<9.30 형식 표준 데이터베이스에서의 SQL 이름>)

- 수정된 객체의 이름
 - 삭제된 개체

<SQL 이름, 또는 이 이름이 없는 경우에는 이전 형식 표준 데이터베이스의 코딩 이름>

- 수정된 개체

<SQL 이름, 또는 이 이름이 없는 경우에는 이전 형식 표준 데이터베이스의 코딩 이름>(<9.30 형식 표준 데이터베이스에서의 SQL 이름>)

- 추가된 개체

<9.30 형식 표준 데이터베이스에서의 SQL 이름>

- 설명:
 - 수정 또는 생성된 개체: 객체의 새 설명.
 - 삭제된 개체 객체의 이전 설명.

diff*.htm 파일 사용

이러한 파일은 HTML 브라우저를 사용하여 참조할 수 있습니다.

구조는 다음과 같습니다. 이러한 파일을 찾아보기 위해 다음 표현식을 검색할 수 있습니다.

1 Deleted table information

이 제목은 삭제된 테이블을 설명하는 각 섹션의 시작 부분에 있습니다.

각 테이블에서 찾을 내용은 다음과 같습니다.



- 삭제된 테이블에 대한 정보
- ■ 삭제된 테이블의 필드
- ■ 삭제된 테이블의 링크
- ■ 삭제된 테이블의 인덱스

2 Inserted table information

이 제목은 추가된 테이블을 설명한 각 섹션의 시작 부분에 있습니다.

각 테이블에서 찾을 내용은 다음과 같습니다.

- 추가된 테이블에 대한 정보
- ■ 추가된 테이블의 필드

-  추가된 테이블의 링크
-  추가된 테이블의 인덱스

3 Modified table

- Deleted objects
- Inserted objects
- Modified objects

diff*.xml 파일 사용

이러한 파일은 XML을 사용해 본 경험이 있으며 XML 파일이 필요한 경우 간편하게 사용할 수 있습니다.

이러한 파일을 스스로 살펴보고 어떤 용도로 사용할지 결정합니다.

수동으로 변환할 응용 프로그램 데이터

이 섹션에는 마이그레이션 중에 검증할 응용 프로그램 데이터의 목록이 포함되어 있습니다.

팁:

이 응용 프로그램 데이터는 버전 9.30에서 삭제되었거나 수정된 테이블, 필드 또는 링크를 참조하므로 반드시 검증해야 합니다.

Asset Manager Script Analyzer 데이터베이스에 저장된 응용 프로그램 데이터

다양한 응용 프로그램 데이터가 다음 범주 중 하나에 입력됩니다.

- Basic 스크립트
- AQL 쿼리
- 테이블 이름을 저장하는 필드
- 필드 이름을 저장하는 필드
- 마법사
- 계산된 문자열(지정된 컨텍스트의 링크 및 필드 문자열)

데이터 및 매개 변수를 확인하고 수정하려면 다음을 참조하십시오: ▶ 수동으로 변환할 응용 프로그램 데이터 처리 [56 페이지].

이 응용 프로그램 데이터는 Asset Manager의 그래픽 인터페이스를 통해 액세스할 수 있습니다.

변환 중에는 이 데이터가 수정되지 않습니다.

Asset Manager Script Analyzer는 잠재적인 문제를 분석하며, 이를 통해 수동으로 변환할 응용 프로그램 데이터를 수동으로 수정할 수 있습니다. 이렇게 하면 9.30 버전 데이터베이스의 구조에 맞게 데이터를 조정할 수 있습니다.

8.1. 표 수동으로 변환할 응용 프로그램 데이터 - 목록

테이블(SQL 이름)	필드 또는 링크(SQL 이름)	제한
amAction	WizardScript 스크립트 MsgTo MsgCc MsgBcc 제목 memMsgText ActionFile Folder 매개 변수 DDEService DDETopic DDECommand ContextTable RefObject	
amQuery	memQueryText TableName	
amWfActivity	memScript ContextTable	
amWfTransition	AQLCond LinkToTargetCtxTbl TargetContextTable	
amWfEvent	AQLCond memScript ContextTable MonitTable MonitFields LinkToMonitTable	
amCalcField	memScript AQL ComputeString TableName 스크립트	버전 4.0.0부터 버전 4.0.0부터
amAccessRestr		버전 4.0.0부터

테이블(SQL 이름)	필드 또는 링크(SQL 이름)	제한
	WriteCond	
	TableName	
	ReadCond	
amTaxFormula	memFormula	
	TableName	
amWfOrgRole	memScript	
	ContextTable	
amFeatParam	AvailScript	
	DefValScript	
	MandatScript	
	ForceDspScript	
	HistScript	
	TableName	
	LinkFilter	
amFeatScript	memScript	
amOption	memOptValue	
amFieldAdjustTempl	memScript	
	ContextTable	
	TargetField	
amFieldAdjust	TargetField	
	AdjustedTable	버전 4.0.0부터
amDeprScheme	memScript	
amLoan	ProrateField	
amCntrRent	ProrateField	
	설명	버전 3.6.0부터
amDateAlarm	MonitoredField	
	MonitoredTable	
amLabelRule	memScript	버전 3.10, 4.0.0 이상
	TableName	버전 3.10, 4.0.0 이상
	FieldName	버전 3.10, 4.0.0 이상
amCatRefScript	memScript	버전 4.0.0부터
amScriptLibrary	memScript	버전 4.0.0부터
amCbKStoredEvent	memScript	버전 4.0.0부터

테이블(SQL 이름)	필드 또는 링크(SQL 이름)	제한
	FieldName	버전 4.0.0부터
	컨텍스트	버전 4.0.0부터
amCbkJRule		
	AmountField	버전 4.0.0부터
	EvtField	버전 4.0.0부터
	컨텍스트	버전 4.0.0부터
amCbkJScript		
	컨텍스트	버전 4.0.0부터
	memScript	버전 4.0.0부터

검증할 기타 응용 프로그램 데이터

다음 응용 프로그램 데이터는 Asset Manager Script Analyzer를 사용하여 변환되거나 검증되지 않습니다.

- 필드 도움말
- 보기
- 가져오기 스크립트
- 다음에 대한 웹 페이지
 - Asset Manager Web
 - Get-It
 - Get-Resources
- HP Connect-It 시나리오
- Asset Manager Export Tool 내보내기 스크립트
- SAP Crystal Reports

이러한 유형의 응용 프로그램 데이터는 하나씩 테스트해야 합니다.

전파된 이전 형식 프로덕션 데이터베이스의 구조적 매개 변수

구조적 매개 변수는 Asset Manager Application Designer를 사용하여 정의됩니다.

이전 형식 프로덕션 데이터베이스의 구조적 매개 변수는 구조적 변경 사항이 전파되는 단계에서 9.30 형식 표준 gbase*.* 파일로 전파됩니다.

- ▶ 3단계 - 이전 형식 프로덕션 데이터베이스에 대한 구조 변경 전파 [37 페이지]

8.2. 표 이전 형식 프로덕션 데이터베이스의 구조적 매개 변수 - 목록

데이터베이스 개체	매개 변수	다음 버전에서 사용 가능		
		3.0.1, 3.0.2, 3.1.0, 3.5.1, 3.5.2 및 3.6.0	4.0.0	4.1.0 이상
테이블	Name	예	예	예
테이블	설명	예	예	예
테이블	계층	아니요	아니요	예
테이블	Feature를 포함할 수 있음	아니요	예	예
테이블	문자열	예	예	예
테이블	유효성	예	예	예
테이블	관련성	아니요	예	예
필드 또는 링크	Name	예	예	예
필드 또는 링크	설명	예	예	예
필드	크기	예	예	예
필드	업데이트됨	아니요	예	예
필드 또는 링크	이력	예	예	예
필드 또는 링크	읽기 전용	예	예	예
필드 또는 링크	필수	예	예	예
필드 또는 링크	비관련성	아니요	예	예
필드	서식	예	예	예
필드 또는 링크	기본값	예	예	예
필드 또는 링크	중복되는 경우 기본값 다시 적용	예	예	예

기타 설명서

이 가이드에서는 마이그레이션 프로세스에 직접 연결된 정보만 다룹니다.
이 가이드에 설명되어 있지 않은 관련 정보를 얻으려면 다음 문서를 읽어 보십시오.

8.3. 표 기타 설명서 - 목록

문서	정보	형식	Asset Manager 설치 폴더에서의 위치
버전 3.x과 9.30의 차이	■ 버전 9.30의 새 기능 목록	인쇄물	\doc\pdf\diff*.pdf
		온라인	\doc\pdf\diff*.pdf
Readme	■ 최신 정보	텍스트	readme.txt

문서	정보	형식	Asset Manager 설치 폴더에서의 위치
릴리스 노트	<ul style="list-style-type: none"> Asset Manager와 함께 제공된 문서 목록 새 기능에 대한 개요 	인쇄물	\doc\pdf\ReleaseNotes*.pdf
		온라인	\doc\pdf\ReleaseNotes*.pdf
설치	<ul style="list-style-type: none"> Asset Manager 프로그램의 목록 지원되는 운영 체제 및 최소 구성 지원되는 DBMS Asset Manager 설치 Asset Manager 버전 4.2.x, 4.3.x 또는 4.4.x에 대한 간단한 업그레이드 	인쇄물	\doc\pdf\Installation*.pdf
		온라인	\doc\chm\install*.chm
데이터베이스 구조	<ul style="list-style-type: none"> 데이터베이스의 테이블, 필드, 링크 및 인덱스 목록 Asset Manager에서 자동으로 트리거되는 에이전트 	텍스트 파일	<ul style="list-style-type: none"> \doc\infos\database.txt \doc\infos\tables.txt
		온라인	\doc\chm\dbstruct*.chm
다른 데이터베이스 버전과의 구조적 차이	<ul style="list-style-type: none"> 변경된 테이블, 필드, 링크 및 인덱스 목록 	텍스트	\doc\infos\diff*.txt
		Html	\doc\infos\diff*.html
Asset Manager 관리	<ul style="list-style-type: none"> Asset Manager Application Designer 가져오기 	인쇄물	\doc\pdf\Administration*.pdf
		온라인	\doc\chm\admin*.chm
고급 사용	<ul style="list-style-type: none"> 데이터 내보내기 	인쇄물	\doc\pdf\AdvancedUse*.pdf
		온라인	\doc\chm\advanced*.chm

XML에 대한 자세한 내용은 웹 사이트를 참조하십시오: <http://www.w3.org/XML/>.

색인

- ;(특수 문자), 100
- !-- (migration.xml), 99
- !-- (migration.xml), 100
- /(문자), 36
- ^(문자), 35
- '(특수 문자), 100
- "(특수 문자), 100
- &(특수 문자), 100
- <(특수 문자), 100
- >(특수 문자), 100
- ||(특수 문자), 100
- 가져오기 스크립트, 111, 77
- 가져오기 - 스크립트, 77
- 계산 문자열, 109
- 고급 사용, 113
- 관리, 113
- 교육, 25
- 구조 - 변경 전파, 37
 - 개요, 38
 - 변환 파일, 41
 - 총돌, 41
 - 프로세스, 39
- 구조적 매개 변수, 46
 - 전파, 111
- 권한 - 제한, 52
- 기능 권한, 75
- 기능적으로 유효(옵션), 68, 59, 58
- 기본값, 47
- 기존 데이터베이스 열기(메뉴), 45, 44, 39
- 내보내기 스크립트, 77
- 내보내기 - 스크립트, 77
- 다른 이름으로 저장(메뉴), 60
- 다음 파일 열기(메뉴), 60
- 데이터베이스
 - 구조, 113
 - 구조 - 버전 간 차이, 106
 - 무결성 - 검증, 67, 51, 31
 - 변환
 - ([관련 항목] 변환)
- 복사, 41
 - Asset Manager Application Designer, 42
 - DBMS 도구, 42
 - 일반 백업 - 문제, 41
- 수동 조정, 33
- 완료, 72
- 유효성 검사, 52
- 지원되지 않는 DBMS, 22
- 차단 및 복사, 71
- 특정 권한 제한, 52

- 프로덕션 단계, 85
- 데이터베이스 구조
 - 변경 - 전파 (참조 구조 - 변경 전파)
 - 정의, 89
- 데이터베이스 변환, 43
- 데이터베이스 변환(메뉴), 45
- 데이터베이스 복구(메뉴), 32 , 32
- 데이터 - 정의, 89
- 디스크 공간, 15
- 레코드의 유효성 확인(옵션), 51 , 32 , 32
- 롤백 세그먼트, 28
- 릴리스 노트, 113
- 마법사, 109
- 마이그레이션
 - Asset Manager - 버전 4.2.x, 4.3.x 또는 4.4.x
 - 시뮬레이션
 - 정의, 87
 - 준비 단계, 23-29
 - 최종 단계, 83-86
 - 최종 변환, 71-72
 - 프로세스, 17-22
 - 마이그레이션 폴더(필드), 67
 - 메시지(창), 57
 - 모든 파일 나열(메뉴), 61 , 57
 - 무결성 - 검증, 51
 - 변환
 - 변환 시스템, 26
 - 새로운 기능, 21
 - 정의, 88
 - 최종 변환, 71
 - 충돌, 49
 - 프로세스, 45
 - 변환 도구, 21
 - 변환 속도, 28
 - 변환 시스템
 - 정의, 88
 - 준비, 26
 - 변환 파일
 - SQL 문, 104
 - 검증, 105
 - 구문, 93
 - 변환 규칙, 92
 - 속성
 - continueonerror, 94

- encoding, 93
- engine, 94
- SYSTEM, 93
- 숫자 필드, 104
- 여러 대상 테이블, 103
- 외래 키, 103
- 요소
 - !--, 99
 - !---, 100
 - MigrationFile, 93
 - PostActions, 99
 - PreActions, 97
 - StartScript, 94
 - Translate, 95
 - 매핑, 95
 - 스크립트, 99
 - 예외, 98
 - 필드, 97
- 용도, 92
- 응용 프로그램 데이터, 104
- 자산, 105
- 정의, 88
- 조인, 102
- 조정, 91 , 43
- 특수 문자
 - ;, 100
 - ', 100
 - ", 100
 - \, 100
 - &, 100
 - <, 100
 - >, 100
 - ||, 100
- 보기, 111 , 78 , 73
- 복구(옵션), 32 , 32
- 복원 가능(옵션), 59
- 분석만, 51
- 브랜드, 50 , 50
- 비즈니스 라인 데이터 - 내보내기, 54
- 빈 데이터베이스로 데이터베이스 복사(메뉴), 42
- 사용 가능한 가이드, 112
- 사용자(필드), 45 , 44
- 사용자 권한, 75
- 사용자 역할, 48

- 사용자 지정 구조 전파(메뉴), 40
- 사용자 프로필, 48
- 새로 만들기(메뉴), 60
- 샘플 데이터 - 내보내기, 54
- 설치, 113
- 소유자(필드), 45 , 44
- 스크립트의 문제 나열(메뉴), 64 , 60 , 57
- 시스템 데이터
 - 내보내기, 54
- 액세스 제한, 75
- 업데이트, 87
- 연결, 84
- 열기(메뉴), 60 , 32 , 32
- 외래 키, 103
- 외부 프로그램
 - 업데이트, 81
- 용어집, 87-90
- 워크플로우(모듈), 36
- 웹 페이지, 111
- 응용 프로그램 데이터
 - ([관련 항목] 구조 - 변경 전파)
 - ([관련 항목] 변환)
 - Asset Manager Script Analyzer를 사용하지 않고 변환할 응용 프로그램 데이터, 111
 - 검증 및 수정, 56
 - 프로세스, 56
 - 내보내기(참조 응용 프로그램 데이터 - 내보내기)
 - 규칙, 54
 - 프로세스, 53
 - 변환 규칙, 48
 - 변환 파일, 104
 - 복원, 66
 - 프로세스, Asset Manager Application Designer 사용, 66
 - 프로세스, Asset Manager Script Analyzer 사용, 66
 - 복원된 데이터 검증, 68
 - 수동으로 변환할 응용 프로그램 데이터 - 목록, 108
 - 수정 - 속도 향상, 59
 - 정의, 89
- 응용 프로그램 데이터 내보내기(메뉴), 53
- 응용 프로그램 데이터 복원(메뉴), 67 , 66 , 61
- 이전 파일 열기(메뉴), 60
- 인덱스, 73 , 47
- 자산, 105
- 작업 폴더(필드), 67 , 66
- 저장(메뉴), 60
- 저장된 프로시저, 73
- 전체 이름, 36
- 정의
 - Asset Manager 데이터베이스 변환, 88
 - Asset Manager 데이터베이스의 구조, 89
 - Asset Manager 프로그램 업데이트, 87
 - 데이터, 89
 - 마이그레이션, 87
 - 변환 시스템, 88
 - 변환 파일, 88
 - 응용 프로그램 데이터, 89
 - 트리거, 89
 - 프로덕션 데이터베이스, 88
- 조달(모듈), 36 , 22
- 조인, 102
- 종료(메뉴), 60
- 지원되는 운영 체제, 15
- 지원되는 환경, 15-15
- 참조, 91-113
- 처리되지 않은 파일 나열(메뉴), 61 , 57
- 추가된 화면, 49
- 카운터, 33
- 캐시, 84
- 테이블
 - 변환 순서, 47
 - 여러 개의 테이블 간 필드 나누기, 100
 - 여러 대상 테이블, 103
- 트랜잭션 로그, 28
- 트리거, 73
 - 정의, 89
- 특수 문자, 100
- 파일에 대해 복원 가능 특성 강제 적용(메뉴), 61
- 평면도 위치, 46
- 프로덕션 데이터베이스, 88
- 필드, 109
 - Feature 전송, 101 , 76
 - 길이, 34
 - 수동으로 변환할 응용 프로그램 데이터를 저장하는 필드, 48

- 숫자 필드, 104
- 여러 개의 테이블 간 나누기, 100
- 필드 이름을 저장하는 필드, 109
- 필수 필드, 46 , 34
- 필드 도움말, 111 , 73
- 필수 링크, 34
- 해당 컨텍스트에서 스크립트 확인(메뉴), 60 , 58

A

- acmig.dtd, 105 , 93
- AQL 쿼리, 109
- Asset Manager Application Designer
 - 구조 - 변경 전파 (참조 구조 - 변경 전파)
 - 구조적 매개 변수 - 전파, 111
 - 데이터베이스 무결성 - 검증, 51 , 31
 - 데이터베이스 - 복사, 42
 - 응용 프로그램 데이터 - 내보내기, 53
 - 응용 프로그램 데이터 - 복원, 66
- Asset Manager Automated Process Manager, 83
- Asset Manager Export Tool, 111
- Asset Manager Script Analyzer
 - .xml 파일, 61
 - 메뉴, 60
 - 스크립트 - 문제, 64
 - 응용 프로그램 데이터 - 검증 및 수정, 56
 - 응용 프로그램 데이터 - 복원, 66
 - 확인되지 않은 응용 프로그램 데이터, 111
 - 확인된 응용 프로그램 데이터, 108
- Asset Manager Web, 111 , 81
- Asset Manager 클라이언트, 85
- Asset Manager 프로그램 - 업데이트, 83-85
 - 프로세스, 84

B

- Basic 스크립트, 109 , 64

C

- config(폴더), 79 , 74
- continueonerror(migration.xml), 94

D

- dbbscripts(폴더), 57 , 57 , 40

DBMS

- 서버 준비, 28
- 지원되는 DBMS, 15
- 지원되지 않는 버전, 22
- DBMS 서버 - 준비, 28
- diff*.*, 106
- diff*.htm, 107
- diff*.txt, 106
- diff*.xml, 108
- dtd(파일 확장자), 93

E

- encoding(migration.xml), 93
- engine(migration.xml), 94
- Exception(migration.xml), 98

F

- Feature, 102 , 101 , 76
- Field(migration.xml), 97
- fromxxx(폴더), 88 , 43

G

- gbbase.xml, 90 , 79 , 74
- gbbase*.*, 111 , 65 , 57 , 46 , 37 , 34 , 18
- Get-It, 111 , 81
- Get-Resources, 111 , 81

H

- HP Connect-It, 111 , 81 , 27
 - 경고, 17
- HP Connect-It 시나리오, 81

I

- infos(폴더), 106

J

- Java Runtime - 설치, 27
- Java - 설치, 27

M

- Mapping(migration.xml), 95
- migration.xml, 88 , 72 , 67 , 62 , 52 , 51 , 41 , 33 , 21 , 21

([관련 항목] 변환 파일)

MigrationFile(migration.xml), 93
 modifications.xml, 67 , 62

N

newdbb.log, 40

P

PostActions(migration.xml), 99
 PreActions (migration.xml), 97

R

Readme, 112
 reference(폴더), 56
 referencenew(폴더), 56

S

SAP Crystal Reports, 111 , 78 , 78
 Script(migration.xml), 99
 sdu.log, 45
 sdu.xml, 105
 sdurest.log, 67 , 66
 sduxprt.log, 53
 SQL 문, 104
 SQL - 문, 104
 StartScript (migration.xml), 94
 SYSTEM(migration.xml), 93

T

Translate(migration.xml), 95

U

up_GetCounterVal(저장된 프로시저), 72 ,
 33

V

Value(속성), 47

X

xerces.jar, 41
 xml(파일 확장자), 93 , 68 , 67 , 66 , 64 ,
 61 , 60 , 57 , 53 , 40
 XML - 편집기, 27

