

IBM® WebSphere Commerce



상점 개발자 안내서

버전 5.4

IBM® WebSphere Commerce



상점 개발자 안내서

버전 5.4

주:

이 책 및 이 책이 지원하는 상품을 사용하기 전에 반드시 주의사항 절의 정보를 읽으십시오.

제 2 판(2002년 5월)

이 책은 다음 제품에 적용됩니다.

- Windows NT® 및 Windows® 2000용 IBM® WebSphere® Commerce Business Edition, 버전 5.4
- AIX®용 IBM WebSphere Commerce Business Edition, 버전 5.4
- Solaris용 IBM WebSphere Commerce Business Edition, 버전 5.4
- Windows NT 및 Windows 2000용 IBM WebSphere Commerce Studio, Business Developer Edition, 버전 5.4
- Windows NT 및 Windows 2000용 IBM WebSphere Commerce Professional Edition, 버전 5.4
- AIX용 IBM WebSphere Commerce Professional Edition, 버전 5.4
- Solaris용 IBM WebSphere Commerce Professional Edition, 버전 5.4
- Windows NT 및 Windows 2000용 IBM WebSphere Commerce Studio, Professional Developer Edition, 버전 5.4

새 개정판에 별도로 명시하지 않는 한 위 제품 목록의 후속 릴리스와 수정에 적용됩니다. 제품 레벨에 맞는 올바른 버전을 사용하십시오.

책에 대한 주문은 한국 IBM 담당자나 해당 지역의 IBM 지방 사무소로 문의하십시오. 다음 주소에서는 책을 구비하고 있지 않습니다.

IBM은 귀하의 의견을 환영합니다. 다음 중 한 가지 방법으로 귀하의 의견을 보내실 수 있습니다.

1. 아래로 전자 우편을 보내십시오.

ibmkspoe@kr.ibm.com

2. 우편으로 보내실 경우에는 다음 주소로 우송해 주십시오.

135-270

서울특별시 강남구 도곡동 467-27, 군인공제회관빌딩

한국 아이.비.엠 주식회사

고객만족센터

IBM에 정보를 보내는 경우, IBM은 귀하의 권리를 침해하지 않는 범위 내에서 적절하다고 생각하는 방식으로 귀하가 제공한 정보를 사용하거나 배포할 수 있습니다.

목차

시작하기 전에	vii
이 책에 사용된 규칙	vii
참조 정보.	viii

제 1 부 개요 1

제 1 장 상점 아키텍처 개요	3
온라인 상점이란?	3
상점 작성	3
상점 아키텍처	4

제 2 장 상점 개발	7
상점 개발 옵션	7
견본에 따른 상점 작성	7
새 상점 자원을 개발하여 상점 작성.	8
견본 상점과 새 상점 자원의 조합을 사용하여 상점 작성	8
상점 아카이브	8
견본 상점 아카이브	9
상점 아카이브 사용 결정.	10
상점 개발 도구	10
상점 입구 개발 도구	10
상점 데이터 개발 도구.	11
백오피스 개발 도구.	13
상점 개발자의 역할.	13

제 2 부 상점 입구 개발 15

제 3 장 상점 입구 개발	17
상점 입구 아키텍처.	17
기본 명령 및 보기	18
상점 페이지 작성	18
상점 페이지 목록 개발	18
명령 및 보기 URL 목록 개발	23
JSP 파일 이름 연관 보기.	24

제 3 부 상점 데이터 개요 27

제 4 장 상점 데이터	29
상점 데이터의 개념.	29
상점 데이터 정보 모델.	29
상점 데이터 자원	30
상점 데이터 아키텍처	31

상점 데이터 아키텍처 및 견본 상점	34
데이터 작성을 위한 도구	34
WebSphere Commerce 로더 패키지.	34
상점 서비스	35
관리 콘솔.	35
WebSphere Commerce 액셀러레이터	35
조직 관리 콘솔	35
도구 및 상점 데이터 정보 요약 표	35

제 4 부 상점 데이터 개발 39

제 5 장 사이트 자원	41
WebSphere Commerce의 사이트 자원 이해	41
언어.	42
구성원 속성	42
속성 유형	42
구성원 그룹 유형	42
사용자	42
조직.	42
역할.	43
수량 단위 변환	43
수량 단위	43
세금 유형	43
계산 사용법	43
통화.	43
번호 사용법	43
항목 유형	44
장치 형식	44

WebSphere Commerce에서 사이트 자원 작성. 44

제 6 장 상점 자원	45
WebSphere Commerce의 상점 자원 이해	45
상점 엔티티	46
WebSphere Commerce에서 상점 자원 작성	46
XML 파일에 상점 데이터 자원 작성.	52

제 7 장 명령, 보기 및 URL 레지스트리 데이터	53
WebSphere Commerce에서 명령, 보기 및 URL 등 록	53
XML 파일을 작성하여 명령, 보기 및 URL 등록 57	

제 8 장 카탈로그 자원	59
WebSphere Commerce의 카탈로그 이해	60

카탈로그	60	장기 구매 계약.	109
카탈로그 그룹	60	거래 계약	109
카탈로그 항목	61	규정	110
상품.	61	비즈니스 정책	110
항목.	61	침부	110
패키지	61	주문 항목	111
번들.	62	WebSphere Commerce에서 기본 장기 구매 계약	
동적 제품군	62	자원 작성	111
상품 세트.	62	비즈니스 정책 XML 파일 작성	115
속성.	62	기본 장기 구매 계약 XML 파일 작성	119
속성값	63	제 11 장 이행 자원	121
패키지 속성	63	WebSphere Commerce에서 이행 자원 이해	122
패키지 속성값	63	서비스 센터.	122
WebSphere Commerce에서 카탈로그 자원 작성	63	수량증.	123
마스터 카탈로그.	64	RaDetail.	123
상점 카탈로그 자원 표시.	84	재고	123
탐색 가능 카탈로그.	87	운송 협정	123
카탈로그 주기 작성.	87	기타 이행 자원.	124
2차 카테고리에 상품을 추가.	89	WebSphere Commerce에서 이행 자원 작성	126
WebSphere Commerce에서 카탈로그 자원 관리	93	상점 이행 자원 작성.	128
카탈로그 그룹	93	제 12 장 캠페인 자원	129
카탈로그 항목	94	WebSphere Commerce의 캠페인 이해.	129
상품 관리 도구	95	WebSphere Commerce의 캠페인 자원 작성	130
Catalog Manager	96	제 13 장 지불 자원	131
제 9 장 가격 책정 자원	99	XML 파일을 사용하여 지불 자원 작성.	131
WebSphere Commerce의 가격 책정	99	제 14 장 언어 자원	135
판매 설정	100	WebSphere Commerce의 언어 자원 이해	135
판매 가격	100	기본 언어	135
거래 포지션 컨테이너.	100	지원 언어	136
규정	100	대체 언어	136
가격 책정 규정의 유형	100	WebSphere Commerce에서 언어 자원 작성	136
거래 계약	101	제 15 장 통화 자원	137
참석자.	101	WebSphere Commerce의 통화 자원 이해.	138
참석자 역할.	101	통화 포맷	138
장기 구매 계약.	101	숫자 사용법.	138
비즈니스 정책	102	통화 포맷 설명.	139
가격 책정 정책.	102	지원 통화	139
카탈로그 항목 운송	102	통화 변환 규칙.	139
기타 가격 책정 자원	103	카운터 통화.	139
WebSphere Commerce에서 가격 책정 자원 작성	103	WebSphere Commerce에서 통화 자원 작성	139
XML 파일에 가격 책정 자원 작성	106	XML 파일을 사용하여 통화 자원 작성.	143
제 10 장 장기 구매 계약 자원	107	제 16 장 측정 자원 단위	145
WebSphere Commerce의 기본 장기 구매 계약 이			
해	108		
계정(비즈니스 계정)	108		

WebSphere Commerce에서 측정 단위 이해 . . .	146
수량 단위 및 수량 단위 포맷	146
WebSphere Commerce에서 측정 단위 작성 . . .	147
제 17 장 관할 구역 자원	149
WebSphere Commerce의 관할 구역 자원 이해	150
WebSphere Commerce에서 관할 구역 자원 작성	150
제 18 장 운송 자원	151
WebSphere Commerce의 운송 자원 이해 . . .	152
운송 방법	152
계산 코드	152
관할 구역 및 관할 구역 그룹	153
WebSphere Commerce에서 운송 자원 작성 . . .	153
XML 파일을 사용하여 운송 자원 작성 . . .	165
운송 이행 자원 작성	167
상점-카탈로그-운송 자원 작성	169
기본 운송 방법 작성	170
제 19 장 세금 자원	171
WebSphere Commerce에서 세금 자원 이해 . . .	171
세금 카테고리	172
계산 코드	172
관할 구역 및 관할 구역 그룹	174
WebSphere Commerce에서 세금 자원 작성 . . .	174
XML 파일을 사용하여 세금 자원 작성 . . .	187
세금 이행 자원 작성	189
상점-카탈로그-세금 자원 작성	191
제 20 장 할인 자원	193
WebSphere Commerce의 할인 이해	193
계산 코드	194
WebSphere Commerce에서 할인 자원 작성 . . .	195
제 21 장 재고 자원	197
WebSphere Commerce의 재고 자원 이해 . . .	198
ATP 재고	198
비ATP 재고	200
WebSphere Commerce에서 재고 자원 작성 . . .	201
제 22 장 주문 자원	203
WebSphere Commerce의 주문 자원 이해 . . .	203
주문 및 주문 항목	203
주문 항목	205
WebSphere Commerce에서 주문 자원 작성 . . .	206
제 23 장 고객 및 판매자 자원	207
WebSphere Commerce의 고객 자원 이해 . . .	207
주소 정보	208

관심 목록	208
WebSphere Commerce의 판매자 자원 이해 . . .	209
상점	210
계정	210
장기 구매 계약	210
상품 세트	210
가격 목록	211
카탈로그	211
서비스 센터	211
재고 항목	212
WebSphere Commerce의 구성원 자원 이해 . . .	213
구성원	213
구성원 속성	214
역할	215
WebSphere Commerce에서 구성원 자원 작성 . .	215
<hr/>	
제 5 부 상점에 액세스 제어 추가	217
제 24 장 상점의 액세스 제어	219
WebSphere Commerce의 액세스 제어에 대한 이	해
액세스 제어 정책	220
상점의 액세스 제어	223
상점에 액세스 제어 추가	227
상점에 액세스 제어 작성	232
상점 아카이브의 액세스 제어 파일 편집 . . .	232
<hr/>	
제 6 부 상점 패키지	237
제 25 장 상점 패키지	239
상점 아카이브 작성	239
견본 상점 아카이브 작성	241
<hr/>	
제 7 부 상점 공개	247
제 26 장 완성된 상점 공개	249
WebSphere Commerce의 공개 이해	249
시작 공개	251
사전 공개 확인	252
자원 공개	253
지불 구성	262
로그 파일 공개	262
제 27 장 상점 데이터 로드 개요	265
WebSphere Commerce의 데이터 로드에 대한 이	해
상점 데이터를 로드하기 위한 로더 패키지 명령	269

데이터를 변환 및 추출하기 위한 로더 패키지 명령	286
로더 패키지 명령과 관련된 도구	295
상점 데이터 로드	296
로더 패키지 명령 및 스크립트 사용	296
식별자를 분석하는 예제	297
데이터 로드 예제	304

제 28 장 WebSphere Commerce 데이터베이스

자원 그룹 로드	307
데이터베이스 자원 그룹	307
데이터베이스 자원 로드 순서	308
상점 로드	309
데이터베이스 자원 그룹 로드	316

제 29 장 비즈니스 계정 및 장기 구매 계약 공개

상점 서비스 또는 명령행을 사용하여 비즈니스 계정 및 장기 구매 계약 공개	324
명령을 사용하여 비즈니스 계정 및 장기 구매 계약 공개	324
비즈니스 계정 자원 공개	324
장기 구매 계약 자원 공개	325

제 30 장 상점 입구 자원 및 상점 구성 파일 공개

상점 서비스 또는 명령행을 사용하여 상점 입구 자원 및 상점 구성 파일 공개	327
WebSphere Commerce 서버에 복사하여 상점 입구 자원 및 상점 구성 파일 공개	328

제 8 부 상점에 WebSphere Commerce

특징 추가 333

제 31 장 상점에 고객 지원 추가	335
상점에서의 고객 지원 이해	335
프레임셋 사용	337
고객 지원을 사용하는 고객 모니터링	338
상점에 고객 지원 추가	343

부분 1: 설치 전제 조건	343
부분 2: 견본 상점으로부터 고객 지원 통합 파일 복사	343
부분 3: 상점에 프레임셋 추가	344
부분 4: 고객 이름 및 ID 확보를 위한 코드 추가	346
부분 5: 고객이 찾아보고 있는 페이지 판별을 위한 코드 추가	347
부분 6: 장바구니에 있는 항목 추적을 위한 코드 추가	347
부분 7: 고객 지원으로의 링크 추가	348
부분 8: 고객에게 표시하는 메시지 변경	348

제 32 장 상점에 e-Marketing 지점 추가 351

e-Marketing 지점	351
e-MarketingSpot bean	354
상점 페이지에 e-Marketing 지점 추가	354

제 9 부 부록 357

부록 A. UML 범례 359

부록 B. 데이터 작성	361
견본 상점의 데이터 작성	361
상점 서비스 및 견본 상점	362

부록 C. sarinfo.xml	365
sarinfo.xml 예제	370

부록 D. sarrule.xml	373
sarrule.xml 예제	375

부록 E. 데이터베이스 자원 그룹	377
데이터베이스 자원 그룹 종속성	378

부록 F. 주의사항	383
상표	385

시작하기 전에

IBM WebSphere Commerce 상점 개발자 안내서는 WebSphere Commerce 상점 아키텍처 및 상점 개발 처리에 대한 정보를 제공합니다. 특히, 다음과 같은 주제에 대한 정보가 들어 있습니다.

- 상점 개발자 옵션
- 상점 아카이브
- 상점 개발 도구
- 상점 입구 개발
- 상점 데이터 개발
- 상점 데이터 아키텍처
- 상점 데이터 정보 모델
- 상점에 액세스 제어 추가
- 상점 패키지
- 상점 공개
- WebSphere Commerce 특징을 상점에 추가

이 책에 사용된 규칙

이 책에서는 다음과 같은 규칙이 사용됩니다.

굵은체는 명령이나 필드 이름, 아이콘 또는 메뉴 선택사항과 같은 GUI(Graphical User Interface) 제어 문자를 나타냅니다.

모노체는 보이는 그대로 정확하게 입력해야 하는 텍스트의 예를 표시합니다.

기울임꼴은 사용자가 값을 대체해야 하는 변수 및 강조에 사용됩니다.



이 아이콘은 추가 정보(태스크 완료에 도움을 주는 추가 정보)를 표시합니다.

- ▶ **NT** Windows NT 고유 정보를 나타냅니다.
- ▶ **2000** Windows 2000 고유 정보를 나타냅니다.
- ▶ **AIX** AIX 고유 정보를 나타냅니다.
- ▶ **Solaris** Solaris Operating Environment 고유 정보를 나타냅니다.
- ▶ **400** IBM eserver iSeries™ 400®(이전에 AS/400®이라고 함) 고유 정보를 나타

냅니다.

▶ **Linux** Linux 고유 정보를 나타냅니다.

▶ **Business** WebSphere Commerce Business Edition에 고유 정보를 나타냅니다.

▶ **Professional** WebSphere Commerce Professional Edition에 고유 정보를 나타냅니다.

참조 정보

이 책은 앞으로 갱신될 수 있습니다. 갱신사항은 WebSphere Commerce 웹 사이트를 확인하십시오.

▶ **Business** http://www.ibm.com/software/webservers/commerce/wc_be/lit-tech-general.html

▶ **Professional** http://www.ibm.com/software/webservers/commerce/wc_pe/lit-tech-general.html

갱신은 새 정보를 포함합니다.

제 1 부 개요

제 1 장 상점 아키텍처 개요

이 장은 WebSphere Commerce 서버 상점 아키텍처에 대해 소개합니다.

온라인 상점이란?

온라인 상점은 인터넷 기술을 이용하여 상품이나 서비스를 판매하거나 교환하는 상점입니다. 온라인 상점은 고객에게 상품을 보여주고 구매할 수 있도록 하는 웹 페이지 콜렉션으로 구성됩니다. 홈페이지는 고객을 상점에 소개하고, 고객을 상품과 서비스로 안내합니다. 온라인 카탈로그 페이지는 상품을 함께 그룹으로 묶고, 상품에 대한 추가 정보를 찾을 수 있는 상품 페이지로 고객을 안내합니다. B2C 상점의 장바구니 페이지는 실제 장바구니와 같은 역할을 수행합니다. 구매하고자 하는 상품을 장바구니에 담은 다음, 주문 시작 페이지에서 상품에 대해 지불합니다. B2B 사이트의 특정 페이지에서는 주문 목록과 RFQ를 제출할 수 있습니다.

JSP(JavaServer Page) 기술을 사용하여 상점 페이지를 작성합니다. 개별 페이지는 정적 콘텐츠용 HTML, 입력 데이터 및 복잡한 데이터 표시를 핸들하는 클라이언트측 JavaScript™, WebSphere Commerce 서버 명령과 기타 보기를 호출할 URL 및 동적 콘텐츠를 생성할 Java™ 코드와 JSP 태그를 포함합니다. WebSphere Commerce Studio 및 WebSphere Commerce와 함께 들어 있는 commerce 데이터 bean 세트는 JavaServer 페이지 파일에서 사용 가능하며, 상품의 가격이나 속성과 같은 데이터베이스의 정보에 액세스할 수 있게 합니다.

상점은 또한 기능하는 상점을 작성하는 데 필요한 데이터베이스 자원으로 구성됩니다. 예를 들어 기능하는 상점은 카탈로그, 세금, 운송 및 통화에 대한 데이터를 포함해야 합니다.

상점 작성

온라인 상점은 다음 자원으로 구성됩니다.

- 상점 입구

상점의 외부 모습이나 고객에게 표시되는 모습으로 상점 입구로 알려져 있습니다. 상점 입구는 HTML 페이지, JSP 파일, 스타일 시트, 이미지, 그래픽 및 기타 멀티미디어 파일 유형과 같은 웹 자원으로 구성됩니다.

이 안내서는 상점 페이지를 빌드하는 JSP 파일 작성과 관련된 개념과 태스크를 설명합니다. 추가 정보는 17 페이지의 제 3 장 『상점 입구 개발』을 참조하십시오.

- 백오피스

고객이 볼 수 없는 상점의 모습, 명령, 사용자 정의된 코드 및 상점 입구에서 고객이 상품을 구매할 수 있게 하는 비즈니스 로직 구현은 백오피스로 알려져 있습니다.

비즈니스 로직이나 사용자 정의된 코드 작성에 대한 자세한 정보는 *IBM WebSphere Commerce 프로그래머 안내서*를 참조하십시오.

- 상점 데이터

상점을 구성하는 데이터 자원입니다. 올바르게 운영하려면 상점은 모든 고객 활동을 지원하도록 제 자리에 데이터를 갖고 있어야 합니다. 예를 들어, 고객이 구매하려면 상점에는 판매할 상품의 카탈로그, 주문을 핸들할 처리, 요청을 수행할 재고 및 적절한 운송 처리가 포함되어야 합니다. 또한 처리 및 지불 징수 방법이 있어야 합니다. 상점 데이터 작성에 관련된 개념과 태스크는 39 페이지의 제 4 부 『상점 데이터 개발』에서 설명합니다.

상점 아키텍처

WebSphere Commerce 상점 아키텍처는 다음 구성요소로 구성됩니다.

- WebSphere Commerce 서버
- WebSphere Commerce 서버 인스턴스
- 상점 구성

WebSphere Commerce 서버

WebSphere Commerce 서버는 e-commerce 솔루션의 상점 및 상업 관련 기능을 핸들하는 서버입니다. 상점 입구 자원 및 백오피스 비즈니스 로직은 WebSphere Commerce 서버 내의 웹 응용프로그램에 있습니다. WebSphere Commerce는 사용자가 사용하거나 또는 직접 작성할 수 있도록 기본 웹 응용프로그램을 제공합니다.

웹 응용프로그램은 한 상점의 자원이나 복수 상점의 자원을 포함할 수 있습니다. 웹 응용프로그램이 복수 상점 입구와 백오피스를 포함할 때, 개별 상점의 자원은 상점 디렉토리(storedir)에 의해 분리됩니다.

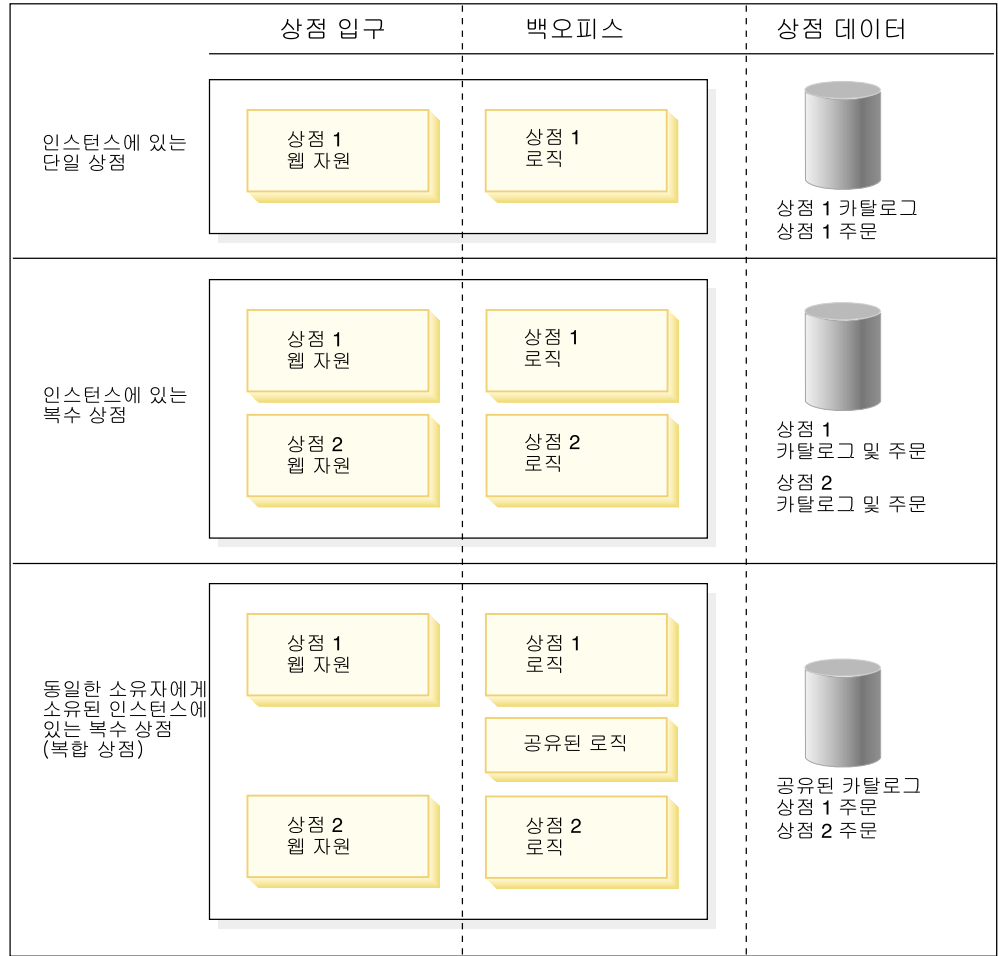
WebSphere Commerce 서버 인스턴스

WebSphere Commerce 서버 인스턴스는 연관된 데이터베이스가 있는 WebSphere Application Server 응용프로그램입니다. 한 인스턴스는 복수 상점을 지원할 수 있습니다. 한 인스턴스의 모든 상점은 같은 데이터베이스를 공유하고 예를 들어, 카탈로그, 이행 또는 수령증과 같은 일부 데이터 유형을 공유할 수도 있습니다. 한 인스턴스의 모든 상점은 같은 EJB 컨테이너를 공유합니다.

상점 구성

WebSphere Commerce는 여러 상점 구성을 지원합니다. 즉, WebSphere Commerce를 사용하면 한 인스턴스에 단일 상점을 작성하거나 별도 상점 입구로 한 인스턴스에 복수 상점, 백오피스 및 상점 데이터를 작성할 수 있습니다. 또는 백오피스를 공유하고

카탈로그를 공유하는 별도 상점 입구로 한 인스턴스에 복수 상점을 작성할 수도 있습니다. 다음 도표는 일부 가능한 상점 구성을 설명합니다.



주: 각 상점은 고유한 식별자를 갖습니다. WebSphere Commerce의 사용권 부여는 작성할 수 있는 상점 수에 대한 특정 한계를 설정합니다. 추가적인 권리를 구입할 수 있습니다. 자세한 정보는 사용권 계약을 참조하십시오.

제 2 장 상점 개발

이 장은 WebSphere Commerce의 상점 개발 처리 개요를 제공합니다.

상점 개발 옵션

WebSphere Commerce는 사이트를 개발할 여러 가지 옵션을 제공합니다.

- 견본에 따른 상점 작성
- 새 상점 자원을 개발하여 상점 작성
- 견본 상점과 새 상점 자원의 조합을 사용하여 상점 작성

견본에 따른 상점 작성

WebSphere Commerce에서 온라인 상점을 작성하는 가장 빠르고 가장 쉬운 방법은 WebSphere Commerce가 제공된 견본 상점들 중 하나를 복사하여 사용자에게 적합하게 사용자 정의하는 것입니다.

견본 상점

WebSphere Commerce에는 상점 작성의 기초로 사용할 수 있는 완전한 기능을 갖춘 여러 개의 온라인 견본 상점이 포함됩니다. B2C 상점과 B2B 사이트를 모두 포함하는 이들 견본 상점은 오늘날 최고 전자 상거래 사이트에서 가장 공통적으로 사용되는 많은 기능을 구현하며, 필요한 모든 상점 자원을 제공합니다. WebSphere Commerce가 제공되는 온라인 상점에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오.


견본 상점부터 시작하는 이유: 처음부터 WebSphere Commerce로 새 온라인 상점을 작성할 수는 있겠지만, 상점의 기초로서 견본 상점 중 하나의 사본을 사용하면 기능적인 상점을 보다 빠르게 작성할 수 있습니다.


WebSphere Commerce는 기능적인 상점을 작성하기 위해 WebSphere Commerce 서버 데이터베이스로 일정한 데이터를 로드할 것과 이 데이터가 스키마가 결정한 순서로 로드될 것을 요구합니다. 견본 상점에는 WebSphere Commerce 서버 데이터베이스가 요구하는 순서와 구조의 모든 필수 데이터가 포함되어 있기 때문에 상점의 기초로서 하나의 견본 상점을 사용하면 초기 작성 기간 동안 많은 시간을 절약할 수 있습니다.

견본 상점의 사본을 작성한 후, 상점의 필요에 따라 어느 정도 편집할 수 있습니다. 예를 들어 WebSphere Commerce로 사용 가능한 도구를 사용하여 데이터를 편집하고, WebSphere Commerce Studio를 사용하여 상점 페이지의 룩 앤드 필을 변경할 수 있습니다. 또는 XML 파일이나 데이터베이스를 직접 편집하여 데이터를 보다 광범위하게

변경하고, 상점 페이지를 수정하여 상점의 플로우와 특징을 변경할 수 있습니다. 상점 편집에 관한 자세한 정보는 WebSphere Commerce 온라인 도움말 주제 "상점 데이터베이스 자원 변경"을 참조하십시오.

WebSphere Commerce는 또한 강조표시된 특징의 코드 견본으로 사용하기 위해 설계된 여러 참조 상점을 제공합니다. 참조 상점은 예를 들어, 쿠폰과 같이 온라인 상점의 선택된 특징에 대해 완전히 기능적인 코드를 포함하고 있는 온라인 상점입니다. 참조 상점은 다음 웹 사이트에서 사용 가능합니다.

 http://www.ibm.com/software/webservers/commerce/wc_be/downloads.html

 http://www.ibm.com/software/webservers/commerce/wc_pe/downloads.html

견본 상점에 기초한 상점 작성에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오.

새 상점 자원을 개발하여 상점 작성

모든 사람이 견본 상점에 기초하여 상점을 작성하고자 하지는 않습니다. 예를 들어, 상점 페이지의 플로우가 제공된 견본과 매우 다르거나 WebSphere Commerce 서버 데이터베이스 스키마를 상당히 사용자 정의할 계획이라면 상점 입구, 백오피스 및 상점 데이터 자원을 개발하여 상점을 작성하려 할 수 있습니다. WebSphere Commerce가 제공된 도구 목록에 대한 자세한 정보는 10 페이지의 『상점 개발 도구』를 참조하십시오.

견본 상점과 새 상점 자원의 조합을 사용하여 상점 작성

견본 상점의 조합 사용 및 새 상점 자원의 개발은 자신에게 가장 알맞은 상점 개발 방법이 될 수 있습니다. 예를 들어, 견본 상점 중 하나의 데이터베이스 자원 일부가 상점의 요구와 거의 일치하지만 상점 페이지의 플로우가 요구와 일치하지 않는 경우, 새 웹 자원 전체를 개발하는 동안 상점에서 데이터베이스 자원을 복사하여 사용자 정의할 수 있습니다.

상점 아카이브

WebSphere Commerce가 포함된 견본 상점은 상점 아카이브 포맷에 제공됩니다. 상점 아카이브 파일(.sar)은 상점을 작성하는 데 필요한 모든 항목이 들어 있는 압축 아카이브 파일(예: ZIP 파일)입니다. 이것은 쉽게 복사할 수 있는 포맷으로 패키지 및 배달 상점의 매개체로서 우선 사용되며, 다음으로는 새 상점을 작성하는 기초로 사용됩니다. 보거나 검색하거나, 구매할 수 있는 운영 가능한 상점을 작성하기 위해서는 상점 아카이브를 WebSphere Commerce 서버에 공개하기만 하면 됩니다.

일반적으로 상점 아카이브는 다음 파일들로 구성됩니다.

- 웹 자원: HTML 파일, JSP 파일, 이미지 및 그래픽과 같은 상점 페이지를 작성하고 파일을 포함하는 데 사용되는 파일. 웹 자원은 상점 아카이브에 압축 파일로 그룹화됩니다.
- 특성 자원 번들(선택적): 상점 페이지의 텍스트 포함. 상점에서 둘 이상의 언어를 지원할 경우, 자원 번들에는 여러 번들이 포함될 수 있습니다. 즉, 언어당 하나의 번들이 있을 수 있습니다.
- 상점 데이터 자원: 데이터베이스에 로드할 데이터. 상점 데이터 자원에는 캠페인, 카탈로그 항목, 통화, 이행 정보, 가격 책정, 운송, 저장 및 세금 정보와 같은 데이터가 포함됩니다. 상점 데이터 자원 목록에 대한 자세한 정보는 39 페이지의 제 4 부 『상점 데이터 개발』을 참조하십시오.

WebSphere Commerce에 제공된 견본 상점 아카이브의 상점 데이터베이스 자원은 로더 패키지에 대해 올바르게 잘 형성된 XML 파일의 형태를 가지고 있습니다. 상점 아카이브 XML 파일은 휴대용으로 만들어진 것으로 데이터베이스의 특정 인스턴스에 특정한 생성된 1차 키를 포함해서는 안됩니다. 대신 상점 공개시 ID 분석기가 해석하는 내부 별명을 사용합니다. 이러한 규칙을 사용하면 견본 상점 아카이브는 여러 번 복사되고 공개될 수 있습니다. 자세한 정보는 361 페이지의 부록 B 『데이터 작성』을 참조하십시오.

로더 패키지에 대한 자세한 정보는 247 페이지의 제 7 부 『상점 공개』를 참조하십시오.

- 지불 자원: IBM Payment Manager의 구성 정보.
- 설명자: XML 파일 sarinfo.xml은 웹 자원 압축 아카이브 파일, 자원 번들 및 상점 데이터베이스 자원 XML 파일을 포함하여 상점 아카이브를 설명합니다. sarinfo.xml 파일에는 또한 공개 처리 동안 필요한 아카이브 파일에 대한 정보 뿐만 아니라 포함 파일과 일관성 확인 파일의 이름이 포함됩니다. sarinfo.xml은 상점 아카이브에서 유일한 필수 파일입니다.

주: **Business** ToolTech 및 NewFashion 견본 상점 아카이브는 다음 파일도 포함합니다.

- tools_properties.zip
- tools_xml.zip
- runtime_xml.zip

이 파일은 상점을 구성하기 위해 상점 서비스에서 사용됩니다. 이 파일은 변경, 제거 또는 다른 상점으로 복사될 수 없습니다.

견본 상점 아카이브

견본 상점 아카이브 파일(.sar)은 새 상점을 작성하기 위한 기준으로 복사되고 사용되는 상점 아카이브입니다. 견본 상점 아카이브는 여러 번 복사하고 공개할 수 있게 하는 몇 가지 규칙을 포함합니다. 이 규칙은 다음을 포함합니다.

- 생성된 1차 키나 foreign key에 대한 참조 없음: 견본 상점 아카이브는 데이터베이스의 특정 인스턴스에 고유한 생성된 1차 키를 포함하지 않습니다. 대신 상점 공개 시 ID 분석기가 해석하는 내부 별명을 사용합니다. 자세한 정보는 361 페이지의 부록 B 『데이터 작성』을 참조하십시오.

WebSphere Commerce에 제공되는 견본 상점은 견본 상점 아카이브입니다. 이 상점은 상점 서비스의 상점 아카이브 작성 페이지에 있는 견본 목록에서 사용 가능합니다.

상점 아카이브 사용 결정

상점 아카이브는 상점을 패키징하고 배달하기 위한 매체로 설계되었습니다. 상점을 견본으로 사용하여 다른 상점에 배달하고, 다른 서버 또는 플랫폼에 배치하거나 다른 상점을 작성하는 기초로서 사용하고자 한다면, 이를 상점 아카이브 형태로 묶는 것을 고려하십시오.

상점이 견본 상점 중 하나를 거의 기초로 하고 아카이브에 대한 변경사항이 많지 않을 경우, 상점 아카이브를 사용하려 할 수 있습니다.

상점 아카이브 사용을 선택하고 견본 상점을 사용하여 상점을 작성했다면, 상점은 이미 상점 아카이브 포맷이 됩니다. 상점을 개발하는 중의 변경 사항이 상점 아카이브에 반영되었는지 확인하여 상점 아카이브 포맷에서 새 상점을 유지보수할 수 있습니다.

상점 아카이브로 다른 방법을 사용하여 작성한 상점을 패키징할 수도 있습니다. 상점 아카이브 작성에 대한 자세한 정보는 237 페이지의 제 6 부 『상점 패키지』를 참조하십시오.

상점 아카이브를 사용하지 않을 경우

상점의 단일 아카이브를 작성 중이거나 기존의 WebSphere Commerce 스키마에 확장된 변경을 행하는 중이라면 상점 아카이브를 사용하지 않기를 선택할 수 있습니다. WebSphere Commerce 스키마에 대한 변경은 상점 아카이브 및 상자 외부의 WebSphere Commerce 상점 개발 도구에 의해 즉시 지원되지 않습니다. 확장된 스키마에 변경을 수행한 후에 상점 아카이브를 유지보수할 경우, 자세한 정보는 IBM 담당자에게 문의하십시오.

상점 개발 도구

WebSphere Commerce는 상점 개발을 돕는 다양한 도구를 제공합니다. 어떤 도구를 사용하느냐는 상점을 개발하고 패키징하는 데 어떤 방법을 선택하느냐에 따라 다릅니다.

상점 입구 개발 도구


상점 입구 자원 개발에는 견본 상점 페이지 사용자 정의, 기존 페이지로 바꾸기, 새 페이지 작성 또는 이 세 가지의 조합이 포함됩니다.

WebSphere Commerce는 다음 도구를 제공하여 상점 입구 자원을 작성하거나 편집합니다.

- WebSphere Commerce Studio: Commerce Studio에는 HTML, 그래픽, 멀티미디어 및 JSP 파일을 포함하여 상점 입구 자원을 작성하고 편집하는 데 필요한 도구가 포함됩니다. Commerce Studio에 포함된 Page Designer를 사용하면 애니메이션 이미지 뿐 아니라 HTML 또는 JSP 파일을 작성할 수 있습니다. 또한 WebSphere Commerce Studio를 구성하여 선택한 다른 웹 개발 도구를 사용할 수 있습니다. WebSphere Commerce Studio에 사용자가 선호하는 도구를 등록하는 방법에 대한 자세한 정보는 WebSphere Studio 온라인 도움말을 참조하십시오.

상점 아카이브 포맷에서 상점과 함께 작업할 계획이라면, WebSphere Commerce Studio를 사용하여 상점 아카이브 구조는 그대로 둔 채 상점 아카이브의 웹 자원을 Studio 프로젝트로 반입할 수 있습니다. Studio 도구를 사용하여 JSP 파일, HTML 파일 및 이미지를 변경한 후, 파일을 WebSphere Commerce 서버의 상점 아카이브로 다시 반출하고 웹 자원을 다시 공개할 수 있습니다.

상점 아카이브를 유지보수하지 않은 경우, WebSphere Commerce Studio를 사용하여 실제 상점에 직접 파일을 공개할 수 있습니다.

- 상점 서비스: 상점 서비스의 웹 자원 대화 상자를 사용하면 다른 웹 자원 세트를 사용하여 상점 아카이브의 웹 자원 압축 아카이브 파일을 바꿀 수 있거나 선호하는 웹 개발 도구를 사용하여 편집할 수 있는 선택 위치에 기존의 웹 자원을 다운로드할 수 있습니다. 상점 아카이브 포맷으로 상점을 작업 중이라면, 웹 자원 대화 상자를 사용하여 변경된 자원을 상점 아카이브로 돌려보낼 수 있습니다. 상점 구성 페이지는 공개 상점의 JSP 파일에서 다른 특징을 사용하거나 사용 안하게 할 수 있습니다. 현재 상점 서비스는 협업 특징(협업 작업 영역 및 고객 지원) 구성만을 지원합니다. 이 특징은  Business ToolTech 및 NewFashion 견본 상점에 근거한 상점의 구성에만 사용 가능합니다.

상점 입구 자원을 작성하고 편집하기 위한 WebSphere Commerce Studio와 상점 서비스의 도구 사용에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오. WebSphere Commerce의 상점 입구 작성에 대한 자세한 정보는 15 페이지의 제 2 부 『상점 입구 개발』을 참조하십시오.

상점 데이터 개발 도구

상점의 데이터베이스 자원 개발 및 편집을 위한 몇 가지 옵션이 있습니다.

- 상점 서비스

상점 서비스는 상점 아카이브에서 작동하는 브라우저 기반의 도구 세트입니다. 상점 서비스를 사용하여, WebSphere Commerce가 제공된 견본에 기초한 상점 아카이브를 신속하게 작성할 수 있습니다. 상점 아카이브를 작성했다면 상점 서비스를 사용하여 다음 태스크를 수행할 수 있습니다.

- 상점 아카이브를 공개하여 기능적인 상점 작성

- 세금 노트북을 사용하여 세금 설정 변경
- 운송 노트북을 사용하여 운송 설정 변경
- 상점 프로파일 노트북을 사용하여 일반 상점 설정 변경

상점 서비스를 사용하면 상점 아카이브의 모든 상점 데이터 자원을 편집할 수 없습니다. 상점 서비스를 사용하여 편집할 수 있는 자원 목록은 WebSphere Commerce 온라인 도움말 주제 "상점 데이터베이스 자원 변경"을 참조하십시오. 상점 아카이브의 다른 자원을 편집하려면, XML 자원을 직접 편집해야 합니다.

상점 서비스 사용에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오.

상점 서비스를 사용할 경우: 상점 서비스를 사용하여 견본 상점 아카이브를 복사하고 상점 아카이브 포맷의 데이터베이스 자원을 편집하십시오.

- WebSphere Commerce 로더 패키지

WebSphere Commerce 로더 패키지는 1차적으로 데이터를 준비하고 데이터를 WebSphere Commerce 데이터베이스로 로드하기 위한 유틸리티로 구성됩니다. 로더 패키지를 사용하여 많은 양의 데이터를 로드할 수 있고, WebSphere Commerce 데이터베이스의 데이터를 갱신할 수 있습니다. 로더 유틸리티는 올바르게 잘 형성된 XML을 입력으로 사용하여 데이터를 데이터베이스로 로드합니다. XML 문서의 구성요소는 데이터베이스에 있는 테이블 이름으로 맵핑하고, 구성 요소 속성은 열로 맵핑합니다.

데이터 자원의 개발 및 로드를 위한 로더 패키지 사용에 대한 자세한 정보는 247 페이지의 제 7 부 『상점 공개』를 참조하십시오.

WebSphere Commerce 로더 패키지 사용 시기: WebSphere Commerce 로더 패키지를 사용하여 데이터베이스 자원을 WebSphere Commerce 데이터베이스로 초기에 로드 및 갱신하십시오.

중요: 데이터베이스 스키마를 변경한 경우, 로더 패키지는 데이터를 데이터베이스로 로드하기 위한 유일한 옵션입니다.

- WebSphere Commerce 액셀러레이터

WebSphere Commerce 액셀러레이터는 다양한 상점 운영을 통해 온라인 상점을 유지보수하기 위해 1차적으로 사용되는 온라인 도구 워크벤치입니다. 그러나 WebSphere Commerce 액셀러레이터를 사용하면 이미 데이터베이스의 데이터를 편집할 수 있으므로, 견본 상점 데이터 또는 작성한 데이터로 구성되어 있던, 초기에 데이터베이스를 반입하기만 하면 이것을 상점 개발 도구로 사용할 수 있습니다. WebSphere Commerce 액셀러레이터를 사용하여 편집할 수 있는 데이터베이스 자원 목록에 대해서는 WebSphere Commerce 온라인 도움말 주제 "상점 데이터베이스 자원 변경"을 참조하십시오.

WebSphere Commerce 액셀러레이터를 사용할 경우: WebSphere Commerce 데이터베이스를 미리 반입한 후 WebSphere Commerce 액셀러레이터를 사용해야 합니다.

- 데이터베이스 직접 편집

SQL을 사용하여 데이터베이스를 직접 편집하는 옵션은 항상 사용 가능합니다.

주: SQL은 특정 데이터입니다. Oracle은 다른 SQL 구문을 요구할 수 있습니다. SQL 문은 반드시 데이터베이스 고유의 값을 가지며, 다른 WebSphere Commerce 서버 인스턴스에서 재사용 가능하지 않을 수 있음에 유의하십시오.

백오피스 개발 도구

명령 작성 및 확장, 사용자 정의 코드 작성 및 비즈니스 로직 구현을 포함하여 백오피스 개발 도구는 *IBM WebSphere Commerce 프로그래머 안내서*에서 설명됩니다.

상점 개발자의 역할

상점 개발자는 상점 자원의 세 가지 유형을 모두 개발합니다. 이들은 JavaServer Pages 파일 및 백오피스, 새 명령 및 필요한 모든 사용자 정의 코드를 포함하여 상점 입구 자원을 설계하고 구현합니다. 이들은 또 상점 데이터를 작성하고 WebSphere Commerce가 포함된 표준 기능을 모두 수정할 수 있습니다.

상점 입구와 상점 데이터를 작성 중인 상점 개발자들은 Java, JavaScript, HTML, JSP 기술의 프로그래밍 기술이 있어야 하며, WebSphere Commerce 상점 아키텍처, 상점 데이터 및 상점 아카이브에 익숙해야 합니다.

백오피스를 작성 중인 상점 개발자들은 Java, JavaBeans™, VisualAge for VisualAge®, J2EE 프로그래밍의 프로그래밍 기술을 가지고 있어야 하며, WebSphere Commerce 프로그래밍 모델 및 오브젝트 모델에 익숙해야 합니다. *IBM WebSphere Commerce 프로그래머 안내서*는 백오피스의 사용자 정의에 대한 추가 정보를 제공합니다.

상점 개발자는 데이터베이스 개발자 및 웹 디자이너와 함께 작업할 수 있습니다. 데이터베이스 개발자는 사용자 정의된 상점 기능을 구현하거나 기존의 데이터베이스 정보를 통합할 목적으로 WebSphere Commerce 데이터베이스 스키마를 수정하고 확장합니다. 일반적으로 데이터베이스 개발자는 DB2 또는 Oracle에 대한 데이터베이스 운영자로서의 기술을 가지고 있습니다.

웹 디자이너는 사이트의 외양과 이미지를 디자인하며 상점 개발자와 함께 상점 페이지를 작성합니다. 웹 디자이너는 멀티미디어 도구, HTML 및 JavaScript 기술 경험을 가지고 있어야 하며, JSP 기술에 대한 지식도 가지고 있어야 합니다.

주: 데이터베이스 개발자 및 웹 디자이너의 역할은 WebSphere Commerce 서버에 한정되지 않습니다. 필요한 경우, 데이터베이스 개발자와 웹 디자이너는 상점 개발자 액세스를 지정되어야 합니다.

상점 아카이브를 작성하면, 상점 개발자는 수동으로 또는 상점 프로파일 노트북과 세금 및 운송 노트북을 사용하여 상점 아카이브를 변경할 권한을 가집니다. 그러나 이들은 상점 아카이브를 WebSphere Commerce 서버에 공개할 권한은 없습니다.

제 2 부 상점 입구 개발

제 3 장 상점 입구 개발

이 장은 상점의 외부 부분, HTML 페이지와 같은 웹 자원, JSP 파일, 스타일 시트, 이미지, 그래픽 및 기타 멀티미디어 파일 유형이 고객에게 표시되는 방법을 포함하여, WebSphere Commerce 상점 입구 아키텍처의 개요를 제공합니다.

상점 입구 아키텍처

WebSphere Commerce는 명령 및 보기를 사용하여 상점 입구의 웹 자원을 고객에게 표시합니다.

- 명령은 장바구니에 상품 담기, 주문 처리, 고객 주소록 갱신 또는 특정 상품 페이지 표시와 같은 특정 비즈니스 처리를 수행합니다. 조치가 완료되었을 때 명령은 보기로 리턴합니다.
- 보기는 명령 및 사용자 활동을 표시합니다. 즉, 보기는 고객에게 상점 페이지(JSP 파일)를 보여줍니다. 보기가 JSP 파일을 호출하도록 하려면, JSP 파일 이름이 보기 레지스트리(VIEWREG) 테이블에 보기와 함께 등록되어야 합니다. 해당 JSP 파일은 WCS Stores webapp doc 루트 밑의 상점의 서브디렉토리(storedir)에 JSP 파일 이름을 사용하여 저장됩니다.

명령과 보기는 모두 URL을 사용하여 호출됩니다. 예를 들어, 고객이 견본 상점의 장바구니를 누르면, 고객은 WebSphere Commerce 서버로 전해지는 URL `https://hostname/path/OrderItemDisplay?`를 호출합니다. WebSphere Commerce 서버는 OrderItemDisplay 명령을 호출하여 장바구니 페이지가 고객에게 표시됩니다.

예를 들어, 고객이 견본 상점의 도움말을 누르면 고객은 WebSphere Commerce 서버로 전해지는 URL `https://hostname/path/HelpView?`를 호출합니다. WebSphere Commerce 서버는 도움말 페이지로 리턴하는 HelpView를 호출합니다.

WebSphere Commerce 서버는 개별 상점이 해당 명령을 선택적으로 자체 구현할 수 있는 복수 명령을 URL에 맵핑할 수도 있습니다.





유사하게, WebSphere Commerce 서버는 사용자가 단일 보기에 복수 JSP 파일을 맵핑하여 개별 상점이 장치 유형에 대해 다른 JSP 파일 이름을 선택적으로 등록할 수 있도록 합니다.

주: 상품 표시 및 카탈로그 표시 명령은 JSP 파일은 물론, 보기를 리턴합니다. 상품 및 카탈로그를 표시하는 이런 JSP 파일 이름은 카탈로그 데이터에 저장됩니다. 자세한

한 정보는 84 페이지의 『상점 카탈로그 자원 표시』를 참조하십시오. 선택적으로 다른 JSP 파일 이름을 지정하여 상점이 지원하는 개별 구성원 그룹이나 언어에 대해 상품 및 카탈로그를 표시할 수 있습니다.

기본 명령 및 보기

WebSphere Commerce는 상점에서 사용할 수 있는 기본 명령 및 보기를 제공합니다. 이런 기본 명령 및 보기는 `wcs.bootstrap.xml` 파일에 나열되어 있습니다. 부트스트랩 파일은 다음 디렉토리에 있습니다.

-  `drive:\WebSphere\CommerceServer\schema\xml`
-  `drive:\Program Files\WebSphere\CommerceServer\schema\xml`
-  `/usr/WebSphere/CommerceServer/schema/xml`
-  `/opt/WebSphere/CommerceServer/schema/xml`
-  `/opt/WebSphere/CommerceServer/schema/xml`
-  `/qibm/proddata/WebCommerce/schema/xml`

필요한 명령이나 보기가 제공되지 않으면, 사용자가 스스로 작성할 수 있습니다. 명령 및 보기 작성에 대한 정보는 *IBM WebSphere Commerce 프로그래머 안내서*를 참조하십시오.

상점 페이지 작성

상점 입구 작성에서 가장 큰 태스크는 실제 상점 페이지 작성입니다. 상점 페이지에 대한 개발 작업을 시작하기 전에, 다음 계획 활동을 완료해야 합니다.

- 필요한 상점 페이지 목록 개발
- 명령 및 보기 URL 목록 개발
- 보기와 연관되는 JSP 파일 이름

상점 페이지 목록 개발

상점을 작성하는 데 필요한 페이지 목록을 개발하려면, 정의된 모든 비즈니스 처리는 물론, 상점의 비즈니스 및 기능적인 요구사항을 알아야 합니다.

사용 예제에서 작업

많은 사람들이 요구사항을 사용 예제 양식으로 수집합니다. 사용 예제는 상점의 비즈니스 처리를 고객과 제안된 시스템 사이의 대화형 작업 양식으로 정의합니다. 온라인 상점의 경우, 사용 예제는 상점에서 고객 등록, 카탈로그 찾기, 항목 주문에 대한 방법을 정의합니다.

견본 상점의 비즈니스 처리를 자세히 보여주는 사용 예제 세트는 온라인 도움말에서 제공됩니다. 이런 사용 예제는 사용자가 견본 상점의 플로우를 더욱 완전하게 이해하도록 도와주고, 자체 소유 상점의 사용 예제를 작성하려고 하는 경우 안내서로 사용될 수 있습니다.

다음은 등록 사용 예제의 예입니다.

등록 사용 예제: 등록 처리에서 고객은 데이터베이스에 개인정보를 입력할 수 있습니다.

행위자:

- 고객

기본 플로우: 고객이 사이드바에서 등록을 선택합니다. 그러면 시스템이 다음 필드가 있는 페이지를 표시합니다.

- 전자 우편
- 암호
- 암호 확인
- 이름
- 성
- 연령(선택적)
- 성별(선택적)

고객은 위의 필드에 적절한 정보를 입력하고 제출을 선택합니다. 시스템은 시스템에 새 고객을 작성하고 고객의 정보(E1, E2, E3)를 저장합니다. 시스템은 고객이 관리 개인 계정 사용 예제의 처리에 따라 자신의 계정을 관리하도록 프롬프트합니다.

대체 플로우: 없음

예외 플로우: E1: 전자 우편 주소가 이미 존재합니다.

- 전자 우편 주소가 이미 시스템에 있는 경우에 시스템은 사용자에게 다른 전자 우편 주소를 입력할 것인지 묻는 오류 메시지를 표시합니다. 사용 예제는 처음부터 다시 시작합니다.

E2: 필수 필드 누락

- 다음 필드(전자 우편, 암호, 암호 확인, 이름, 성) 중 하나라도 완료되지 않으면, 시스템은 오류 메시지를 발행합니다. 사용 예제는 처음부터 다시 시작합니다.

E3: 올바르지 않은 암호

- 암호가 올바르지 않거나 확인 암호와 일치하지 않으면, 시스템은 경고를 발행합니다.

상점 쇼핑 플로우 결정: 상점의 비즈니스 처리를 보여주는 사용 예제를 개발 또는 다른 방법의 사용에 관계없이 비즈니스 처리가 사용 가능하면, 상점의 쇼핑 플로우를 작성할 수 있습니다.

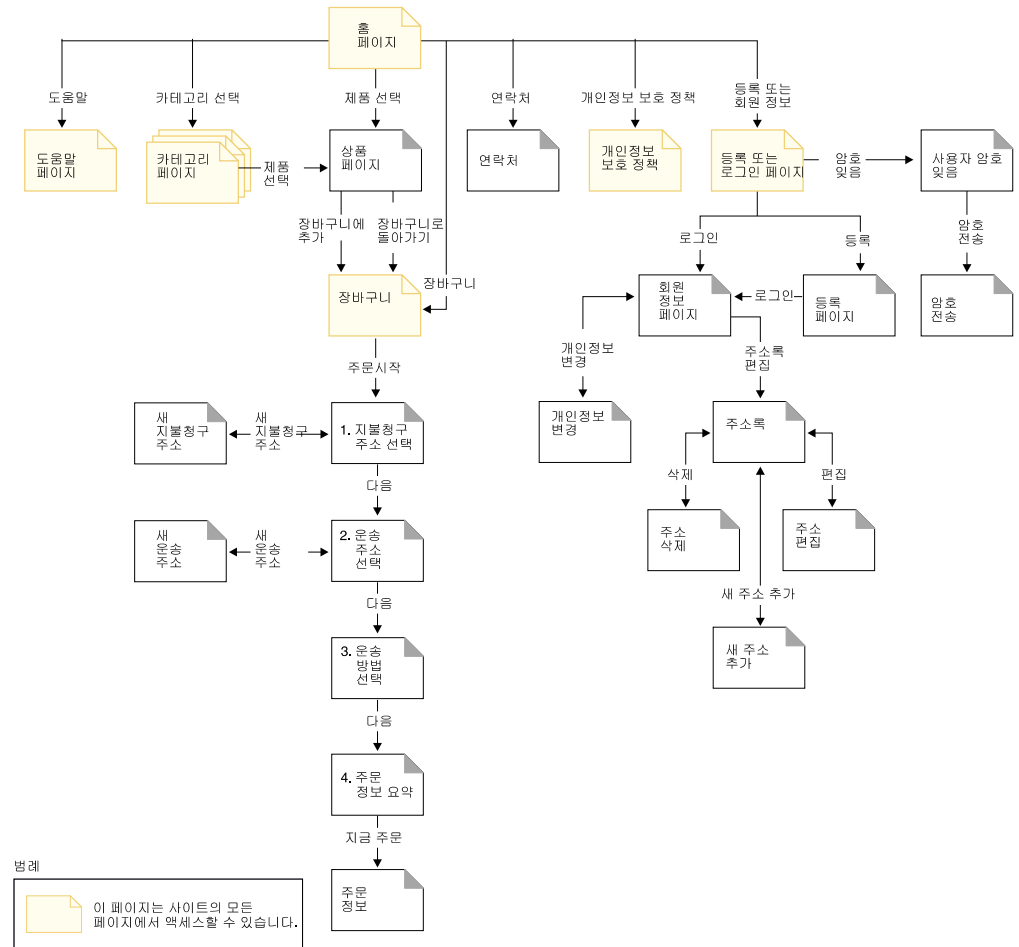
주: 사용 예제는 종종 "고객이 제출을 선택하면 주문 페이지가 표시됩니다"와 같은 플로우 정보가 있기 때문에, 사용 예제는 쇼핑 플로우 도표에 대해 유용한 정보를 제공합니다.

쇼핑 플로우는 고객이 상점에서 이동하는 방법을 보여주는 상점에 대해 정의된 요구사항 및 비즈니스 처리를 반영합니다. 예를 들어, 고객이 홈페이지를 통해 사이트에 들어와서 카탈로그를 찾아보기 전에 등록할 것인지 질문하거나, 등록하지 않고 게스트로서 고객이 카탈로그를 보게 할 수 있습니다. 다른 쇼핑 플로우가 고객이 구매할 때마다 모든 주문 단계를 완료하도록 요구하는 반면, 일부 쇼핑 플로우는 고객이 "바로 주문하기"를 완료하게 합니다. 그렇지 않으면 쇼핑 플로우가 고객이 두 가지 주문을 선택할 수 있도록 합니다.



상점 플로우 도표가 완료되었는지 검증하려면, 상점 사용 예제의 모든 단계가 상점 플로우 도표에 표시되는지 확인하십시오.

InFashion 기본 상점의 쇼핑 플로우의 다음 도표처럼, 쇼핑 플로우를 시각적으로 맵핑하면 사용자는 고객이 상점을 이동하는 방식을 볼 수 있습니다.



InFashion 쇼핑 플로우의 도표는 매우 단순합니다. 고객의 상점 여행에 대한 기본 플로우는 들어 있지만, 오류 시나리오는 들어 있지 않습니다(예를 들어, 고객이 잘못된 암호를 사용하여 로그인하거나 올바른 신용 카드 번호를 입력했을 경우). 그러나 이처럼 단순한 도표도 상점에 필요한 페이지의 목록을 개발할 수 있게 합니다. 시작하려면 쇼핑 플로우 도표에 나열된 모든 페이지의 보기를 작성할 필요가 있습니다.

예를 들어 InFashion 도표에서와 같은 쇼핑 플로우를 사용하여 상점을 작성하려는 경우, 다음 페이지를 작성해야 합니다.

주: 다음 표는 InFashion 상점에서 사용된 보기 이름을 나열합니다.

InFashion 쇼핑 플로우 도표 페이지(고객에 표시된 대로)	해당 보기
홈페이지	StoreCatalogDisplayView
도움말 페이지	HelpView
연락처	ContactView
개인정보 보호 정책	PrivacyView

InFashion 쇼핑 플로우 도표 페이지(고객에 표시된 대로)	해당 보기
등록 또는 로그인 페이지	LogonForm
암호를 잊으셨습니까?	LogoffView
암호 보내기	ResetPasswordForm
회원 정보 페이지	LogonForm
개인정보 변경	UserRegistrationForm
주소록	AddressBookForm
주소 새로 추가	AddressForm
주소 삭제	AddressBookForm
주소 편집	AddressForm
등록 페이지	UserRegistrationForm
장바구니	OrderItemDisplayViewShiptoAssoc
지불 청구 주소 선택	OrderItemDisplayViewShiptoAssoc
새 지불 청구 주소	OrderItemDisplayViewShiptoAssoc
운송 주소 선택	OrderItemDisplayViewShiptoAssoc
새 운송 주소	AddressForm
운송 방법 선택	OrderItemDisplayViewShiptoDsp
주문 정보 요약	OrderDisplayPendingView
주문 확인	OrderOKView

주: InFashion에서 사용되는 많은 보기는 특별히 InFashion용으로 작성되었습니다. 이런 보기는 InFashion 상점 아카이브의 command.xml 파일에 나열되어 있습니다. 자세한 정보는 53 페이지의 『WebSphere Commerce에서 명령, 보기 및 URL 등록』을 참조하십시오.

위의 테이블은 작성하는 데 필요한 기본 페이지 세트만 포함합니다. 작성하는 데 필요한 다른 페이지를 결정하려면, 사용 예제를 보다 자세히 살펴보거나 또는 비즈니스 처리를 정의하는 데 사용된 다른 방법을 볼 수 있습니다.

오류 페이지: 사용 예제의 예외 플로우도 상점을 작성하는 데 필요한 오류 페이지를 결정하는 데 도움을 줍니다. InFashion의 등록 사용 예제는 다음 예외 플로우를 지정합니다.

- 전자 우편 주소가 이미 존재: 전자 우편 주소가 이미 시스템에 있는 경우에 시스템은 사용자에게 다른 전자 우편 주소를 입력할 것인지 묻는 오류 메시지를 표시합니다. 사용 예제는 처음부터 다시 시작합니다.
- 필수 필드 누락: 다음 필드(전자 우편, 암호, 암호 확인, 이름, 성) 중 하나라도 완료되지 않으면, 시스템은 오류 메시지를 발행합니다. 사용 예제는 처음부터 다시 시작합니다.
- 올바른지 않은 암호: 암호가 확인 암호와 일치하지 않으면, 시스템은 경고를 발행합니다.

결과적으로 개별 예외 플로우에 대해 오류 페이지나 오류 메시지를 작성할 필요가 있습니다.

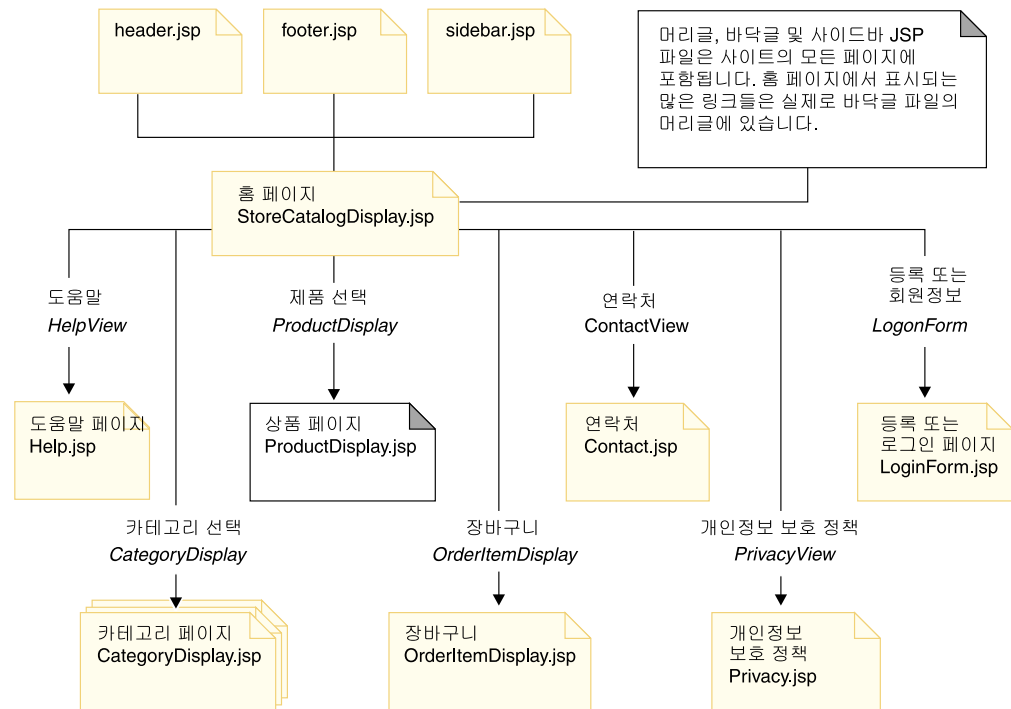
명령 및 보기 URL 목록 개발

InFashion 쇼핑 플로우 도표에서 보여진 것처럼 주문 및 등록과 같은 비즈니스 처리에는 여러 페이지가 필요합니다. 이런 페이지를 단순한 페이지 수집이 아니라 비즈니스 처리나 플로우 작업에 결합하려면, 페이지에 명령과 보기를 포함시켜야 합니다.

필요한 URL 목록 개발

상점을 작성하는 데 필요한 페이지 목록을 개발했던 것처럼 상점의 비즈니스 처리를 구현하는 데 필요한 명령 및 보기 URL 목록도 개발할 필요가 있습니다. 상점의 쇼핑 플로우 도표 및 기본 명령과 보기 목록을 사용하여 개별 조치를 완료하는 데 필요한 URL을 식별하십시오.

전본 상점에서 사용된 명령과 보기 URL을 이해하면, 상점에 필요한 URL을 결정하는 데 도움이 됩니다. 다음 도표는 InFashion 쇼핑 플로우 도표의 일부 조치에 대해 URL을 식별합니다. 자세한 정보는 WebSphere Commerce 온라인 도움말의 전본 상점에 대한 정보를 참조하십시오.



범례

이 페이지는 사이트의 모든 페이지에서 액세스할 수 있습니다.

굵은체 고객 조치
이탤릭체 URL

JSP 파일 이름 연관 보기

WebSphere Commerce 서버는 요청에 대한 응답으로 보기 명령을 사용하여 보기를 구성합니다. WebSphere Commerce 서버는 다음 보기 명령을 제공합니다.

- `HttpForwardViewCommandImpl`: 이 보기 명령은 보기 요청을 JSP 파일에 전달합니다.
- `HttpRedirectViewCommandImpl`: 이 보기 명령은 보기 요청을 다른 URL로 다시 지정합니다.
- `HttpDirectViewCommandImpl`: 이 유형의 보기 명령은 응답 보기를 직접 클라이언트에게 전송합니다. JSP 파일을 호출하지 않습니다. 직접 보기는 제어기 명령이(보기 명령보다는) 출력 응답을 생성하게 합니다.

`HttpForwardViewCommandImpl` 보기 명령을 사용하여 JSP 파일을 직접 표현하십시오. 예를 들어 InFashion에서 사용된 URL을 나타내는 도표에서 도움말 페이지(`Help.jsp`)를 표시하려면, `HelpView`가 보기 레지스트리에 등록되고 `Help.jsp` 및 `HttpForwardViewCommandImpl` 명령에 연관되어야 합니다. 다음 예에서 설명됩니다.

```
<viewreg
viewname="HelpView"
devicefmt_id="-1"
storeent_id="@storeent_id_1"
interfacename="com.ibm.commerce.command.ForwardViewCommand"
classname="com.ibm.commerce.command.HttpForwardViewCommandImpl"
properties="docname=help.jsp"
internal="0"
https="0"
/>
```

인터페이스와 구현 클래스에 대해 완전한 클래스 이름이 사용됨에 주의하십시오.

주: 이 예에서 보기를 호출하는 URL은 제한되지 않습니다. 즉, 모두 URL에 직접 액세스할 수 있습니다. 이 기술을 사용하는 경우, JSP 파일이 공개 데이터만 표현하도록 하십시오.

`HttpForwardViewCommandImpl` 보기 명령을 사용하여 표시 명령에서 리턴된 보기를 표현하십시오. 표시 명령은 데이터베이스에서 데이터를 읽지만, 데이터를 변경하지는 않습니다. 예를 들어, InFashion에서 사용된 URL을 나타내는 도표에서 `OrderItemDisplay` 명령은 `OrderItemDisplayViewShiptoAssoc` 보기를 리턴합니다. 이 보기가 보기 레지스트리에 등록되었을 때, `OrderItemDisplay.jsp` 및 `HttpForwardViewCommandImpl` 이 이와 연관되었습니다. 다음 예에서 설명됩니다.

```
<viewreg
viewname="OrderItemDisplayViewShiptoAssoc"
devicefmt_id="-1"
storeent_id="@storeent_id_1"
interfacename="com.ibm.commerce.command.ForwardViewCommand"
classname="com.ibm.commerce.command.HttpForwardViewCommandImpl"
/>
```

```
properties="docname=OrderItemDisplay.jsp"  
internal="0"  
https="0"  
</>
```

주: 이 예에서 보기를 호출하는 URL은 제한됩니다. 즉, 지정된 액세스를 사용하는 사람들만 URL을 직접 호출할 수 있습니다. 이 보기에 대한 액세스는 명령의 URL에 액세스한 사람에 의해 제어됩니다.

사용하는 모든 표시 명령(예: OrderItemDisplay)과 연관된 모든 보기의 JSP 파일 이름을 연관시켜야 합니다. JSP 파일 이름을 보기와 연관시키는 자세한 정보는 53 페이지의 『WebSphere Commerce에서 명령, 보기 및 URL 등록』을 참조하십시오.

주: ProductDisplay 및 CategoryDisplay는 보기 레지스트리보다는 카탈로그 데이터에서 연관된 JSP 파일 이름을 나열합니다.

HttpRedirectViewCommandImpl 보기 명령을 사용하여 데이터베이스를 변경하는 명령에서 리턴된 보기를 표현하십시오. 보기 다시 지정을 사용하려면, URL에서 &URL= 매개변수를 사용하여 보기 이름을 지정하십시오. 예를 들어, InFashion 견본 상점 AddressForm에 주소 정보를 추가하고 제출을 누를 때, 이는 AddressAdd 명령을 호출합니다. AddressAdd 명령을 호출하는 데 사용된 URL은 &URL= 매개변수를 사용하여 AddressBookForm을 보기로 지정합니다. 이는 AddressBookForm 보기로 다시 지정하게 됩니다. AddressBookForm 보기가 보기 레지스트리에 등록되었을 때, AddressBookForm.jsp 및 HttpForwardViewCommandImpl이 이와 연관되었습니다.

모든 비표시 명령에 대해서는 URL=parameter 기술을 사용해야 합니다. 비표시 명령은 데이터베이스에서 데이터로의 변경을 유발하는 명령입니다.

제 3 부 상점 데이터 개요

제 4 장 상점 데이터

이 장은 WebSphere Commerce 서버 상점 데이터 아키텍처 및 상점 작성 데이터 자원의 개요를 제공합니다. 이 장에서는 WebSphere Commerce 서버 정보 모델도 설명합니다.

상점 데이터의 개념

상점 데이터는 WebSphere Commerce 서버 데이터베이스에 로드된 정보이며, 사용자의 상점이 기능하도록 해줍니다. 상점을 잘 운영하려면 모든 고객 활동을 지원하는 적절한 데이터가 상점에 있어야 합니다. 예를 들어, 고객이 구매하기 위해서는 상점에 판매용 상품 카탈로그(카탈로그 데이터), 주문 처리와 연관된 데이터(세금 및 운송 데이터) 및 요청을 서비스할 재고가 있어야 합니다.

상점 데이터 정보 모델

이 안내서는 정보 모델을 사용하여 상점 데이터가 WebSphere Commerce 서버에서 어떻게 구축되는지를 보여줍니다. WebSphere Commerce 서버 정보 모델은 WebSphere Commerce 서버 데이터 및 오브젝트 모델에 포함된 상위 레벨의 추상화입니다. 데이터 및 오브젝트 모델의 가장 중요한 특징을 강조표시하여 나타내지만, 정보 모델은 스키마와 오브젝트 구현에 특정한 하위 레벨 정보는 포함하지 않습니다.

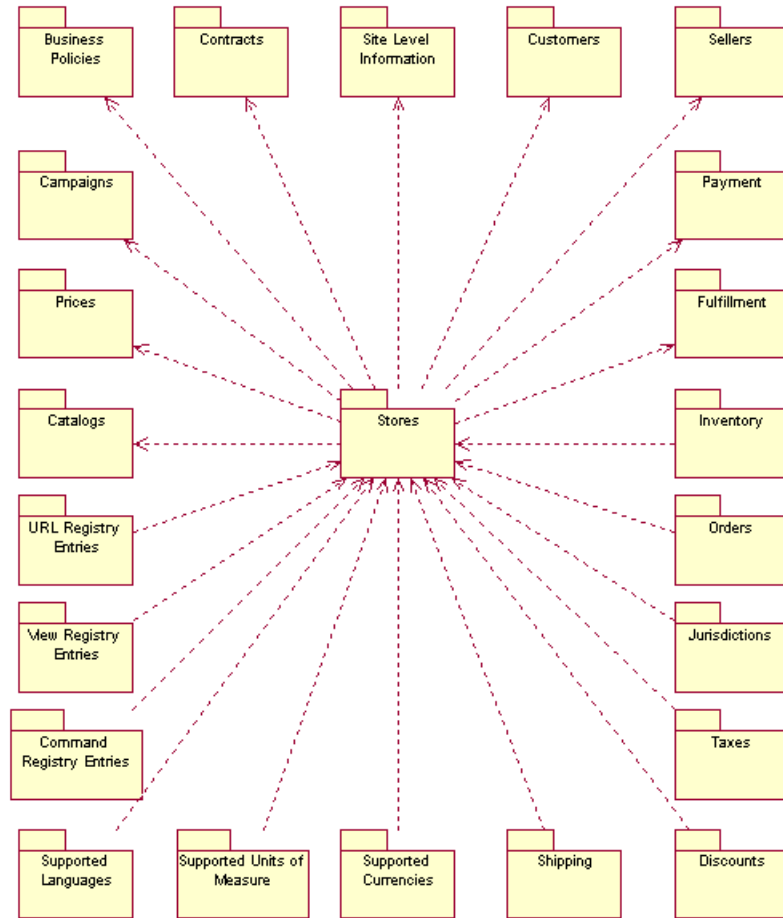
예를 들어, foreign key 쌍과 같은 엔티티 관계 데이터를 포함하는 데이터 또는 오브젝트 모델에 있는 일부 테이블 및 오브젝트는 정보 모델에서 엔티티로 표시되지 않습니다. 대신 이러한 엔티티 관계 데이터는 정보 모델에 있는 엔티티 간의 관계 행에 의해 암시됩니다. 정보 모델 역시 자세한 확장자(구현 결과 개별 테이블에 저장된 엔티티의 추가 데이터 속성. 예를 들어, 상품 설명은 상품 엔티티의 개별적으로 저장된 확장자입니다)를 보여주지 않습니다. 관계 오브젝트 및 자세한 확장자에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말에서 오브젝트 모델을 참조하십시오.



WebSphere Commerce 오브젝트 및 데이터 모델에 대한 자세한 정보는
WebSphere Commerce 온라인 도움말을 참조하십시오.

상점 데이터 자원

다음 도표는 WebSphere Commerce 상점의 데이터 자원을 설명합니다.



이 도표와 상점 데이터 절의 모든 기타 도표들은 WebSphere Commerce 정보 모델의 일부입니다. 정보 모델에 대한 자세한 정보는 29 페이지의 『상점 데이터 정보 모델』을 참조하십시오. 이 도표에서 사용된 규칙에 대한 자세한 정보는 359 페이지의 부록 A 『UML 범례』를 참조하십시오.

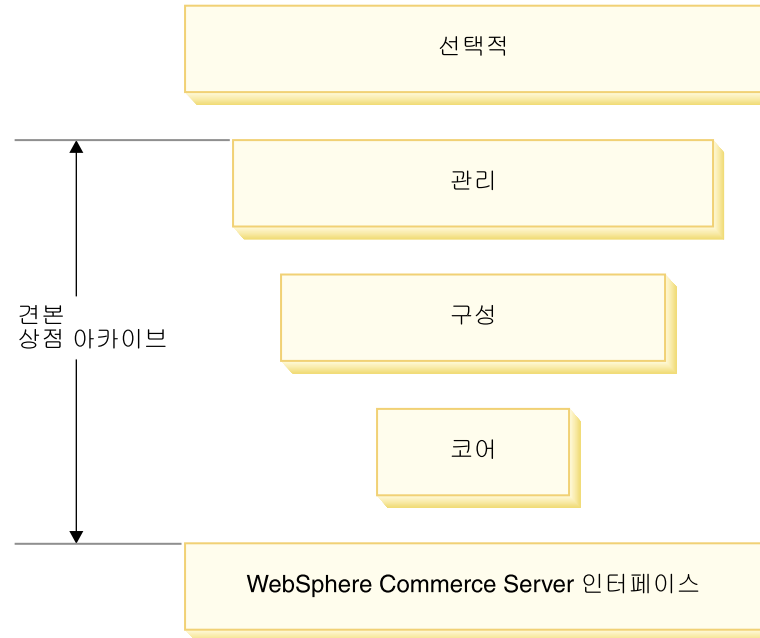
도표에서 화살표의 두 가지 방향에 주의하십시오. 일부 경우에는 화살표가 상점을 향하고 있습니다(예: 통화에서 상점까지). 이 경우에는 선택한 통화 자원이 특정 상점에 독점적이며 상점의 일부입니다. 이 통화 자원들은 이 상점이 지원하는 통화입니다. 상점을 삭제하면 상점이 지원하는 통화의 목록도 삭제됩니다.

화살표가 상점에서 자원(예: 카탈로그)까지를 가리키면 자원을 다른 상점과 공유할 수 있습니다. 하나의 카탈로그를 여러 상점에서 공유할 수 있습니다. 그러나 상점을 삭제해도 카탈로그는 계속 남아 있습니다.

위의 도표에서 설명한 개별 데이터 자원은 39 페이지의 제 4 부 『상점 데이터 개발』에서 자세히 설명됩니다.

상점 데이터 아키텍처

WebSphere Commerce 상점의 데이터는 다음 도표에 나타난 아키텍처와 일치합니다. 30 페이지의 『상점 데이터 자원』의 도표에서 설명된 개별 상점 데이터 자원들은 아래에 설명된 하나 이상의 상점 데이터 유형에 속하는 것으로 분류할 수 있습니다.



WebSphere Commerce 서버 인스턴스

기본 레벨 데이터는 WebSphere Commerce 서버 인스턴스에 포함됩니다. 인스턴스가 작성되면 XML 포맷으로 로드되는 부트스트랩 파일이 정보를 가진 데이터베이스를 대량 자료 반입합니다. 부트스트랩 파일은 다음과 같은 유형의 데이터를 작성합니다.

- 계산 사용법 유형, 신용 카드 유형(브라우저, 전자 우편, I 모드 등), 메시지 유형 및 역할
- 운영자용 기본 ID, WCSADMIN
- 기본 명령, 보기 및 URL
- 기본 비즈니스 정책
- 인스턴스가 지원하는 언어 및 통화
- 상점 소유자로 사용할 수 있는 기본 조직
- 기본 사이트 조직
- 기본 상점 그룹

이 정보는 해당 인스턴스에 있는 모든 상점에서 사용 가능하며, 30 페이지의 『상점 데이터 자원』 도표에서는 사이트 레벨 정보로 식별됩니다.

이러한 정보가 대량 자료 반입되어 있는 부트스트랩 파일 및 데이터베이스 테이블에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오.

코어 데이터

다음 레벨의 상점 데이터는 코어 데이터입니다. 코어 데이터는 다음을 포함하여 상점에 대한 최소한의 데이터를 작성합니다.

- STOREENT 테이블의 상점 식별자. 데이터베이스에 상점을 작성합니다.
- 기본 장기 구매 계약
- 장기 구매 계약 데이터베이스 테이블에 있는 상점 식별자
- 장기 구매 계약 데이터베이스 테이블에서 상점을 소유하고 있는 조직에 대한 구성원 식별자
- STORE 테이블의 상점 디렉토리. 상점 디렉토리는 상점의 웹 자원이 위치한 디렉토리입니다.
- STADDRESS 테이블에서 상점의 주소에 대한 별명 또는 식별자. 별명은 각 상점에 대해 고유합니다.

상점을 작성하기 위해 상점 서비스를 사용한 경우, 이 정보는 새 상점 아카이브로 작성됩니다. 상점 서비스는 사용자가 상점 소유자로서 조치할 수 있는 기본 조직을 선택하게 해주거나, 사용자는 다른 조직을 작성하여 소유자로서 관리 콘솔을 사용하여 조치할 수 있습니다. 상점 서비스를 사용하여 상점을 작성하지 않을 경우에는 이 정보를 로더 패키지를 사용하여 데이터베이스로 로드하거나 데이터베이스를 직접 편집해야 합니다.

30 페이지의 『상점 데이터 자원』의 도표에 있는 상점 데이터는 코어 데이터입니다.

구성 데이터

구성 데이터는 commerce 서버 런타임을 제어합니다. 공통 서버 런타임은 상거래 응용 프로그램이 전개 및 실행되는 프레임워크를 제공합니다. 프레임워크는 프로그래밍 모델, 프로세스 모델, 예외상황 처리, 트랜잭션 제어, 데이터 액세스 및 지속성 모델로 구성됩니다. 공통 서버 런타임은 WebSphere Application Server가 제공하는 런타임 서비스를 활용하여 WebSphere Commerce 서버 응용 프로그램을 지원합니다. 구성 데이터는 상점 페이지를 표시하기 위하여 사용자의 상점이 어떤 명령, 보기 및 JSP 파일들을 사용할 것인지를 결정합니다.

30 페이지의 『상점 데이터 자원』 도표에서 식별된 다음 데이터 자원들은 다음과 같이 구성 데이터로 분류됩니다.

- 명령 레지스트리 항목
- 보기 레지스트리 항목
- URL 레지스트리 항목

관리 데이터

관리 데이터는 판매자가 작성하는 데이터이며, 판매자 사이트의 고객을 위한 읽기 전용입니다. 이러한 상태의 데이터를 판매자가 완전하게 제어하고 있기 때문에 관리 데이터는 콘텐츠 관리 시스템을 통해 관리될 수도 있습니다.

다음은 관리 데이터로 분류되는 30 페이지의 『상점 데이터 자원』 도표에서 식별된 데이터 자원들입니다.

- 캠페인
- 비즈니스 정책
- 장기 구매 계약
- 서비스 센터
- 관할 구역
- 세금
- 할인
- 운송
- 통화
- 측정단위
- 언어
- 카탈로그
- 가격
- 고객
- 판매자
- 지불

운영 데이터

운영 데이터는 사이트와 상호작용한 결과로 사이트의 고객들이 직접 또는 간접적으로 작성하거나 변경한 데이터입니다. 예를 들어, 고객의 주문 목록은 상점을 운영함에 따라 증가하거나 감소하는 재고 레벨처럼 운영 데이터로 간주됩니다. 고객 또한 운영 데이터로 간주됩니다. 판매자가 작성한 데이터도 역시 운영 데이터가 될 수 있습니다.

운영 데이터에 대한 변경은 판매자의 완전한 제어를 받지 않기 때문에 콘텐츠 관리 시스템을 사용하여 이 데이터를 관리하는 것은 적절하지 않습니다.

다음은 운영 데이터로 분류되는 30 페이지의 『상점 데이터 자원』 도표에서 식별된 데이터 자원들입니다.

- 주문 목록
- 재고
- 이행

- 고객

주: 일부 인스턴스에서는 운영 데이터와 관리 데이터 간의 행을 결정하기가 어려울 수 있습니다. 예를 들어, 한 상점에서는 고객과 장기 구매 계약 데이터가 관리 데이터로 여겨질 수 있지만, 다른 상점에서는 동일한 유형의 데이터가 운영 데이터로 여겨질 수 있습니다. 처음 상점에서는 특정한 세트의 고객들(예: 온라인으로 등록할 수 없는 고객들)을 가지고 있기 때문에 자신의 고객 데이터 및 관련 장기 구매 계약을 관리할 수 있습니다. 그러나 두 번째 상점은 고객들의 온라인 등록과 장기 구매 계약의 온라인 작성을 허용합니다.

두 번째 예는 카탈로그 데이터를 포함합니다. 단일 판매자 사이트에서 카탈로그는 관리 데이터로 간주됩니다. MarketPlace 사이트에서 카탈로그 데이터는 운영 데이터로 간주됩니다.

일부 사이트에서는 동일한 데이터 유형의 일부 레코드들이 관리 데이터로 간주되지만 다른 레코드들은 운영 데이터로 간주됩니다. 예를 들어, 기본 장기 구매 계약은 관리 데이터일 수 있지만 온라인으로 협의된 특정한 장기 구매 계약은 운영 데이터입니다.

상점 데이터 아키텍처 및 견본 상점

WebSphere Commerce가 제공된 견본 상점은 상점 데이터 아키텍처에 있는 대부분 유형의 상점 데이터를 포함합니다. 예를 들어, 견본 상점을 사용하여 상점을 작성하거나 견본 상점을 공개하기 전에 WebSphere Commerce 서버 인스턴스가 있어야 합니다. 그런 다음 상점 서비스에서 도구를 사용하여 견본 상점에 기초한 상점을 작성할 때 코어 데이터가 작성됩니다. 견본 상점들은 필요한 모든 구성을 포함하며, 기능적인 상점에 필수적인 대부분의 관리 데이터를 포함합니다. 견본 상점에 기초한 상점을 작성할 때는 WebSphere Commerce 액셀러레이터의 도구를 사용하여 데이터의 일부 설치를 완료하도록 지시받을 수도 있습니다.

데이터 작성을 위한 도구

WebSphere Commerce는 상점 데이터를 작성하고 조작하기 위한 여러 가지 도구를 제공합니다. 이 도구들은 아래에 나열되어 있습니다.

WebSphere Commerce 로더 패키지

로더 패키지는 1차적으로 데이터를 준비하고 데이터를 WebSphere Commerce 데이터 베이스로 로드하기 위한 유틸리티로 구성됩니다. 자세한 정보는 247 페이지의 제 7 부 『상점 공개』를 참조하십시오.

상점 서비스

상점 서비스는 데이터베이스의 실제 데이터라기 보다는 상점 아카이브 형태로 미리 공개된 데이터를 편집합니다. 상점 서비스는 또한 모든 상점 데이터 자원을 데이터베이스에 공개하는 것을 허용합니다. 자세한 정보는 247 페이지의 제 7 부 『상점 공개』를 참조하십시오.

관리 콘솔

관리 콘솔은 관리 운영 및 구성 태스크를 완료해서 사용자의 사이트나 상점을 제어할 수 있도록 해줍니다. 사용자 역시 관리 콘솔을 사용하여 새 조직 및 사용자를 작성할 수 있을 뿐 아니라 사용자에게 역할(상점 개발자, 상점 운영자 및 사이트 운영자 등)을 지정할 수 있습니다. 관리 콘솔은 사용자가 자신의 상점에서 어떤 통지 및 메시지 유형이 사용 가능한지를 식별하게 해줍니다.

WebSphere Commerce 액셀러레이터

WebSphere Commerce 액셀러레이터는 사용자가 여러 상점 자원을 작성하고 유지보수할 수 있게 해주는 온라인 도구의 워크벤치입니다. 대부분의 상점 데이터는 WebSphere Commerce 액셀러레이터에서 도구를 사용하여 작성하고 관리할 수 있습니다. 그러나 일부 경우에는, WebSphere Commerce 액셀러레이터를 사용하여 데이터를 작성하기 전에 상점 서비스나 로더 패키지에서 공개 도구를 사용하여 일정한 데이터를 데이터베이스로 로드한 상태여야 합니다. 추가 정보는 『도구 및 상점 데이터 정보 요약 표』를 참조하십시오.

조직 관리 콘솔

조직 관리 콘솔은 사이트나 상점에 액세스하는 조직을 관리하게 합니다. 조직 관리 콘솔은 구매자의 운영자가 조직 내에서 구매자를 관리하도록 합니다.

도구 및 상점 데이터 정보 요약 표

다음 표는 각 유형의 데이터를 작성하기 위해 어떤 도구를 사용할 수 있는지를 표시합니다.

데이터 작성을 위한 도구	코어 데이터	구성 데이터	관리 데이터	운영 데이터
WebSphere Commerce 로더 패키지	로더 패키지를 사용하여 XML 파일 양식의 코어 데이터를 로드하십시오. 자세한 정보는 47 페이지의 『XML 파일에 상점 데이터 자원 작성』을 참조하십시오.	로더 패키지를 사용하여 XML 파일 양식의 구성 데이터를 로드하십시오. 자세한 정보는 54 페이지의 『XML 파일을 작성하여 명령, 보기 및 URL 등록』을 참조하십시오.	로더 패키지를 사용하여 XML 파일 양식의 관리 데이터를 로드하십시오. 자세한 정보는 관리 데이터 자원과 해당 장을 참조하십시오.	일반적으로 운영 데이터는 로더 패키지로 로드될 수 없습니다.

데이터 작성을 위한 도구	코어 데이터	구성 데이터	관리 데이터	운영 데이터
상점 서비스(상점 아카이브 형태로 미리 공개된 데이터에 대해서만)	상점 서비스를 사용하여 새 상점 아카이브를 작성할 때 코어 데이터가 작성됩니다. 상점 서비스 사용에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오.	사용할 수 없음	상점 서비스를 사용하면 다음과 같이 일정량의 관리된 자원을 작성하고 편집할 수 있습니다. <ul style="list-style-type: none"> • 관할 구역 • 세금 • 운송 • 통화 • 언어 어느 정도 비율의 데이터베이스 자원 상점 서비스를 사용하여 편집하거나 작성할 수 있는지에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말 "상점 데이터베이스 자원 변경"을 참조하십시오.	사용할 수 없음
관리 콘솔	관리 콘솔을 사용하여 상점 소유자로서 활동하는 조직을 작성하십시오.	사용할 수 없음	사용할 수 없음	사용할 수 없음

데이터 작성을 위한 도구	코어 데이터	구성 데이터	관리 데이터	운영 데이터
WebSphere Commerce 액셀러레이터	사용할 수 없음	사용할 수 없음	<p>WebSphere Commerce 액셀러레이터를 사용하여 다음 데이터를 작성하거나 편집하십시오.</p> <ul style="list-style-type: none"> • 캠페인 • 장기 구매 계약 (WebSphere Commerce 액셀러레이터에서 비즈니스 관계 관리 도구를 사용하여 추가 장기 구매 계약을 작성하거나 기존의 장기 구매 계약을 변경하기 전에 기본 장기 구매 계약이 데이터베이스에 있어야 합니다. 로더 패키지 또는 상점 서비스를 사용하여 데이터베이스에서 기본 장기 구매 계약을 작성하십시오.) 	<p>고객은 상점에 등록하거나 상점에서 상품을 구매할 때 운영 데이터를 작성합니다. 그러나 일부 경우에는, 사용자가 WebSphere Commerce 액셀러레이터를 사용하여 고객을 대신해 주문하거나 반품할 수 있습니다.</p> <p>WebSphere Commerce 액셀러레이터를 사용하면 재고를 관리할 수도 있습니다.</p>

데이터 작성을 위한 도구	코어 데이터	구성 데이터	관리 데이터	운영 데이터
WebSphere Commerce 엑셀러레이터 계속	사용할 수 없음	사용할 수 없음	<ul style="list-style-type: none"> 이행 할인 카탈로그 (WebSphere Commerce 엑셀러레이터에서 상품 관리 도구를 사용하여 이동 카탈로그를 작성하거나, 상품 정보를 추가하거나 변경하기 전에 마스터 카탈로그가 데이터베이스에 있어야 합니다. 로더 패키지 또는 상점 서비스를 사용하여 데이터베이스에서 마스터 카탈로그를 작성하십시오.) 가격 	사용할 수 없음
조직 관리 콘솔	사용할 수 없음	사용할 수 없음	사용할 수 없음	고객과 구매자가 상점에 들어오면 고객과 구매자가 작성됩니다. 그러나 조직 관리 콘솔에서 구매자를 승인하거나 새 구매자를 작성할 수 있습니다.

제 4 부 상점 데이터 개발

여기서는 장은 개별 상점 데이터 자원을 자세히 설명합니다. 이 부분의 상점 데이터 자원은 WebSphere Commerce 상점 데이터 아키텍처 구조에 따라 조직되었습니다.

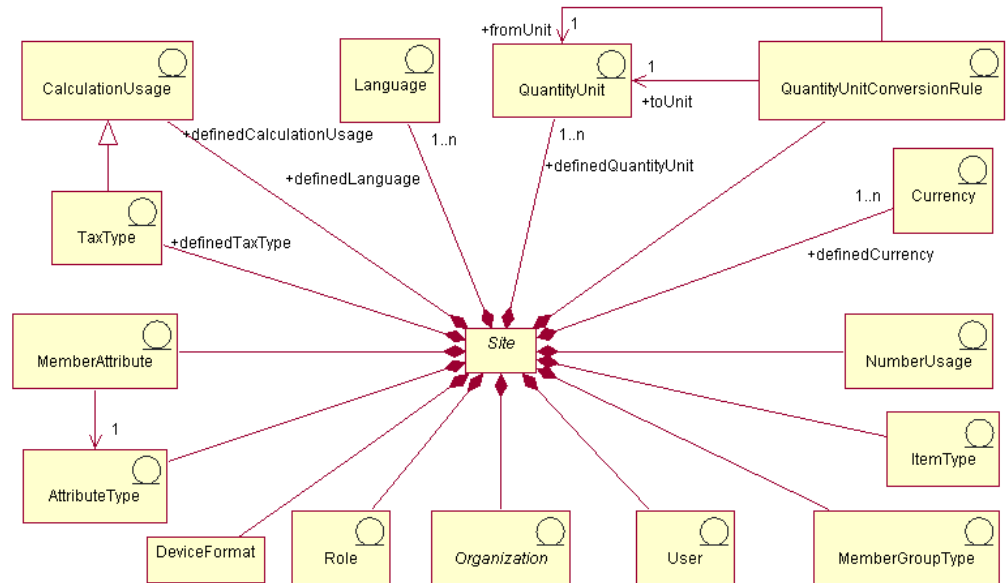
- WebSphere Commerce 서버 인스턴스
 - 사이트
- 코어 데이터
 - 상점
- 구성 데이터
 - 명령 레지스트리
 - 보기 레지스트리
 - URL 레지스트리
- 관리 데이터
 - 공유 자원
 - 카탈로그
 - 가격
 - 장기 구매 계약(비즈니스 정책 포함)
 - 이행
 - 캠페인
 - 지불
 - 상점 자원 독점
 - 지원되는 언어
 - 지원되는 통화
 - 지원되는 측정단위
 - 관할 구역
 - 운송
 - 과세
 - 할인
- 운영 데이터
 - 재고
 - 주문 목록
 - 고객

제 5 장 사이트 자원

각 WebSphere Commerce 서버 인스턴스는 관계 정보의 자체 데이터베이스를 갖습니다. 스키마가 작성된 후, 정보가 있는 데이터베이스 테이블에 대량 자료를 반입하는 부트스트랩 파일이 인스턴스를 작성합니다. 데이터가 로드되면 적절한 데이터베이스 테이블에 사전 로드된 정보를 볼 수 있습니다. 많은 데이터베이스 테이블은 상점이나 상점 그룹에 특정 상점이나 상점 그룹 레벨 정보를 포함합니다. 이 정보는 보통 상점 운영자가 관리합니다. 그러나 일부 테이블은 인스턴스에 있는 모든 상점이 사용할 수 있는 WebSphere Commerce 사이트 레벨 성능을 나타내고, WebSphere Commerce 사이트 운영자가 관리할 수 있는 정보가 있습니다. 이러한 성능이 이 장에서 설명됩니다. 부트스트랩에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오. 상점 특정 자원 정보에 대한 자세한 정보는 45 페이지의 제 6 장 『상점 자원』을 참조하십시오.

WebSphere Commerce의 사이트 자원 이해

다음 도표는 사이트가 포함하는 데이터 유형 및 이 데이터 유형과 사이트의 관계를 나타냅니다.



이 도표에 사용된 규칙에 대한 자세한 정보는 359 페이지의 부록 A 『UML 범례』를 참조하십시오. 이 도표 및 상점 데이터 절의 기타 모두는 WebSphere Commerce 정보 모델의 일부입니다. 정보 모델에 대한 자세한 정보는 29 페이지의 『상점 데이터 정보 모델』을 참조하십시오.

언어

사이트는 LANGUAGE 테이블에 여러 언어를 정의하고 LANGUAGES 테이블에서 이를 설명할 수 있습니다. 각 상점은 일반적으로 STORELANG 테이블에 행을 추가해서 이런 언어의 서브셋을 지원합니다. 사전 정의된 10개의 언어는 독일어, 대만어, 중국어, 일본어, 한국어, 이탈리아어, 프랑스어, 스페인어, 포르투갈어 및 영어입니다.

구성원 속성

구성원 속성은 MBRATTR 테이블에 저장되고, 조직 또는 사용자를 위해 값을 저장할 수 있는 정의된 속성 이름 세트를 나타냅니다. 이런 속성 이름의 예로는 JobFunction, ProcurementCard, SpendingLimit, ReferredBy 및 CountryOfOperation이 포함됩니다. 특정 조직 또는 사용자의 속성값은 MBRATTRVAL 테이블에 저장될 수 있고, 이런 값은 다른 상점 또는 상점 그룹에 대해 달라질 수 있습니다.

속성 유형

속성 유형은 ATTRTYPE 테이블에 저장되고 속성값을 나타내는 데 사용될 수 있는 정의된 데이터 유형을 나타냅니다. 데이터 유형의 예에는 INTEGER, STRING 및 FLOAT이 포함됩니다.

구성원 그룹 유형

구성원 그룹 유형은 MBRGRPTYPE 테이블에 저장되고, 정의된 구성원 그룹 사용법의 세트를 나타냅니다. MBRGRPUSG 테이블에 행을 추가해서 구성원 그룹을 사용법에 지정합니다. 구성원 그룹 사용법의 예에는 AccessGroup(엑세스 제어 정책과 함께 사용하기 위해) 및 UserGroup(고객 그룹과 같은 일반적인 목적을 위해)이 포함됩니다.

사용자

사용자는 인증된 사용자 동일성을 나타냅니다. 사용자는 보통 구매 조직을 대신해서 고객 제출 또는 승인 주문서, 판매 조직이나 상점 레벨 자원을 유지할 판매 에이전트 또는 WebSphere Commerce 서버 인스턴스를 유지할 사이트 운영자를 나타냅니다. 각 사용자는 하나의 사이트와 연결되며 USERS 테이블에 정의됩니다.

조직

조직은 조직 및 조직 내의 조직 단위를 나타냅니다. 조직은 보통 구매 또는 판매를 책임지는 비즈니스 엔티티를 나타냅니다. B2B 구매 조직에서 고객이 제출한 주문 수는 구매 조직을 대신해서 제출한 것으로 기록됩니다. 상점, 카탈로그 및 서비스 센터는 판매의 일정 면을 책임지는 조직에서 소유합니다. 조직은 ORGENTITY 테이블에서 정의됩니다.

역할

역할은 조직 내에서 사용자를 지정할 수 있는 정의된 역할의 세트를 나타냅니다. 예를 들어, 사용자에게 판매 조직 내의 고객 서비스 영업 대표의 역할을 지정하거나, 구매 조직 내의 구매자 승인자의 역할을 지정할 수 있습니다. 기본 역할의 이름이나 설명은 ROLE 테이블에 대량 자료 반입됩니다. 특정 역할에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오.

수량 단위 변환

각 사이트는 수량 변환을 가집니다. 수량 변환은 다른 측정단위 간에 변환하는 데 사용되는 곱하기나 나누기 조작을 나타냅니다. 이는 QTYCONVERT 테이블에 대량 자료 반입됩니다.

수량 단위

수량 단위는 사이트의 측정 단위 세트를 나타냅니다. 수량 단위는 QTYUNIT 테이블에 정의되며 QTYUNITDSC 테이블에서 설명됩니다. 각 상점은 QTYFORMAT 테이블에 행을 추가해서, 원하는 사용법에 따라 개별 측정 단위에서 반올림하고 표시를 위해 포맷할 총계를 지정할 수 있습니다.

세금 유형

세금 유형은 세금을 계산하는 계산 사용법을 나타냅니다. 판매세 및 운송세는 세금을 계산하는 두 개의 다른 계산 사용법입니다. 세금 유형은 TAXTYPE 테이블에서 정의됩니다.

계산 사용법

계산 사용법은 OrderPrepare 명령으로 수행될 수 있는 다른 종류의 계산을 나타냅니다. 계산 사용법은 할인, 운송, 판매세, 운송세 및 e-쿠폰용으로 정의됩니다. 계산 사용법은 CALUSAGE 테이블에서 정의됩니다.

통화

각 사이트는 SETCURR 테이블에서 통화 수를 정의하고 SETCURRDSC 테이블에서 이를 설명합니다. 각 상점은 지원되는 개별 통화에 대해 한 행씩 CURLIST 테이블에 행을 추가해서 이런 통화의 서브세트를 지원합니다.

번호 사용법

번호 사용법은 번호에 의도된 사용법을 나타냅니다. 상점들은 숫자 사용법에 따라 표시하는 숫자에 대해 서로 다른 반올림 및 포맷팅 규칙을 지정할 수 있습니다. 예를 들어, 상점이 "단위 가격" 사용법을 지정해서 소수점 이하 네 번째 자리까지 단위 가격을 반

올림할 수 있고, 다른 화폐 총계는 "기본" 사용법을 지정해서 소수점 이하 두 번째 자리까지 반올림할 수 있습니다. 숫자 사용법은 NUMBRUSG 테이블에서 정의되고, NUMBRUSGDS 테이블에서 설명됩니다.

항목 유형

항목 유형은 다른 종류의 기본 항목을 나타냅니다. WebSphere Commerce의 두 개의 기본 항목 유형은 동적 제품군 및 정상 항목입니다. 항목 유형은 ITEMTYPE 테이블에 사전 정의됩니다. 기본 항목에 대한 자세한 정보는 197 페이지의 제 21 장 『재고 자원』을 참조하십시오.

장치 형식

장치 형식은 DEVICEFMT 테이블에 저장되고 브라우저, I_MODE, 전자 우편, MQXML 및 MQNC와 같이 사이트가 사용하는 여러 장치 형식을 나타냅니다. 이런 장치 형식은 모두 다양한 매체를 통해 사용자가 사이트와 대화형 작업을 할 수 있도록 합니다.

주: 언어, 통화, 수량 단위 및 수량 단위 변환 규칙과 같은 일부 사이트 자원의 경우, 사이트 운영자가 적절한 테이블에 행을 추가해서 사이트 레벨 성능을 확장할 수 있습니다. 기타의 경우, 그것들이 나타내는 사이트 레벨 성능을 확장하려면 관련 사용자 정의가 필요할 수 있습니다. 예를 들어, 사이트 운영자가 사용자 정의된 통화 기호를 사용하여 소계를 표시할 새 번호 사용법을 추가한 경우, 표시할 소계 총계를 포맷할 때, 새 소계 번호 사용법을 지정하도록 소계를 표시하는 프로그램을 사용자 정의해야 합니다.

WebSphere Commerce에서 사이트 자원 작성

WebSphere Commerce 서버에 인스턴스를 작성할 때, 사이트 자원이 작성됩니다. WebSphere Commerce 서버에서 인스턴스 작성에 대한 자세한 정보는 *IBM WebSphere Commerce 설치 안내서*의 제 5 장 "인스턴스 작성 또는 수정"을 참조하십시오.

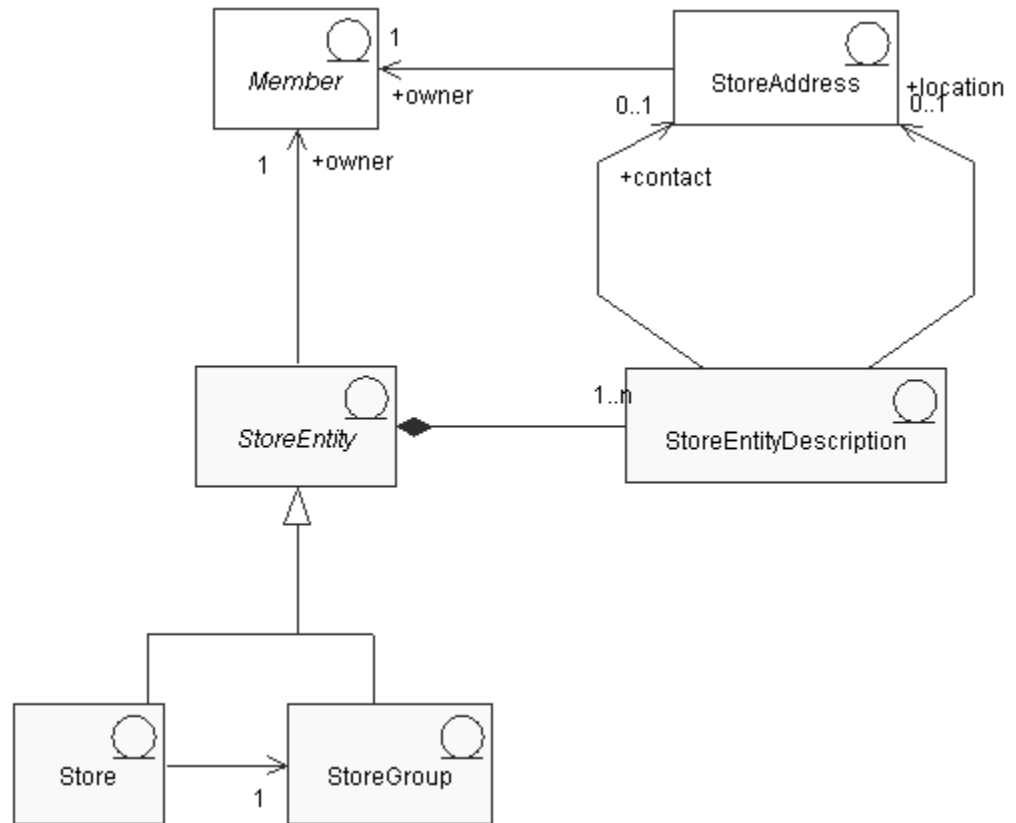
제 6 장 상점 자원

WebSphere Commerce에 상점을 작성하려면, 먼저 데이터베이스에 다음을 작성해야 합니다.

- 상점
- 상점이 속하는 그룹
- 상점 또는 상점 그룹을 이중으로 나타내는 추상 상점 엔티티 오브젝트

WebSphere Commerce의 상점 자원 이해

다음 도표는 WebSphere Commerce 서버의 상점 자원을 나타냅니다.



이 도표 및 상점 데이터 질의 기타 모두는 WebSphere Commerce 서버 정보 모델의 일부분입니다. 정보 모델에 대한 자세한 정보는 29 페이지의 『상점 데이터 정보 모델』을 참조하십시오. 이 도표에 사용된 규칙에 대한 자세한 정보는 359 페이지의 부록 A 『UML 범례』를 참조하십시오.

상점 엔티티

상점 엔티티는 상점 또는 상점 그룹을 나타내는 추상 최상위 클래스입니다.

상점 엔티티는 한 명의 소유자(구성원)를 가집니다. 구성원에 대한 자세한 정보는 207 페이지의 제 23 장 『고객 및 판매자 자원』을 참조하십시오.

상점 엔티티 설명

상점 엔티티 설명은 상점 엔티티를 설명합니다. 상점 엔티티는 설명을 포함할 수 있습니다. 상점이 다국어를 지원하는 경우, 상점 엔티티 설명이 다국어로 되어 있을 수 있습니다. 설명은 상점 엔티티에 대한 위치 주소뿐 아니라 연락 주소도 포함할 수 있습니다.

상점

상점은 상점 엔티티입니다. 상점은 상점 그룹에 속해야 합니다.

상점 그룹

상점 그룹은 상점의 컬렉션입니다. 상점 그룹은 상점 엔티티입니다. 상점 그룹은 상점 그룹 레벨에 저장되고, 상점 그룹의 모든 상점이 공유하는 공통 정보의 컨테이너로서 활동합니다. 예를 들어, 같은 상점 그룹의 상점은 세금 카테고리, 지원되는 언어, 지원되는 통화, 계산 코드 및 운송 관할 구역과 같은 정보를 공유할 수 있습니다.

현재, 하나의 상점 그룹만 WebSphere Commerce 서버 내의 사이트 관리 레벨에서 존재하고 유지보수될 수 있습니다.



WebSphere Commerce 서버의 상점 자원 구조에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말의 상점 오브젝트 및 데이터 모델을 참조하십시오.

WebSphere Commerce에서 상점 자원 작성

WebSphere Commerce의 상점 서비스 도구를 사용하면, 다음과 같은 상점 자원을 작성하거나 편집할 수 있습니다.

- 연락 자원의 상점 식별자 및 구성원 식별자
- STOREENT 테이블의 상점 식별자
- STORE 테이블의 상점 디렉토리
- STADDRESS 테이블의 주소 별명
- 상점 설명
- 상점 주소

주: 상점 서비스 도구는 상점 아카이브 양식의 사전 대량 자료 반입된 XML 파일을 사용하여 작업합니다.

결과적으로 상점 자원 작성에 대해 다음과 같이 두 가지 옵션이 있습니다.

- WebSphere Commerce와 함께 제공된 견본 상점 중 하나 또는 기존 상점 아카이브에서 기존 운송 자원 편집
- 상점 아카이브의 일부로서 공개되거나, 로더 패키지를 사용하여 로드되는 XML 파일 양식으로 상점 자원 작성

기존 상점 아카이브의 상점 자원 편집에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오. XML 파일 양식으로 상점 자원 작성에 대한 자세한 정보는 『XML 파일에 상점 데이터 자원 작성』을 참조하십시오.

XML 파일에 상점 데이터 자원 작성

로더 패키지를 사용하여 데이터베이스에 로드할 수 있는 XML 파일 양식으로 상점 자원을 작성하십시오. 다국어 지원 상점을 작성하는 경우에는 상점이 지원하는 로케일 각각에 대한 별도 XML 파일을 작성할 수 있습니다. 로케일 고유의 파일은 쉽게 번역할 수 있도록 모든 설명 정보를 지정해야 합니다. 로더 패키지에 대한 자세한 정보는 247 페이지의 제 7 부 『상점 공개』를 참조하십시오.

이들 태스크에 사용되는 견본 상점은 번역할 필요가 없는 모든 정보에 대해서는 하나의 store.xml 파일을 사용하고 번역할 필요가 있는 정보에 대해서는 상점이 지원하는 로케일 각각에 대해 다른 store.xml 파일을 사용합니다. 로케일 고유의 파일은 모든 설명 정보를 포함합니다.

상점 자원을 작성하려면 다음을 수행하십시오.

1. 견본 상점의 상점 자원을 작성하는 데 사용된 XML 파일을 검토하십시오. 견본 상점의 모든 파일은 해당 상점 아카이브 파일에 있습니다.

상점 아카이브 파일은 다음 디렉토리에 있습니다.







- **NT** drive:\WebSphere\CommerceServer\samplstores
- **2000** drive:\Program Files\WebSphere\CommerceServer\samplstores
- **AIX** /usr/WebSphere/CommerceServer/samplstores
- **Solaris** /opt/WebSphere/CommerceServer/samplstores
- **Linux** /opt/WebSphere/CommerceServer/samplstores
- **400** /qibm/proddata/WebCommerce/samplstores

주: WebSphere Commerce 온라인 도움말에는 견본 상점에 포함된 데이터 자원 각각에 대한 정보가 포함됩니다.

개별 견본 상점에는 상점 정보를 포함하는 두 개의 store.xml 파일이 들어 있습니다. 상점 아카이브의 store.xml 파일을 보려면 ZIP 프로그램을 사용하여 상점

아카이브의 압축을 푸십시오. store.xml 파일은 데이터 디렉토리에 있습니다. 언어별 store.xml은 데이터 디렉토리의 로케일 특정 서브디렉토리에 있습니다.

2. 361 페이지의 부록 B 『데이터 작성』 정보를 검토하십시오.
3. 견본 상점 아카이브의 store.xml 파일 중 하나를 복사하거나 새 파일을 작성해서 store.xml 파일을 작성하십시오. 자세한 정보는 store.xml에 해당 DTD 파일을 참조하십시오. DTD 파일은 다음 디렉토리에 있습니다.

-  NT drive:\WebSphere\CommerceServer\xml\sar
-  2000 drive:\Program Files\WebSphere\CommerceServer\xml\sar
-  AIX /usr/WebSphere/CommerceServer/xml/sar
-  Solaris /opt/WebSphere/CommerceServer/xml/sar
-  Linux /opt/WebSphere/CommerceServer/xml/sar
-  400 /qibm/proddata/WebCommerce/xml/sar

4. 상점 엔티티를 작성하십시오.

- a. 다음 예를 참조하여 XML 파일의 STOREENT 테이블에 상점 엔티티를 정의하십시오.

```
<storeent
  storeent_id="@storeent_id_1"
  member_id("&MEMBER_ID"
  type="S"
  identifier="ToolTech"
  setcurr="USD"
/>
```

여기서

- storeent_id는 생성된 고유 키입니다.
 - member_id는 상점 엔티티의 소유자입니다.
 - type은 상점 엔티티의 종류로 G는 상점 그룹, S는 상점입니다.
 - identifier는 소유자와 함께 상점 엔티티를 고유하게 식별하는 문자열입니다.
 - setcurr는 상점 엔티티에 대한 기본 통화입니다. 즉, 선호 통화를 가지지 않은 고객이 사용할 통화입니다. 이것이 상점에 대해 널(NULL)값이면, 상점 그룹에서 기본 통화를 얻습니다.
5. 상점 주소를 작성하십시오.
 - a. 다음 예를 참조하여 XML 파일의 STADDRESS 테이블에 상점 주소를 작성하십시오. 다국어 지원 카탈로그를 작성하는 경우에는 이 정보를 로케일 고유의 XML 파일에 포함시켜야 합니다.

```

<staddress
staddress_id="@staddress_id_en_US_1"
  member_id="&MEMBER_ID"
nickname="storeaddress_English"
address1="12xx Martindale Avenue"
address2="Suite 9xx"
businesstitle="ToolTech"
city="Toolsville"
state="Ontario"
zipcode="Lxx lxx"
country="Canada"
phone1="1-800-555-1234"
fax1="1-800-555-4321"
email1="info@tooltech.xxx"
/>

```

여기서

- staddress_id는 생성된 고유 키입니다.
- member_id는 상점 엔티티의 소유자입니다.

6. 상점 엔티티에 대한 설명을 작성하십시오.

- a. 다음 예를 참조하여 XML 파일의 STOREENT 테이블에 상점 엔티티에 대한 설명을 작성하십시오. 다국어 지원 카탈로그를 작성하는 경우에는 이 정보를 로케일 고유의 XML 파일에 포함시켜야 합니다.

```

<storeentds
description="Commerce Models Store entity"
language_id="&en_US"
displayname="ToolTech"
storeent_id="@storeent_id_1"
staddress_id_cont="@staddress_id_en_US_1"
staddress_id_loc="@staddress_id_en_US_1"

```

여기서

- description은 고객에게 표시하기 적절한, 상점 엔티티에 대한 보다 자세한 설명입니다.
- language_id는 상점에서 구매하는 고객에게 표시되는 정보의 기본 언어입니다.
- displayname은 고객에게 표시하기 적절한 상점 엔티티에 대한 간략한 설명입니다.
- storeent_id는 상점 엔티티입니다.
- staddress_id_cont는 상점 엔티티의 연락 주소입니다.
- staddress_id_loc는 상점 엔티티의 실제 주소입니다.

7. 데이터베이스에 상점을 작성하십시오.

- a. 다음 예를 참조하여, XML 파일의 STORE 테이블에 상점을 정의하십시오.

```

<store
store_id="@storeent_id_1"
directory="ToolTech"
ffmcenter_id="@ffmcenter_id_1"
language_id="&en_US"
storegrp_id="-1"
allocationgoodfor="43200"
bopmpadfactor="0"
defaultboffset="2592000"
ffmcselectionflags="0"
maxboffset="7776000"
rejectedordexpiry="259200"
rtnffmctr_id="@ffmcenter_id_1"
pricerefflags="0"
storetype="B2B"
/>

```

여기서

- store_id는 생성된 고유 키입니다.
- directory는 상점 지정 웹 자원을 찾을 수 있는 디렉토리입니다. 파일 시스템에서 이 자원의 실제 위치는 StoresDocRoot, StoresWebPath 및 StoresPropertiesPath와 같은 WebSphere Commerce 구성 파일의 여러 구성 매개변수를 더하여 이 열의 값을 기반으로 합니다. 예를 들어, StoresDocRoot가 D:\WebSphere\wcs\stores, StoresWebPath가 web, StorePropertiesPath가 properties, 이 열의 값이 mystore인 경우에는 JSP 파일은 D:\WebSphere\wcs\stores\web\mystore 디렉토리에 있고 특성 파일은 D:\WebSphere\wcs\stores\properties\mystore에 있습니다.
- ffmcenter_id는 상점의 기본 서비스 센터입니다.
- language_id는 상점에서 구매하는 고객에게 표시되는 정보의 기본 언어입니다.
- storegrp_id는 이 상점이 연관되는 상점 그룹입니다. 이 번호는 STOREGRP 테이블에 생성됩니다.
- allocationgoodfor는 할당 후, 시간(초)이 많이 경과했을 때 ReleaseExpiredAllocations 스케줄러 작업을 사용하여 ATP 할당을 할 수 있음을 의미합니다.
- bopmpadfactor는 이 상점이 주문 총계(세금이나 운송 경비와 같은)를 서비스 센터에 따라 다르게 계산하는 경우 서비스 센터를 최종적으로 백오더된 항목에 할당할 때 이전에 제출된 주문의 주문 총계를 변경할 수 있음을 의미합니다. 이 채우기 요인은 백분율을 나타내는데, 필요한 경우 지불 관리자에게 제시된 주문 총계를 증가시킬 수 있습니다. 예를 들어, 5를 지정하여 5퍼센트까지 증가시킬 수 있습니다.
- defaultboffset는 백오더된 OrderItem에 대해 예상 유효 시간을 결정할 수 없을 때, 미래의 시간(초)을 설정할 수 있습니다.

- maxboffset는 백오더된 OrderItem에 대해 예상 유효 시간이 많이 초과한 경우, 미래의 시간(초)을 설정할 수 있습니다.
- rejectedordexpiry는 초수 보다 긴 거부 상태의 지불이 있는 주문이며 취소에 대한 후보입니다.
- rtnffmctr_id는 상점으로 리턴되는 판매 계획에 대한 기본 서비스 센터입니다.
- pricerefflags는 GetContractUnitPrices 태스크 명령의 기본 구현으로 가격을 최신 정보로 고쳤을 때, 검색되는 TradingAgreements 및 판매 설정을 통제하는 비트 플래그를 포함합니다.
 - 1 = usePreviousOnly - OrderItems에 의해 참조된 것 사용. 더 이상 사용할 수 없으면 실패합니다.
 - 2 = usePreviousOrSearchAgain - usePreviousOnly와 같지만, 더 이상 사용할 수 없을 때 실패하는 대신에 ORDIOFFER 및 ORDITRD 테이블에 저장된 것을 검색합니다.
 - 4 = alwaysSearchAgain - 항상 ORDIOFFER 및 ORDITRD 테이블에 저장된 것을 검색합니다.
- storetype은 상점 유형에 따라 적절한 기능을 제공하는 사용자 인터페이스에 의해 사용하기 위해 B2B = Business-to-Business, B2C = Business-to-Consumer와 같은 상점 유형 중 하나를 나타냅니다.

8. 상점에 지원되는 언어를 정의하십시오.

- a. 다음 예를 참조하여, STORELANG 테이블에 정보를 추가할 XML 파일에 상점에서 지원되는 언어를 정의하십시오. 상점이 다국어를 지원하는 경우에는 로케일 고유의 XML 파일(상점이 지원하는 개별 언어에 대해 하나씩)에 이 정보를 포함시켜야 합니다.

```
<storelang
language_id="&en_US"
storeent_id="@storeent_id_1"
/>
```

여기서

- language_id는 상점 엔티티가 지원하는 언어입니다.
 - storeent_id는 상점 엔티티입니다.
- b. 다음 예를 참조하여, STORELANGDS 테이블에 언어에 대한 정보를 추가하십시오. 상점이 다국어를 지원하는 경우에는 로케일 고유의 XML 파일(상점이 지원하는 개별 언어에 대해 하나씩)에 이 정보를 포함시켜야 합니다.

```
<storlangds
  description="United States"
  language_id="&en_US"
  storeent_id="@storeent_id_1"
  language_id_desc="& en_US"
/>
```

여기서

- description은 선택 목록에서 고객에게 표시하기 적절한, 언어에 대한 간략한 설명입니다.
- language_id는 설명하는 언어입니다.
- storeent_id는 언어를 지원하는 상점 엔티티입니다.
- language_id_desc는 설명되고 있는 언어입니다.



@ 및 & 사용에 대한 자세한 정보는 361 페이지의 부록 B 『데이터 작성』을 참조하십시오.

제 7 장 명령, 보기 및 URL 레지스트리 데이터

WebSphere Commerce 서버 인스턴스를 작성할 때, 기본 명령, 보기 및 WebSphere Commerce가 제공된 URL이 CMDREG, VIEWREG, URLREG 등 해당 테이블의 WebSphere Commerce 서버 데이터베이스에 등록됩니다. 이러한 명령, 보기 및 URL은 인스턴스에 상주하는 모든 상점에서 사용 가능합니다.

WebSphere Commerce는 또 기본 JSP 파일을 제공하여 기본 보기를 표시합니다. 이러한 JSP 파일들은 VIEWREG 테이블의 보기와 연관됩니다.

새 명령, 보기 또는 URL을 작성하거나 기존의 명령, 보기 또는 URL을 사용자 정의할 경우, 상점에서 사용 가능하기 전에 CMDREG, VIEWREG, URLREG 등 해당 데이터베이스 테이블에 등록해야 합니다. 상점에서 사용할 새 JSP 파일을 작성할 경우, VIEWREG 테이블의 해당 보기와 이것들을 연관시켜야 합니다.

주: 새 JSP 파일을 작성하지만 보기와 연관된 기본 JSP 파일과 동일한 이름을 새 JSP 파일에 명명하고자 할 때는, VIEWREG 테이블에 새 JSP 파일을 등록할 필요가 없습니다.

명령, 보기 또는 URL 작성 또는 사용자 정의에 대한 자세한 정보는 *IBM WebSphere Commerce 프로그래머 안내서*를 참조하십시오. *프로그래머 안내서*에도 명령, 보기, URL 및 JSP 파일 등록 방법 및 시간에 관한 정보가 포함됩니다.



WebSphere Commerce 서버의 명령 및 보기 자원의 구조에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말의 명령 및 보기 데이터 모델을 참조하십시오.

WebSphere Commerce에서 명령, 보기 및 URL 등록

상점에 여러 개의 새 명령, 보기, URL 또는 JSP 파일들을 작성 또는 사용자 정의할 경우, 로더 패키지를 사용하여 데이터베이스로 로드할 수 있는 XML 파일을 사용하여, 또는 상점 서비스를 사용하여 공개할 수 있는 상점 이카이브의 일부로서 이것들을 등록하고자 할 수도 있습니다. 로더 패키지에 대한 자세한 정보는 247 페이지의 제 7 부 『상점 공개』를 참조하십시오.

주: XML 파일을 작성하여 새 명령 또는 사용자 정의 명령을 로드하기 전에, 작업 명령 방법에 대한 자세한 정보는 *프로그래머 안내서*를 참조하십시오.

XML 파일을 작성하여 명령, 보기 및 URL 등록

XML 파일을 작성하여 상점에 새 명령, 보기 및 JSP 파일을 등록하려면, 다음을 수행하십시오.

1. 견본 상점의 명령, 보기, JSP 파일들을 등록하는 데 사용된 XML 파일을 검토하십시오. 각 견본 상점은 특성이 들어 있는 `command.xml` 파일을 갖고 있습니다. 상점 아카이브 파일들은 다음 디렉토리에 있습니다.

- **NT** drive:\WebSphere\CommerceServer\samplstores
- **2000** drive:\Program Files\WebSphere\CommerceServer\samplstores
- **AIX** /usr/WebSphere/CommerceServer/samplstores
- **Solaris** /opt/WebSphere/CommerceServer/samplstores
- **Linux** /opt/WebSphere/CommerceServer/samplstores
- **400** /qibm/proddata/WebCommerce/samplstores

주: WebSphere Commerce 온라인 도움말은 견본 상점에 포함된 각 데이터 자원에 대한 정보를 포함합니다.

상점 아카이브의 콘텐츠를 보려면, 압축 풀기 프로그램을 사용하십시오. `command.xml` 파일은 `data` 디렉토리에 있습니다.

2. 361 페이지의 부록 B 『데이터 작성』 정보를 검토하십시오.
3. 견본 상점 아카이브에서 `command.xml` 파일들 중 하나를 복사하거나 새 파일을 작성하여 `command.xml` 파일을 작성하십시오. 자세한 정보는 `command.xml`에 해당 DTD 파일을 참조하십시오. DTD 파일은 다음 디렉토리에 있습니다.

- **NT** drive:\WebSphere\CommerceServer\xml\sar
- **2000** drive:\Program Files\WebSphere\CommerceServer\xml\sar
- **AIX** /usr/WebSphere/CommerceServer/xml/sar
- **Solaris** /opt/WebSphere/CommerceServer/xml/sar
- **Linux** /opt/WebSphere/CommerceServer/xml/sar
- **400** /qibm/proddata/WebCommerce/xml/sar

4. 제어기 명령은 URLREG 테이블 및 CMDREG 테이블에 등록되어야 합니다. URLREG 테이블에 새 제어기 명령 또는 사용자 정의 제어기 명령을 등록하려면, 다음 예를 참조하여 새 사용자 정의 제어기 명령 각각에 대한 XML 파일에서 하나의 항목을 작성하십시오.

```
<urlreg
url="MyProductDisplay"
storeent_id="@storeent_id_1"
```



```

interfacename="com.mystore.commerce.catalog.commands.
ProductDisplayCmd"
https="0"
description="Product display command for my store"
authenticated="0"
internal="0" />

```

여기서

- urlreg는 이 정보가 반입되어 있는 데이터베이스 테이블(URLREG)의 이름입니다.
 - url은 URI 이름입니다.
 - storeent_id는 상점 엔티티 식별자이며, @ 기호 사용은 내부 별명 분석이라고도 합니다. 내부 별명 분석을 사용할 때, 별명은 XML 문서의 1차 키(식별자) 대신 대체됩니다. 이 별명을 XML 파일 어디에서나 해당 요소를 참조하는 데 사용할 수 있습니다. 그렇게 되면 XML 파일을 빌드하는 데 필요한 고유 색인을 알 필요가 없습니다. 공개하는 동안, ID 분석기가 @ 기호를 고유한 값으로 바꿉니다. 자세한 정보는 361 페이지의 부록 B 『데이터 작성』을 참조하십시오.
 - interfacename은 제어기 명령 인터페이스 이름입니다.
 - https는 이 URL 요청에 필요한 보안 HTTP입니다. 보안 HTTP가 필수인 경우에는 1을, 그렇지 않으면 0을 사용하십시오.
 - authenticated는 URL 요청에 대해 사용자 로그인이 필요한지 여부를 나타냅니다. 인증이 필수인 경우에는 1을, 그렇지 않으면 0을 사용하십시오.
 - internal은 명령이 WebSphere Commerce의 내부인지 여부를 나타냅니다. 내부 URL은 WebSphere Commerce 도구에서 사용됩니다. 보기가 내부이면 1을 사용하고, 명령이 외부이면 0을 사용하십시오. 작성하는 URL은 외부이어야 합니다.
5. CMDREG 테이블에 새 제어기 명령 또는 새 태스크 명령을 등록하려면, 참조로 ToolTech 견본 상점 command.xml 파일에서 태스크 명령의 새 예를 사용하여 새 제어기 또는 태스크 명령이나 사용자 정의 제어기 또는 태스크 명령 각각에 대한 XML 파일에서 하나의 항목을 작성하십시오.

```

< cmdreg
storeent_id="@storeent_id_1"
interfacename="com.ibm.commerce.payment.commands.DoPaymentCmd"
classname="com.ibm.commerce.payment.commands.DoPaymentMPFCmdImpl" />

```

여기서

- cmdreg는 이 정보가 반입되어 있는 데이터베이스 테이블(CMDREG)의 이름입니다.

- storeent_id는 상점 엔티티 식별자이며, @ 기호 사용은 내부 별명 분석이라고도 합니다. 내부 별명 분석을 사용할 때, 별명은 XML 문서의 1차 키(식별자) 대신 대체됩니다. 이 별명을 XML 파일 어디에서나 해당 요소를 참조하는 데 사용할 수 있습니다. 그렇게 되면 XML 파일을 빌드하는 데 필요한 고유 색인을 알 필요가 없습니다. 공개하는 동안, ID 분석기가 @ 기호를 고유한 값으로 바꿉니다. 자세한 정보는 361 페이지의 부록 B 『데이터 작성』을 참조하십시오.
 - interfacename은 명령 인터페이스 이름입니다.
 - classname은 명령 구현 클래스 이름입니다. 일반적으로 이 이름은 끝에 Impl이 추가된 인터페이스 이름입니다.
6. 새 보기를 등록하거나 새 JSP 파일을 보기와 연관시키려면, ToolTech 견본 상점 command.xml 파일에서 다음 예를 사용하여 VIEWREG 테이블에서 하나의 항목을 작성하십시오.

```
< viewreg
viewname="OrderOptionsView"
devicefmt_id="-1"
storeent_id="@storeent_id_1"
interfacename="com.ibm.commerce.command.ForwardViewCommand"
classname="com.ibm.commerce.command.HttpForwardViewCommandImpl"
properties="docname=Shipping.jsp"
internal="0"
https="0" />
```

여기서

- viewreg는 이 정보가 반입될 데이터베이스 테이블(VIEWREG) 이름입니다.
- viewname은 보기 이름입니다.
- devicefmt_id는 이 보기가 사용될 장치 유형으로, 예를 들어 브라우저입니다.
- storeent_id는 상점 엔티티 식별자이며, @ 기호 사용은 내부 별명 분석이라고도 합니다. 내부 별명 분석을 사용할 때, 별명은 XML 문서의 1차 키(식별자) 대신 대체됩니다. 이 별명을 XML 파일 어디에서나 해당 요소를 참조하는 데 사용할 수 있습니다. 그렇게 되면 XML 파일을 빌드하는 데 필요한 고유 색인을 알 필요가 없습니다. 공개하는 동안, ID 분석기가 @ 기호를 고유한 값으로 바꿉니다. 자세한 정보는 361 페이지의 부록 B 『데이터 작성』을 참조하십시오.
- interfacename은 보기 명령 인터페이스 이름입니다. 기본 옵션은 ForwardView, DirectView 및 RedirectView입니다.
- classname은 보기 구현 클래스 이름입니다. 일반적으로 이 이름은 끝에 Impl이 추가된 인터페이스 이름입니다.

- `properties`는 명령에 입력 특성으로 설정된 기본 이름-값 쌍입니다. 동일한 페이지가 항상 표시되면 이 특성에 JSP 파일 이름을 설정하십시오(예: `docname=Shipping.jsp`).
- `internal`은 보기가 WebSphere Commerce의 내부인지 여부를 나타냅니다. 내부 보기는 WebSphere Commerce 도구에 의해 사용됩니다. 보기가 내부이면 1을 사용하고, 명령이 외부이면 0을 사용하십시오. 작성하는 보기는 외부이어야 합니다.
- `https`는 이 URL 요청에 필요한 보안 HTTP입니다. 보안 HTTP가 필수인 경우에는 1을, 그렇지 않으면 0을 사용하십시오.



@ 및 &의 사용에 대한 자세한 정보는 361 페이지의 부록 B 『데이터 작성』을 참조하십시오.

제 8 장 카탈로그 자원

전통적인 카탈로그와 마찬가지로 사용자의 온라인 카탈로그는 사용자가 판매를 위해 제공하는 상품 및 서비스로 구성되어 있습니다. 온라인 카탈로그의 크기 및 구조가 상점마다 다르다 해도, 구매시 사용 가능한 판매 계획의 유형과 총계에 따라 카탈로그는 다음 사항을 요구합니다.

- 판매하는 상품은 다음을 포함합니다.
 - 거의 항상 온라인 카탈로그에 포함되어 있는 가격
 - 사용자의 판매 계획에 대한 설명 및 이미지와 같은 상품 데이터
 - 대부분은 카테고리가 있으나 고객의 탐색을 용이하게 하기 위해 모든 카테고리가 판매 계획을 카테고리로 분리하지는 않습니다.
- 판매 상품의 표시 방법: 카탈로그 표시 페이지는 페이지가 고객에게 표시되는 방식을 보여주고, 여러 카탈로그 페이지 간에 일관된 특엔필을 제공합니다. 카탈로그의 구조를 정하는 방법은 판매 계획에 따라 다릅니다.

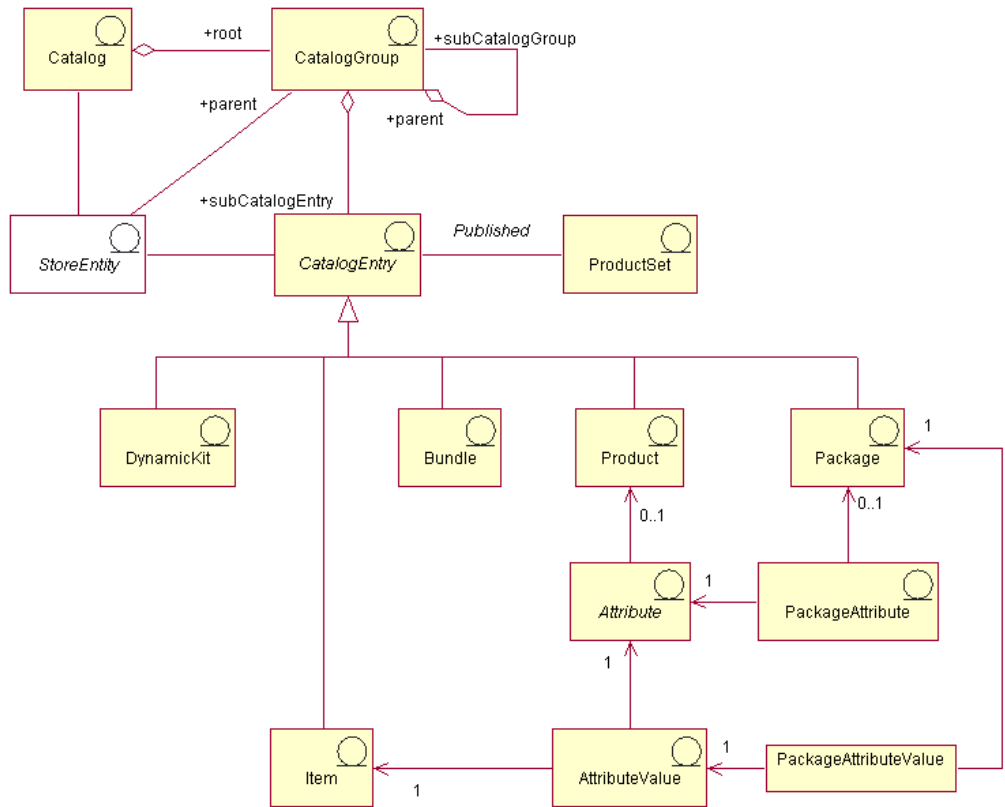
WebSphere Commerce의 카탈로그 이해

WebSphere Commerce는 상점의 온라인 카탈로그에 여러 가지 요구사항을 기입해 놓았습니다. WebSphere Commerce 시스템의 모든 상점은 마스터 카탈로그를 갖고 있어야 합니다. 마스터 카탈로그는 상점 판매 계획 관리의 기본 위치입니다. 이는 모든 상품, 항목, 관계 그리고 상점에서 판매하는 모든 상품에 대한 가격을 포함하고 있는 단일 카탈로그입니다.

상점 간에 마스터 카탈로그를 공유할 수 있고 필요한 수량만큼의 상점을 정의할 수 있습니다. 카탈로그 관리를 위해 마스터 카탈로그를 작성하는 것 이외에도 표시 목적을 위해 하나 이상의 탐색 카탈로그를 작성할 수도 있습니다. 탐색 카탈로그에는 마스터 카탈로그와 동일한 항목들이 포함되지만 고객 표시 목적을 위한 유연한 구조를 가질 것입니다. 원하는 수량의 탐색 카탈로그를 가질 수 있습니다. 그러나 온라인 판매 계획을 관리하기 위해 사용되는 것이 마스터 카탈로그이기 때문에 마스터 카탈로그를 탐색 카탈로그로 사용하여 유지보수 오버헤드를 최소화할 것을 권장합니다.

WebSphere Commerce 상점에서 사용할 목적으로 새 마스터 카탈로그를 작성하거나, ToolTech와 같이 WebSphere Commerce 견본 상점에서 사용 가능한 기존의 마스터 카탈로그를 사용할 경우, 카탈로그를 수정하여 이러한 요구사항들을 충족시켜야 합니다.

다음 도표는 WebSphere Commerce의 카탈로그 기본 구조를 설명합니다.



이 도표와 상점 데이터 질의 모든 기타 도표는 WebSphere Commerce 정보 모델의 일부분입니다. 정보 모델에 대한 자세한 정보는 29 페이지의 『상점 데이터 정보 모델』을 참조하십시오. 이 도표에서 사용된 규칙에 대한 자세한 정보는 359 페이지의 부록 A 『UML 범례』를 참조하십시오.

카탈로그

카탈로그는 정보 모델에 대한 시작 지점입니다. 카탈로그는 온라인 카탈로그에 대한 모든 계층 정보 및 탐색 정보를 포함하고 있으며, 온라인 상점에서 표시 및 구매를 위해 사용 가능한 카탈로그 그룹 및 카탈로그 항목의 컬렉션입니다.

WebSphere Commerce에서는, 카탈로그가 카탈로그 엔티티에 의해 데이터베이스에 표시됩니다. 카탈로그 엔티티는 독특한 카탈로그 ID 및 카탈로그 설명(예: 카탈로그 이름)으로 구성됩니다. 개별 카탈로그가 독립적이고 독특한 엔티티이기 때문에 하나 이상의 상점과 쉽게 연관될 수 있습니다. WebSphere Commerce 시스템의 모든 상점은 적어도 하나의 카탈로그와 연관되어 있어야 합니다. 마스터 카탈로그는 온라인 상점에서 판매하는 모든 항목을 포함하고 있는 특별한 카탈로그입니다.

카탈로그 그룹

카탈로그 그룹은 탐색 및 카탈로그 파티션을 목적으로 작성된 카탈로그 항목들의 일반 그룹입니다. 카탈로그 그룹은 하나의 카탈로그에 속하며, 두 개 이상의 카탈로그 그룹

이나 카탈로그 항목을 포함합니다. 카탈로그 그룹을 두 개 이상의 카탈로그에 연관시킬 수 있습니다. 카탈로그 그룹은 카테고리로도 알려져 있습니다.

플랫 카탈로그는 포함하는 상품을 카테고리로 그룹화하지 않습니다. 대신, 상품의 목록을 표시합니다. WebSphere Commerce에서 플랫 카탈로그를 작성하는 것이 가능하지만, 구조 및 탐색 목적의 카탈로그 그룹 작성을 권장합니다.

카탈로그 그룹을 작성할 때, 먼저 카탈로그를 계층 또는 역 트리 포맷으로 배열하십시오. 트리는 일반 카탈로그 그룹(루트 카테고리라고 함)에서 시작하여, 더 이상 분할될 수 없을 때까지 특정 서브카테고리로 점점 분기됩니다. 상품만 들어 있는 가장 낮은 레벨의 각 카탈로그 그룹을 최하위 그룹이라 합니다. 카탈로그 그룹은 바로 아래 있는 카테고리의 상위 항목이며 바로 위에 있는 카테고리의 하위 항목입니다. 예를 들어, 남성 의류는 남성 의류 카테고리 그룹이며, 카탈로그 그룹 바지 및 셔츠는 상품 그룹입니다.

카탈로그 항목

개별 카탈로그 그룹은 카탈로그 항목을 포함합니다. 카탈로그 항목은 온라인 카탈로그에서 주문할 수 있는 상품을 나타냅니다. 이 항목들은 일반적으로 이름이나 부품 번호, 설명, 하나 이상의 가격, 이미지 및 기타 정보를 갖습니다. 카탈로그 항목은 상품, 항목, 패키지, 번들 또는 동적 제품군이 될 수 있습니다. 필요한 경우, 다섯 개의 기존 모델 중 어느 것에도 맞지 않은 새 카탈로그 항목 유형을 작성할 수 있습니다. 개별 유형의 카탈로그 항목에 대한 자세한 정보는 아래에서 사용 가능합니다.

상품

상품은 카탈로그 항목의 한 유형입니다. 상품은 동일한 속성을 보여주는 항목 또는 SKU의 그룹에 대한 템플릿로서 행동합니다. 예를 들어, 셔츠는 카탈로그에 있는 상품입니다. 셔츠에 속성 및 속성값을 추가하면 개별 변형은 검은색 작은 셔츠와 같이 하나의 항목이 됩니다.

항목

항목은 특정 이름, 부품 번호 및 가격을 갖고 있는 판매 계획의 실재하는 단위입니다. 예를 들어, 검은색 작은 셔츠는 셔츠가 상품인 동안에는 하나의 항목입니다. 특정 상품과 연관된 모든 항목들은 동일한 속성 세트를 보이며, 속성값에 의해 구별됩니다.

주: WebSphere Commerce 액셀러레이터 사용자의 경우, 항목 및 SKU가 동의어로 여겨집니다. WebSphere Commerce 액셀러레이터에서 Product Management 도구를 사용할 때, 주문 가능 항목을 SKU라 합니다. WebSphere Commerce 데이터베이스 스키마에서, 이 특정 유형의 카탈로그 항목은 항목이라 합니다.

패키지

패키지는 카탈로그 항목의 자동형 컬렉션입니다. 예를 들어, 컴퓨터 패키지에는 개별적으로 판매할 수 없는 특정 CPU, 모니터, 하드 드라이브가 포함될 수 있습니다. 상품과

비슷하게 패키지는 속성을 정의하며, 완성된 패키지를 위한 컨테이너입니다. 완성된 패키지는 SKU에 비유할 수 있습니다. 패키지는 자체 가격을 가지며, 장비구니에 추가할 수 있는 실질적인 주문 가능 SKU입니다. 탐색하는 동안 또는 패키지가 장비구니에 제출된 후에는 패키지를 분해하거나 수정할 수 없습니다.

번들

번들은 고객이 한 번에 여러 개의 항목을 구입할 수 있게 해주는 카탈로그 항목의 콜렉션입니다. 예를 들어, 컴퓨터용 번들은 CPU, 모니터, 하드 드라이브 및 CD-ROM 드라이브로 구성될 수 있습니다. 항목 그룹, 상품 조합, 항목과 완성된 패키지 등이 번들이 될 수 있습니다. 일정한 항목만을 포함하는 번들을 선택하면, 번들은 장비구니에 개별적으로 추가되는 개별 주문 가능 SKU로 분해될 수 있습니다. 그러나, 상품을 포함하는 번들을 선택하면, 이 상품들을 장비구니에 추가하기 전에 SKU 분석을 통해 이 상품들을 항목으로 완성해야 합니다. 각각의 경우에, 일단 번들을 분해하여 자체 구성요소 항목들을 장비구니에 추가하면, 각 항목을 수정하거나 제거할 수 있습니다.

동적 제품군

동적 제품군은 고객이 동적으로 구성할 수 있는 카탈로그 항목 유형입니다. 이 상품 구성(또는 그룹)은 고객 요구사항을 따르며 단일 단위로 판매됩니다. 동적 제품군의 구성요소는 미리 정의된 규칙 및 사용자 상호작용 세트를 통해 외부 상품 구성자에 의해 구성됩니다. 주문에 동적 제품군을 추가하는 것은 패키지를 추가하는 것과 유사합니다. 패키지와 같이, 동적 제품군의 구성요소는 개별 수정할 수 없으며 전체 구성은 전체적으로 처리되어야 합니다. 그러나 외부 상품 구성자를 사용하여 동적 제품군을 재구성해서 동적 제품군 구성요소를 변경할 수 있습니다.

상품 세트

상품 세트를 공개된 카탈로그 항목과 연관시킬 수 있습니다. 상품 세트는 카탈로그를 논리적 서브세트로 분할하는 메커니즘을 제공합니다. 이러한 분할을 통해 카탈로그의 서로 다른 부분을 서로 다른 사용자에게 보여줄 수 있습니다. 장기 구매 계약을 작성하고, 장기 구매 계약의 참여자만이 사전 정의된 상품 세트에 속하는 상품을 구매할 자격이 있음을 지정할 수 있습니다. WebSphere Commerce는 마스터 카탈로그로부터 상품 세트를 작성하고 이들을 장기 구매 계약에서 자격 필터링에 사용할 수 있도록 하는 도구를 제공합니다.

속성

속성은 색상 또는 크기와 같은 온라인 상점의 상품 특성입니다. 속성은 상품에 속합니다. 속성 및 속성값의 가능한 각각의 조합이 항목을 정의합니다.

속성값

속성값은 특정 색상(파란색 또는 노란색) 또는 크기(대, 중, 소)와 같은 속성의 특성입니다. 이런 특성을 항목에 지정하기 전에 속성값을 미리 정의해야 합니다. 속성 및 속성값의 가능한 각각의 조합이 항목을 정의합니다.

패키지 속성

패키지 속성은 패키지 내에 포함되어 있는 상품의 속성에서 상속됩니다. 패키지는 패키지 속성을 갖지 않거나 여러 개의 패키지 속성을 갖습니다. 패키지 속성은 하나의 속성을 나타냅니다.

패키지 속성값

패키지 속성값은 패키지 속성에 지정된 값입니다. 패키지 속성값은 패키지 내에 포함되어 있는 상품의 속성값에서 상속됩니다. 패키지 속성값은 하나의 속성값을 나타냅니다.



WebSphere Commerce의 카탈로그 자원 구조에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오.

WebSphere Commerce에서 카탈로그 자원 작성

상점에 대한 카탈로그 자원을 작성하려면, 여러 개의 WebSphere Commerce 데이터베이스 테이블에 정보를 추가해서 마스터 카탈로그를 작성해야 합니다. 로더 패키지로 데이터베이스에 로드되어 있는 XML 파일을 사용하여 카탈로그를 작성할 수 있습니다. 다국어 지원 카탈로그를 작성하는 경우에는 상점이 지원하는 로케일 각각에 대한 XML 파일을 분리해야 합니다. 각 로케일에 고유한 XML 파일은 카탈로그, 카탈로그 그룹 및 카탈로그 항목에 대한 설명과 같이 번역 가능한 정보를 추가합니다.

다음은 카탈로그 처리의 개요입니다.

1. WebSphere Commerce에서 카탈로그는 XML 파일을 사용하여 작성됩니다. 카탈로그는 페이퍼 카탈로그와 동일한 사용자 데이터베이스, 카탈로그 엔티티로 시작합니다.
2. 카탈로그 그룹을 추가하여 카탈로그 구조 및 탐색을 작성하여 사용자의 판매 계획에 대한 카테고리 및 레이아웃을 판별합니다.
3. 카탈로그 항목에 대한 기본으로서 재고 정보를 작성합니다.
4. 제품, SKU, 패키지, 번들 및 동적 제품군을 표시하는 카탈로그 항목의 양식에 사용자의 판매 계획을 추가합니다.
5. 속성 및 속성값이 사용자의 카탈로그 제품에 추가되어 다른 것으로부터의 다른 SKU를 구별합니다.
6. 승급 목적을 위해 특정 카탈로그 항목들을 함께 그룹화하려면 패키지와 번들을 작성할 수 있습니다.

7. 카탈로그 그룹과 카탈로그 항목 간의 관계가 다음에 작성됩니다. 이것은 카탈로그 그룹에 속한 항목을 판별합니다.
8. 제품 추천 상품 전략으로서 사용자의 카탈로그 항목에 대한 판매 계획 연관을 작성할 수 있습니다.
9. 카탈로그, 카탈로그 그룹 및 카탈로그 항목을 사용자의 WebSphere Commerce 상점과 연관시킵니다.
10. 마지막 단계에서 다음을 작성해야 합니다.
 - a. 사용자의 판매 계획에 대한 세금
 - b. 운송 방법
 - c. 재고 창고와 운송 및 수령 센터로서 활동할 수 있는 서비스 센터. 하나의 상점은 하나 이상의 정의된 서비스 센터를 가질 수 있습니다.
 - d. 사용자의 판매 계획에 대한 가격

마스터 카탈로그

여러 레벨 카테고리를 포함하는 마스터 카탈로그를 작성하려면 다음과 같이 하십시오.

부분 1: 카탈로그 준비

1. WebSphere Commerce 내에 있는 카탈로그 정보와 그에 해당하는 오브젝트 및 데이터 모델을 검토하십시오. 카탈로그 정보는 온라인 카탈로그 탐색, 파티션, 카테고리화 그리고 주문 가능 판매 계획을 위한 연관을 제공하는 WebSphere Commerce 서버의 구성요소입니다.
2. WebSphere Commerce 로더 패키지 정보를 검토하십시오. 로더 패키지는 1차적으로 데이터를 준비하고 데이터를 WebSphere Commerce 데이터베이스로 로드하기 위한 유틸리티로 구성됩니다. 로더 패키지를 사용하여 많은 양의 데이터를 로드할 수 있고, 데이터베이스에 있는 데이터를 갱신할 수 있습니다. 로더 패키지에 대한 자세한 정보는 247 페이지의 제 7 부 『상점 공개』를 참조하십시오.
3. 361 페이지의 부록 B 『데이터 작성』에서 정보를 검토하십시오.
4. 회사카탈로그 소유자로서 행동하는 관리 콘솔을 통해 조직을 작성하십시오. 자세한 정보는 WebSphere Commerce 온라인 도움말 주제 "조직 작성"을 참조하십시오.
5. ToolTech 견본 상점을 참조하여 기존 XML 항목과 catalog.xml 파일을 사용하여 마스터 카탈로그에 대해 새 XML 파일을 작성하십시오. 다국어 지원 카탈로그를 작성하는 경우에는 상점이 지원하는 로케일 각각에 대한 개별적인 catalog.xml 파일을 작성하십시오. 로케일 고유의 파일은 모든 설명 정보를 지정해야 하며, 따라서 쉽게 번역할 수 있습니다. 이 예에서, 첫 번째 catalog.xml 파일은 번역할 필요가 없는 모든 정보에 대해 사용되고, 두 번째 catalog.xml은 상점이 지원하는 각 로케일에 대해 사용되며 번역할 필요가 있는 정보를 포함합니다. 또는 원할 경우, ToolTech 견본 상점에서 기존 XML 파일을 사용할 수 있고, 필요한 만큼 정보를 변경할 수 있습니다. ToolTech 견본 상점에서 사용한 catalog.xml 파일은

자체 상점 아카이브 파일에 위치합니다. catalog.xml 파일을 보려면 ZIP 프로그램을 사용하여 상점 아카이브의 압축을 푸십시오. catalog.xml 파일은 다음 데이터 디렉토리에 위치합니다.

- ▶ NT `drive:\WebSphere\CommerceServer\samplstores`
- ▶ 2000 `drive:\Program Files\WebSphere\CommerceServer\samplstores`
- ▶ AIX `/usr/WebSphere/CommerceServer/samplstores`
- ▶ Solaris ▶ Linux `/opt/WebSphere/CommerceServer/samplstores`
- ▶ 400 `/qibm/proddata/WebCommerce/samplstores`

주: WebSphere Commerce 온라인 도움말에는 견본 상점에 포함된 데이터 자원 각각에 대한 정보가 포함됩니다.

catalog.dtd 파일은 다음 디렉토리에 있습니다.

- ▶ NT `drive:\WebSphere\CommerceServer\xml\sar`
- ▶ 2000 `drive:\Program Files\WebSphere\CommerceServer\xml\sar`
- ▶ AIX `/usr/WebSphere/CommerceServer/xml/sar`
- ▶ Solaris ▶ Linux `/opt/WebSphere/CommerceServer/xml/sar`
- ▶ 400 `/qibm/proddata/WebCommerce/xml/sar`

부분 2: 카탈로그 엔티티 작성

1. 다음 ToolTech 견본 상점 예를 참조하여 CATALOG 및 CATALOGDSC 테이블에 정보를 추가하여 카탈로그 엔티티를 작성하십시오. 카탈로그 데이터는 데이터베이스에 있는 카탈로그를 나타냅니다.

```
<catalog
catalog_id="@catalog_id_1"
member_id("&MEMBER_ID;"
identifier="ToolTech"
description="ToolTech Catalog"
tpclevel="0"
/>
```

여기서

- catalog_id는 내부 참조 번호입니다.
 - member_id는 카탈로그 소유자를 식별하는 내부 참조 번호입니다.
 - identifier는 카탈로그에 대한 외부 이름입니다.
 - description은 카탈로그에 대한 설명입니다.
2. 다음 ToolTech 견본 상점 예를 참조하여 번역 목적으로 로케일 고유의 XML 파일에 카탈로그 설명을 추가하십시오.

```
<catalogdsc
catalog_id="@catalog_id_1"
language_id("&en_US;"
name="Store master catalog"
/>
```

여기서

- catalog_id는 해당 언어 지정 정보를 카탈로그에 연결시키는 내부 참조 번호입니다.
- language_id는 언어 식별자입니다.
- name은 카탈로그의 언어 종속 이름입니다.

부분 3: 카탈로그 그룹 작성

1. 다음 ToolTech 견본 상점 예를 참조하여 CATGROUP 및 CATGRPDESC 테이블에 정보를 추가하여 카탈로그 그룹을 작성하십시오. 카테고리로 알려진 카탈로그 그룹은 다른 카탈로그 그룹 또는 상품의 그룹입니다. 카탈로그의 각 카탈로그 그룹에 대한 이 작업을 완료하십시오.

```
<catgroup
catgroup_id="@catgroup_id_1"
member_id("&MEMBER_ID;"
identifier="Woodworking"
markfordelete="0"
/>
```

여기서

- catgroup_id은 카탈로그 그룹의 내부 참조 번호입니다.
- member_id는 카탈로그 소유자를 식별하는 내부 참조 번호입니다.
- identifier는 카탈로그에 대한 외부 이름입니다.
- markfordelete는 카탈로그 그룹이 삭제를 위해 표시되어 있는지 여부를 나타냅니다.
 - 0 = 아니오
 - 1 = 예

2. 다음 ToolTech 견본 상점 예를 참조하여 번역 목적으로 로케일 고유의 XML 파일에 카탈로그 그룹의 설명을 추가하십시오. 카탈로그의 각 카탈로그 그룹에 대한 이 작업을 완료하십시오.

```
<catgrpdesc
language_id("&en_US;"
catgroup_id="@catgroup_id_1"
name="Woodworking"
shortdescription="Woodworking"
longdescription="Woodworking"
published="1"
/>
```

여기서

- language_id는 언어 식별자입니다.
- catgroup_id는 카탈로그 그룹의 내부 참조 번호입니다.
- name은 카탈로그의 언어 종속 이름입니다.
- shortdescription은 카탈로그 그룹의 간단한 설명입니다.
- longdescription는 카탈로그 그룹의 자세한 설명입니다.
- published는 이 카탈로그 그룹이 language_id가 가리키는 언어에 대해 표시되어야 하는지 여부를 나타냅니다.
 - 0 = 아니오
 - 1 = 예

주: 카탈로그 그룹과 설명을 작성할 때마다 catgroup_id는 새 카탈로그 그룹을 표시하기 위해 변경합니다. 예를 들어, catgroup_id="@catgroup_id_2", catgroup_id="@catgroup_id_3", catgroup_id="@catgroup_id_4" 등으로 변경합니다.

3. 카탈로그 그룹을 작성한 후에는 CATTGRP 테이블에 정보를 추가하여 카탈로그에 최상위 레벨 카탈로그 그룹을 지정하십시오. 이 카탈로그 그룹은 바로 아래에 있는 카탈로그 그룹의 상위 항목입니다. 카탈로그의 각 최상위 레벨 카탈로그 그룹에 대한 이 태스크를 완료하십시오. 참조로 ToolTech 견본 상점 예를 사용하십시오.

```
<cattgrp  
catalog_id="@catalog_id_1"  
catgroup_id="@catgroup_id_1"  
>
```

여기서

- catalog_id는 카탈로그의 참조 번호입니다.
- catgroup_id는 카탈로그 그룹의 참조 번호입니다.

주: 카탈로그에 최상위 레벨 카탈로그 그룹을 지정할 때마다, catgroup_id는 새 카탈로그 그룹 연관을 표시하기 위해 수정됩니다. 예를 들어, catgroup_id="@catgroup_id_2", catgroup_id="@catgroup_id_3" 및 catgroup_id="@catgroup_id_4" 등으로 변경합니다.

4. 일단 카탈로그 그룹에 대한 상위 구조와 하위 구조가 결정되면, CATGRPREL 테이블에 정보를 추가하여 카탈로그 그룹 사이의 관계를 작성하십시오. 카탈로그의 각 상위 및 하위 카탈로그 그룹 구조에 대한 이 태스크를 완료하십시오. 참조로 ToolTech 견본 상점 예를 사용하십시오.

```
<catgrprel
catgroup_id_parent="@catgroup_id_1"
catgroup_id_child="@catgroup_id_11"
catalog_id="@catalog_id_1"
sequence="0"
/>
```

여기서,

- catgroup_id_parent는 이 관계의 소스 카탈로그 그룹입니다.
- catgroup_id_child는 이 관계의 대상 카탈로그 그룹입니다.
- catalog_id는 카탈로그의 참조 번호입니다.
- sequence는 카탈로그 그룹의 콘텐츠 표시 순서를 결정하는 번호입니다.

주: 각 카탈로그 그룹 관계에서 catgroup_id_child 및 sequence는 새 관계를 표시하기 위해 수정됩니다. 예를 들어, 후속 관계는

```
catgroup_id_child="@catgroup_id_12", sequence="1",
catgroup_id_child="@catgroup_id_13" 및 sequence="2" 등으로 표시됩니다. 카탈로그에서 탐색 가능한 구조를 사용 중이지 않은 경우, CATGRPREL 관계를 제거할 수 있습니다.
```

부분 4: 재고 정보 작성

1. 다음 ToolTech 견본 상점 예를 참조하여 BASEITEM, BASEITEMDSC, ITEMSPC, ITEMSPC, ITEMVERSN, VERSIONSPC, DISTARRANG 및 STOREITEM 테이블에 정보를 추가하여 재고 정보를 작성하십시오. BASEITEM 테이블에 정보를 추가하고 기본 항목을 작성해서 시작하십시오. 기본 항목들은 공통 이름 및 설명과 함께 상품의 일반 계열을 나타냅니다. 카탈로그의 각 재고 항목의 그룹에 대해 이 태스크를 완료하십시오.

```
<baseitem
baseitem_id="@baseitem_id_102"
member_id="&MEMBER_ID;"
markfordelete="0"
partnumber="tooltech_sku_102"
itemtype_id="ITEM"
quantitymeasure="C62"
quantitymultiple="1.0"
/>
```

여기서

- baseitem_id는 생성된 고유 키입니다.
- member_id는 기본 항목 소유자입니다.
- markfordelete는 기본 항목이 삭제를 위해 표시되어 있는지 여부를 나타냅니다.
 - 0 = 아니오
 - 1 = 예

- partnumber는 소유자에 대한 기본 항목을 고유하게 식별합니다.
- itemtype_id는 기본 항목의 유형입니다.
 - ITEM = 항목, 패키지 또는 번들
 - DNKT = 동적 제품군
- quantitymeasure는 여러 수량에 대한 측정 단위입니다.
- quantitymultiple은 통합 단위에서 측정되는 기본 항목의 총계입니다. quantitymeasure와 함께, 이것은 각 통합 단위가 표시하는 방법을 지시합니다.

주: 카탈로그를 작성할 모든 상품에 대한 기본 항목을 작성해야 합니다. 기본 항목을 작성할 때마다 baseitem_id 및 partnumber는 새 기본 항목을 작성하기 위해 번호를 변경합니다. 예를 들어, 새 기본 항목은 항목으로 baseitem_id="@baseitem_id_147"과 partnumber="tooltech_sku_147"을 포함하지만, 다른 기본 항목은 항목으로 baseitem_id="@baseitem_id_192"와 partnumber="tooltech_sku_192" 등을 포함합니다.

2. 다음 ToolTech 견본 상점 예를 참조하여 데이터베이스에 지정된 항목에 대해 정보를 추가하십시오. 지정된 항목은 관련된 모든 속성에 대한 값이 있는 항목이며 카탈로그에서 항목, 패키지, 번들 또는 동적 제품군을 나타냅니다. 카탈로그의 지정된 각 항목에 대한 이 태스크를 완료하십시오.

```
<itemspc
itemspc_id="@itemspc_id_106"
baseitem_id="@baseitem_id_102"
markfordelete="0"
partnumber="T0000106"
member_id("&MEMBER_ID;"
discontinued="N"
/>
```

여기서

- itemspc_id는 생성된 고유 키입니다.
- baseitem_id은 상품 기본 항목입니다.
- markfordelete는 지정된 항목이 삭제를 위해 표시되어 있는지 여부를 나타냅니다.
 - 0 = 아니오
 - 1 = 예
- partnumber는 소유자에 대한 지정된 항목을 고유하게 식별합니다.
- member_id는 지정된 항목의 소유자입니다.
- discontinued는 지정된 항목이 중단되었는지 여부를 나타냅니다.
 - Y = 중단됨. 충분한 재고가 있지만 백오더할 수 없는 경우 명령할 수 있음
 - N = 활성화. 재고 없는 경우 백오더 될 수 있음

주: 카탈로그에서 작성할 각 항목에 대해 지정된 항목을 작성해야 합니다. 지정된 항목을 정의할 때마다, `itemspc_id="@itemspc_id_107"`, `baseitem_id="@baseitem_id_102"`, `partnumber="T0000107"`는 지정된 새 항목을 작성하기 위해 번호를 변경합니다. 예를 들어, 지정된 새 항목은 항목으로 `itemspc_id="@itemspc_id_108"`, `baseitem_id="@baseitem_id_102"` 및 `partnumber="T0000108"`을 포함하지만, 지정된 다른 항목은 항목으로 `itemspc_id`, `baseitem_id` 및 `partnumber` 등을 포함합니다.

3. 다음 ToolTech 견본 상점 예를 참조하여 상품 버전과 데이터베이스에 대한 기본 항목 사이의 관계에 대해 다음 정보를 추가하십시오. 카탈로그의 이러한 각 관계에 대한 이 태스크를 완료하십시오.

```
<itemversn
itemversn_id="@itemversn_id_102"
baseitem_id="@baseitem_id_102"
expirationdate="2010-01-01 00:00:00.000000"
versionname="version"
/>
```

여기서

- `itemversn_id`는 항목 버전을 식별하는 생성된 참조 번호입니다.
- `baseitem_id`은 기본 항목입니다.
- `expirationdate`는 항목 버전이 만기하는 시점입니다.
- `versionname`은 기본 항목에 대한 항목 버전을 고유하게 식별합니다.

주: 항목 버전과 기본 항목 간의 관계를 작성할 때마다, `itemversn_id` 및 `baseitem_id` 번호는 새로운 관계를 작성하기 위해 변경됩니다. `baseitem_id`는 기존의 기본 항목과 일치합니다. 예를 들어, 새 관계는 항목으로 `itemversn_id="@itemversn_id_107"` 및 `baseitem_id="@baseitem_id_107"`을 포함하지만, 다른 관계는 항목으로 `itemversn_id="@itemversn_id_108"` 및 `baseitem_id="@baseitem_id_108"` 등을 포함합니다.

4. 다음 ToolTech 견본 상점 예를 참조하여 상품 버전과 데이터베이스에 지정된 항목 사이의 관계에 대해 다음 정보를 추가하십시오. 카탈로그의 이러한 각 관계에 대한 이 태스크를 완료하십시오.

```
<versionspc
versionspc_id="@versionspc_id_106"
itemspc_id="@itemspc_id_106"
itemversn_id="@itemversn_id_102"
/>
```

여기서

- `versionspc_id`는 생성된 고유 식별자입니다.
- `itemspc_id`는 카탈로그 항목과 관련된 지정된 항목입니다.
- `itemversn_id`는 항목 버전을 식별합니다.

주: 상품 버전과 지정된 항목 사이의 관계를 작성할 때마다 versionspc_id와 itemspc_id는 새 관계를 작성하기 위해 번호를 변경합니다. itemspc_id는 기존의 지정된 항목과 일치합니다. 예를 들어, 새 관계는 항목으로 versionspc_id="@versionspc_id_107"과 itemspc_id="@itemspc_id_107"을 포함하지만, 다른 관계는 항목으로 versionspc_id="@versionspc_id_108"과 itemspc_id="@itemspc_id_108" 등을 포함합니다.

5. 다음 ToolTech 견본 상점 예를 참조하여 데이터베이스에 분배 배열을 추가하십시오. 분배 협정은 상점이 재고를 판매할 수 있게 합니다. 카탈로그에서 각 분배 협정에 대해 이 태스크를 완료하십시오.

```
<distarrang
distarrang_id="@distarrang_id_102"
wholesalestore_id="@storeent_id_1"
merchantstore_id="@storeent_id_1"
baseitem_id="@baseitem_id_102"
pickingmethod="F"
startdate="2000-12-25 00:00:00.000000"
enddate="2010-01-01 00:00:00.000000"
/>
```

여기서,

- distarrang_id는 분배 협정의 참조 번호입니다.
- wholesalestore_id는 판매자 상점에서 판매할 수 있는 재고를 소유하는 도매 상점이다. 이 도매 상점은 merchantstore_id와 동일해야 합니다.
- merchantstore_id는 도매 상점의 재고에서 판매할 수 있는 판매자 상점입니다. 이 판매자 상점은 wholesalestore_id와 동일해야 합니다.
- baseitem_id는 분배 협정에 포함되는 상품입니다.
- pickingmethod는 이 협정 아래의 RECEIPT 테이블에서 오더피킹되는 재고의 순서를 판별합니다.
 - F = FIFO(First In First Out) - 가장 먼저 받은 재고
 - L = LIFO(Last in First Out) - 가장 최근에 받은 재고
- startdate는 분배 협정이 유효하게 되는 시작 시점입니다.
- enddate는 분배 협정이 유효하지 않게 되는 시점입니다.

주: 분배 협정을 작성할 때마다, distarrang_id 및 baseitem_id 번호는 새 분배 재열을 작성하기 위해 변경됩니다. 예를 들어, 두 번째 분배 협정은 값 distarrang_id="@distarrang_id_147" 및 baseitem_id="@baseitem_id_147"을 포함할 수 있으며, 세 번째 분배 협정은 distarrang_id="@distarrang_id_192" 및 baseitem_id="@baseitem_id_192" 등을 포함할 수 있습니다.

6. ToolTech 견본 상점 예를 사용하여, 데이터베이스에 대한 특정 기본 항목의 지정된 항목에 대해 재고를 할당하는 방식에 영향을 미치는 속성을 추가하십시오. 카탈로그의 각 기본 항목에 대한 이 태스크를 완료하십시오.

```
<storeitem
baseitem_id="@baseitem_id_102"
storeent_id="@storeent_id_1"
trackinventory="Y"
forcebackorder="N"
releaseseparately="N"
returnnotdesired="N"
backorderable="Y"
creditable="Y"
minqtyforsplit="0"
/>
```

여기서,

- baseitem_id은 기본 항목입니다.
- storeent_id는 상점 또는 상점 그룹입니다.
- trackinventory는 재고가 RECEIPT 테이블에서 추적되는지 여부를 제어합니다.
 - N = 재고가 추적되지 않으며 RECEIPT 테이블에 항목이 없습니다.
 - Y = 재고가 RECEIPT 테이블에서 추적됩니다.
- forcebackorder는 기본 항목에 지정된 항목의 할당을 일시적으로 보류합니다.
 - N = 재고가 할당될 수 있습니다(정상 행위).
 - Y = 충분한 재고가 있어도 재고를 할당할 수 없습니다.
- releaseseparately는 기본 항목의 지정 항목에 대하여 지정된 주문 항목이 출고되는 방식을 제어합니다.
 - N = 주문 항목이 다른 주문 항목과 함께 출고될 수 있습니다.
 - Y = 주문 항목이 개별적으로 출고되어야 합니다(자체 상자에서).
- returnnotdesired는 고객이 반품하려고 하거나 할 수 있는 경우에도, 항목 반품이 되지 않음을 나타냅니다.
 - N = 항목을 반품하려고는 하지만 하지 않을 것으로 예측되는 고객의 의도에 근거하여 평가되는 신용에 대한 요청
 - Y = 반품이 예상되는 것처럼 평가되는 신용에 대한 요청
- backorderable은 기본 항목에 지정된 항목이 백오더될 수 없음을 나타냅니다.
 - N = 항목을 백오더할 수 없습니다.
 - Y = 항목을 백오더할 수 있습니다.
- creditable은 대체하지 않고 판매자가 이 항목에 대한 신용을 발행할 지 여부를 나타냅니다.
 - N = 현상태대로 판매

- Y = 신용 가능

- minqtyforsplit는 할당되지 않은 나머지 수량이 지정된 최소 수량보다 적은 경우, 주문 항목이 재고 할당 중 자동으로 분할되지 않음을 나타냅니다.

주: 저장 항목에 대한 재고 할당 규칙을 정의할 때마다, baseitem_id 번호는 새 기본 항목을 나타내기 위해 변경됩니다. 예를 들어, 새 할당은

baseitem_id="@baseitem_id_147"을 포함할 수 있지만 세 번째 할당은 baseitem_id="@baseitem_id_192" 등을 포함할 수 있어야 합니다.

7. 다음 ToolTech 견본 상점 예를 참조하여 번역 목적으로 로케일 고유의 XML 파일에 기본 항목 설명을 추가하십시오. 카탈로그의 각 기본 항목 설명에 대한 이 태스크를 완료하십시오.

```
<baseitmdsc
baseitem_id="@baseitem_id_102"
language_id("&en_US;"
shortdescription="Circular Saw"
longdescription="Light on weight but not in quality. The Circular Saw weighs a maximum of 10.9lbs., with a choice of a 12 or 14 amp motor, and speeds of up to 600 rpms! Low friction 220V aluminum alloy shoe will ensure the job gets done on time."
/>
```

여기서

- baseitem_id은 생성된 고유 키입니다.
- language_id는 이 정보의 언어입니다.
- shortdescription은 기본 항목의 간단한 설명입니다.
- longdescription은 기본 항목의 자세한 설명입니다.

부분 5: 카탈로그 항목 작성

1. 다음 ToolTech 견본 상점 예를 참조하여 CATENTRY 및 CATENTDESC 테이블에 정보를 추가하여 카탈로그 항목을 작성하십시오. 상품, 항목, 패키지, 번들 및 동적 제품군 등 각 유형의 카탈로그 항목은 카탈로그의 판매를 목적으로 한 주문 가능 판매 계획의 항목을 나타냅니다. 각 상품 카탈로그 항목에 대해 기본 항목을 정의할 필요가 있습니다. 카탈로그의 각 상품 카탈로그 항목에 대한 이 태스크를 완료하십시오.

```
<catentry
catentry_id="@product_id_102"
baseitem_id="@baseitem_id_102"
member_id("&MEMBER_ID"
catenttype_id="ProductBean"
partnumber="T0000102"
mfpartnumber="Sprain-Tools-102"
mfname="Sprain Tools"
markfordelete="0"
buyable="1"/>
```

여기서

- `catentry_id`는 상품 카탈로그 항목의 내부 참조 번호입니다.
- `baseitem_id`는 카탈로그 항목과 관련된 기본 항목입니다.
- `member_id`는 카탈로그 항목을 식별하는 참조 번호입니다.
- `catenttype_id`는 다음과 같은 카탈로그 항목의 유형을 식별합니다.
 - `ItemBean` = 항목 식별
 - `ProductBean` = 상품 식별
 - `PackageBean` = 패키지 식별
 - `BundleBean` = 번들 식별
 - `DynamicKitBean` = 동적 제품군 식별
- `partnumber`는 카탈로그 항목의 부품 번호를 식별하는 참조 번호입니다.
- `mfpartmentnumber`는 카탈로그 항목을 식별하기 위해 제조업체가 사용하는 부품 번호입니다.
- `mfname`는 카탈로그 항목의 제조업체 이름입니다.
- `markfordelete`는 카탈로그 항목이 삭제를 위해 표시되어 있는지 여부를 나타냅니다.
 - 0 = 아니오
 - 1 = 예
- `buyable`는 개별적으로 카탈로그 항목을 구매할 수 있는지 여부를 나타냅니다.
 - 0 = 아니오
 - 1 = 예

주: 기본 항목을 카탈로그 항목에 추가할 때마다, `catentry_id`와 `baseitem_id`는 새 카탈로그 항목을 나타내기 위해 순서를 변경합니다. `catenttype_id`는 카탈로그 항목의 유형에 따라 다릅니다.

- 다음 ToolTech 견본 상점 예를 참조하여 각 카탈로그 항목에 대해 지정된 항목을 정의하십시오. 카탈로그의 각 카탈로그 항목에 대한 이 태스크를 완료하십시오.

```
<catentry
catentry_id="@catentry_id_106"
itemspc_id="@itemspc_id_106"
member_id="&MEMBER_ID"
catenttype_id="ItemBean"
partnumber="T0000106"
mfpartmentnumber="Sprain-Tools-106"
mfname="Sprain Tools"
markfordelete="0"
buyable="1"/>
```

여기서,

- catentry_id는 카탈로그 항목의 내부 참조 번호입니다.
- itemspc_id는 카탈로그 항목이 속하는 지정된 항목입니다.
- member_id는 카탈로그 항목을 식별하는 참조 번호입니다.
- cattentype_id는 다음과 같은 카탈로그 항목의 유형을 식별합니다.
 - ItemBean = 항목 식별
 - ProductBean = 상품 식별
 - PackageBean = 패키지 식별
 - BundleBean = 번들 식별
 - DynamicKitBean = 동적 제품군 식별
- partnumber는 카탈로그 항목의 부품 번호를 식별하는 참조 번호입니다.
- mfpartmentnumber는 카탈로그 항목을 식별하기 위해 제조업체가 사용하는 부품 번호입니다.
- mfname은 카탈로그 항목의 제조업체 이름입니다.
- markfordelete는 카탈로그 항목이 삭제를 위해 표시되어 있는지 여부를 나타냅니다.
 - 0 = 아니오
 - 1 = 예
- buyable은 개별적으로 카탈로그 항목을 구매할 수 있는지 여부를 나타냅니다.
 - 0 = 아니오
 - 1 = 예

주: 지정된 항목을 카탈로그 항목에 추가할 때마다, catentry_id 및 itemspc_id 순서는 새 카탈로그 항목을 나타내기 위해 변경됩니다. cattentype_id는 카탈로그 항목의 유형에 따라 다릅니다. 마스터 카탈로그 구조 제한하에서, 카탈로그 항목은 둘 이상의 카테고리에 속할 수 없습니다. 둘 이상의 카탈로그에 카탈로그 항목을 위치시키려면, 탐색 카테고리를 사용해야 합니다.

- 다음 ToolTech 견본 상점 예를 참조하여 로케일 고유의 XML 파일에 설명을 추가하십시오. 카탈로그의 각 카탈로그 항목 설명에 대한 이 태스크를 완료하십시오.

```
<catentdesc
catentry_id="@product_id_102"
language_id("&en_US"
name="Circular"
shortdescription="Circular Saw"
longdescription="Light on weight but not in quality. The Circular Saw weighs a maximum of 10.9lbs., with a choice of a 12 or 14 amp motor, and speeds of up to 600 rpms! Low friction 220V aluminum alloy shoe will ensure the job gets done on time."
thumbnail="images/circular_saw_sm.gif"
```

```
fullimage="images/circular_saw.gif"
available="1"
published="1"
/>
```

여기서

- catentry_id는 이 언어 특정 정보와 관련되어 있는 카탈로그 항목을 표시하는 내부 참조 번호입니다.
- language_id는 언어 식별자입니다.
- name은 카탈로그 항목의 언어 종속 이름입니다.
- shortdescription은 카탈로그 항목의 간단한 설명입니다.
- longdescription은 카탈로그 항목의 자세한 설명입니다.
- thumbnail은 소형 이미지 경로입니다.
- fullimage는 전체 이미지 경로입니다.
- available은 카탈로그 항목의 사용 가능한 시간의 길이를 나타냅니다.
- published는 language_id가 표시한 언어에 대해 이 카탈로그 항목을 표시해야 하는지 여부를 나타냅니다.
 - 0 = 표시
 - 1 = 표시 안함

부분 6: 속성 및 속성값 작성

1. 다음 ToolTech 견본 상점 예를 참조하여 번역 목적으로 로케일 고유의 XML 파일에서 ATTRIBUTE 및 ATTRVALUE 테이블에 정보를 추가하여 제품에 대한 속성 및 속성값을 작성하십시오. 카탈로그의 각 상품에는 셔츠나 바지의 크기 및 색상과 같이 특정 속성 세트가 있습니다. 항목은 속성값 별로 정의됩니다. 예를 들어, 중간 크기의 검은색 셔츠는 셔츠가 상품인 동안에는 하나의 항목입니다. 카탈로그의 각 속성에 대한 이 태스크를 완료하십시오.

```
<attribute
attribute_id="@attribute_id_103"
language_id("&en_US"
attrtype_id="STRING"
name="Amps"
sequence="0"
description="Amps"
catentry_id="@product_id_102"
description2="Amps"
/>
```

여기서,

- attribute_id는 속성의 내부 참조 번호입니다.
- language_id는 이 속성이 부착되어 있는 언어입니다.
- attrtype_id는 해당 속성값 유형입니다.

- name은 속성 이름입니다.
- sequence는 제공된 상품에 대한 속성 표시 순서를 결정하는 순서 번호입니다.
- description은 속성 설명입니다.
- catentry_id는 이 속성이 속해 있는 상품의 참조 번호입니다.
- description2는 속성의 추가 설명입니다.

주: 상품에 catentry_id가 정의한 속성을 추가할 때마다 attribute_id는 새 속성을 나타내기 위해 순서를 변경합니다.

2. 다음 ToolTech 견본 상점 예를 참조하여 속성값을 추가하십시오. 카탈로그의 각 속성값에 대한 이 태스크를 완료하십시오.

```
<attrvalue
attrvalue_id="@attrvalue_id_114"
language_id="&en_US"
attribute_id="@attribute_id_103"
name="12.0amps"
attrtype_id="STRING"
stringvalue="12.0amps"
sequence="0"
catentry_id="@catentry_id_106"
/>
```

여기서

- attrvalue_id는 속성값의 내부 참조 번호입니다.
- language_id는 이 속성이 부착되어 있는 언어입니다.
- attribute_id는 값과 연관되어 있는 속성의 내부 참조 번호입니다.
- name은 속성값 이름입니다.
- attrtype_id는 속성값 유형입니다.
- stringvalue는 속성값입니다.
- sequence는 제공된 속성에 대한 속성값 표시 순서를 결정하는 순서 번호입니다.
- catentry_id는 이 속성값이 설명하는 항목 ID입니다.

주: 속성에 속성값을 추가할 때마다 attrvalue_id는 다른 값을 나타내기 위해 순서를 변경합니다. attribute_id는 다른 속성을 나타내기 위해 순서를 변경합니다. sequence는 새로운 각 속성값과 함께 증가합니다. 예를 들어, 후속 속성값은 sequence="1", sequence="2", sequence="3" 등이 됩니다.

부분 7: 상품과 항목 사이의 관계 작성

1. 카탈로그의 상품 및 항목을 작성한 후, CATENTREL 테이블에 정보를 추가하여 각 상품과 항목 간의 관계를 정의해야 합니다. 다음 ToolTech 견본 상점 예를 참조하십시오. 카탈로그의 각 상품과 항목 관계 값에 대한 이 태스크를 완료하십시오.

```
<catentrel
catentry_id_parent="@product_id_147"
catreltype_id="PRODUCT_ITEM"
catentry_id_child="@catentry_id_152"
sequence="2"
quantity="1"
/>
```

여기서

- `catentry_id_parent`는 이러한 관계를 가진 소스 카탈로그 항목의 참조 번호, 즉 상품입니다.
- `catreltype_id`는 다음과 같은 관계 유형입니다(예: `PRODUCT_ITEM`).
- `catentry_id_child`는 이러한 관계를 가진 대상 카탈로그 항목의 참조 번호, 즉 항목입니다.
- `sequence`는 표시 순서를 결정하는 데 사용되는 순서 번호입니다.
- `quantity`는 관계와 연관될 수 있는 수량입니다.

주: 상품과 항목간에 관계를 추가할 때마다 `catentry_id_parent`와 `catentry_id_child`는 `catreltype_id`에 따라 다른 관계를 작성하기 위해 번호를 변경합니다. 새로운 각 관계에 따라 `sequence` 번호가 다릅니다. 예를 들어, `sequence="2"`라면 다음 관계는 `sequence="3"`이고, 그 다음은 `sequence="4"` 등이 됩니다.

부분 8: 패키지 및 번들 작성

1. 일단 상품과 항목을 작성했다면 `CATENTRY`, `CATENTDESC` 및 `CATENTREL` 테이블에 정보를 추가하여 패키지 및 번들을 작성해야 합니다. 다음 ToolTech 건 본 상점 예를 참조하여 `CATENTRY` 테이블에 정보를 추가하여 패키지나 번들 작성을 시작해야 합니다. 카탈로그의 각 패키지 및 번들에 대한 이 태스크를 완료하십시오.

```
<catentry
catentry_id="@package_id_102"
member_id("&MEMBER_ID"
catenttype_id="PackageBean"
partnumber="sku-@package_id_102"
mfpartnumber="sku-@package_id_102"
mfname="ToolTech"
markfordelete="0"
buyable="1"/>
```

여기서,

- `catentry_id`는 카탈로그 항목의 참조 번호입니다.
- `member_id`는 카탈로그 항목의 소유자를 식별하는 참조 번호입니다.
- `catenttype_id`는 다음과 같은 카탈로그 항목의 유형을 식별합니다.
 - `PackageBean` = 패키지 식별

- BundleBean = 번들 식별
- partnumber는 카탈로그 항목의 부품 번호를 식별하는 참조 번호입니다.
- mfpartmentnumber는 카탈로그 항목을 식별하기 위해 제조업체가 사용하는 부품 번호입니다.
- mfname은 카탈로그 항목의 제조업체 이름입니다.
- markfordelete는 카탈로그 항목을 삭제하도록 표시되어 있는지 여부를 나타냅니다.
 - 0 = 아니오
 - 1 = 예
- buyable은 카탈로그 항목을 개별적으로 구매할 수 있는지 여부를 나타냅니다.
 - 0 = 아니오
 - 1 = 예

주: 패키지나 번들을 작성할 때마다, catentry_id, partnumber 및 mfpartmentnumber 번호는 다른 패키지나 번들을 작성하도록 변경됩니다. 예를 들어, 새 패키지를 작성하기 위해 catentry_id="@package_id_103", partnumber="sku-@package_id_103" 및 mfpartmentnumber="sku-@package_id_103"을 사용하고, 패키지로 항목을 식별하기 위해 catenttype_id="PackageBean"을 포함할 수 있습니다. 새 번들을 작성하기 위해 catentry_id="@package_id_110", partnumber="sku-@package_id_110" 및 mfpartmentnumber="sku-@package_id_110"을 사용하고, catenttype_id="BundleBean"을 사용하여 번들로 항목을 식별하는 등의 작업을 할 수 있습니다.

- 다음 ToolTech 견본 상점 예를 참조하여 번역 목적으로 로케일 고유의 XML 파일의 CATENTDESC 테이블에 정보를 추가하여 패키지나 번들 설명을 추가하십시오. 카탈로그의 각 패키지 및 번들에 대한 이 태스크를 완료하십시오.

```
<catentdesc
catentry_id="@catentry_id_102"
language_id="-1"
name="computer"
shortdescription="Computer"
longdescription="A combination of a central processing unit, monitor,
hard drive, and color printer. An ideal starter system."
thumbnail="images/package_system_sm.gif"
fullimage="images/package_system.gif"
available="1"
published="1"
/>
```

여기서

- catentry_id는 이 언어 특정 정보와 관련되어 있는 카탈로그 항목을 표시하는 내부 참조 번호입니다.

- language_id는 언어 식별자입니다.
- name은 카탈로그 항목의 언어 종속 이름입니다.
- shortdescription은 카탈로그 항목의 간단한 설명입니다.
- longdescription은 카탈로그 항목의 자세한 설명입니다.
- thumbnail은 카탈로그 항목의 소형 이미지 경로입니다.
- fullimage는 카탈로그 항목의 전체 이미지 경로입니다.
- available은 카탈로그 항목의 사용 가능한 시간의 길이를 나타냅니다.
- published는 language_id가 표시한 언어에 대해 이 카탈로그 항목을 표시해야 하는지 여부를 나타냅니다.
 - 0=카테고리 항목을 표시 안함
 - 1=카테고리 항목을 표시함
- 다음 ToolTech 견본 상점 예를 참조하여 CATENTREL 테이블에 정보를 추가하여 패키지나 번들과 구성요소 간의 관계를 작성하십시오. 카탈로그의 각 패키지나 번들 구성요소 관계에 대한 이 태스크를 완료하십시오.

```
<catentrel
catentry_id_parent="@catentry_id_102"
catreltype_id="PACKAGE_COMPONENT"
catentry_id_child="@catentry_id_97"
sequence="1.0"
quantity="1.0"
/>
```

여기서,

- catentry_id_parent는 이러한 관계를 가진 소스 카탈로그 항목의 참조 번호, 즉 패키지나 번들입니다.
- catreltype_id는 이러한 관계의 유형입니다.
 - PACKAGE_COMPONENT는 패키지와 패키지 구성요소 간의 관계를 표시합니다.
 - BUNDLE_COMPONENT는 번들 및 번들 구성요소 간의 관계를 표시합니다.
- catentry_id_child는 이러한 관계를 가진 대상 카탈로그 항목의 참조 번호, 즉 구성요소입니다.
- sequence는 표시 순서를 결정하는 데 사용되는 순서 번호입니다.
- quantity는 관계와 연관될 수 있는 수량입니다.

주: 패키지와 번들 간의 관계를 작성할 때마다 catentry_id_parent와 catentry_id_child는 기존의 카탈로그 항목들을 일치시키기 위해 번호를

변경합니다. 새로운 각 관계에 따라 sequence 번호가 다릅니다. 예를 들어, sequence="1.0"으로 시작하면 다음 관계는 sequence="2.0" 이고, 그 다음은 sequence="3.0" 등이 됩니다.

부분 9: 카탈로그 그룹과 카탈로그 항목 사이의 관계 작성

1. 카탈로그의 카탈로그 그룹과 카탈로그 항목을 작성한 후, CATGPENREL 테이블에 정보를 추가하여 카탈로그 그룹과 카탈로그 항목 간의 관계를 정의해야 합니다. 마스터 카탈로그 구조 제한하에서, 카탈로그 항목은 둘 이상의 카테고리에 속할 수 없습니다. 둘 이상의 카탈로그에 카탈로그 항목을 위치시키려면, 탐색 카테고리를 사용해야 합니다. 다음 ToolTech 견본 상점 예를 참조하십시오. 카탈로그의 각 카탈로그 그룹과 카탈로그 항목 관계에 대한 이 태스크를 완료하십시오.

```
<catgpenrel
catgroup_id="@catgroup_id_11"
catalog_id="@catalog_id_1"
catentry_id="@product_id_102"
sequence="0"
/>
```

여기서,

- catgroup_id는 이러한 관계의 소스 카탈로그 그룹입니다.
- catalog_id는 내부에서 이러한 관계를 볼 수 있는 카탈로그입니다.
- catentry_id는 이러한 관계의 대상 카탈로그 항목입니다.
- sequence는 카탈로그 그룹의 콘텐츠 표시 순서를 결정하는 순서 번호입니다.

주: 카탈로그 그룹과 카탈로그 항목간의 관계를 작성할 때마다 catgroup_id와 catentry_id는 서로 다른 카탈로그 그룹과 카탈로그 항목 간에 새로운 관계를 형성하기 위해 번호를 변경합니다. 새로운 각 관계에 따라 sequence 번호가 다릅니다. 예를 들어, sequence="0"으로 시작하면 다음 관계는 sequence="1"이고, 그 다음은 sequence="2" 등이 됩니다.

부분 10: 판매 계획 연관 작성

1. 다음 ToolTech 견본 상점 예를 참조하여 MASSOCECE 테이블에 정보를 추가하여 카탈로그 항목 간의 판매 계획 연관을 작성해야 합니다. 카탈로그의 각 판매 계획 연관에 대한 이 태스크를 완료하십시오.

```
<massoccece
massoccece_id="@relationship_id_100"
massoctype_id="X-SELL"
catentry_id_from="@product_id_1"
catentry_id_to="@product_id_15"
massoc_id="REQUIRES"
quantity="2.0"
rank="1.00000"
/>
```

여기서,

- massoccece_id는 이 항목의 참조 번호입니다.
- massotype_id는 다음과 같은 연관 유형의 식별자입니다.
 - X-SELL = 비슷한 모델 소개
 - UPSELL = 나은 모델 소개
 - ACCESSORY = 부속 상품
- catentry_id_from은 연관 소스인 카탈로그 항목입니다.
- catentry_id_to는 연관 대상인 카탈로그 항목입니다.
- massoc_id는 의미 지정자의 식별자입니다.
 - REQUIRED
 - OPTIONAL
 - COMES WITH
- quantity는 이 연관과 관련된 수량입니다.
- rank는 표시 순서를 나타내기 위해 사용되는 주문 번호입니다.

주: 판매 계획 연관을 추가할 때마다, massoccece_id는 새 관계를 나타내기 위해 번호를 변경합니다. catentry_id_from와 catentry_id_to는 연관에 대한 새 판매 계획 콘텐츠를 작성하기 위해 번호를 변경합니다.

부분 11: 상점에 카탈로그를 연관

1. 참조로 ToolTech 견본 상점에서 기존의 storecatalog.xml을 사용하여 데이터베이스의 상점에 카탈로그, 카탈로그 그룹 그리고 카탈로그 항목들을 지정하여 카탈로그를 상점과 연관시켜야 합니다. 또 카탈로그 그룹과 카탈로그 항목에 표시 페이지를 지정해야 합니다. 이 정보를 STORECAT, STORECENT, STORECGRP, DISPCGPREL 및 DISPENTREL 테이블에 추가하십시오. 다국어 지원 카탈로그를 작성하는 경우에는 상점이 지원하는 로케일 각각에 대한 개별적인 상점 카탈로그 관계 XML 파일을 작성하십시오.

```
<storecat
catalog_id="@catalog_id_1"
storeent_id="@storeent_id_1"
mastercatalog="1"
/>
```

여기서

- catalog_id는 카탈로그의 참조 번호입니다.
- storeent_id는 데이터베이스의 상점 엔티티 참조 번호입니다.
- mastercatalog는 상점에 대한 마스터 카탈로그를 지정합니다. 1이라는 값은 이 카탈로그가 마스터 카탈로그로 지정되었음을 나타냅니다.

2. 다음 ToolTech 견본 상점 예를 참조하여 상점 카탈로그 관계에 카탈로그 항목을 추가하십시오. 카탈로그의 각 카탈로그 항목에 대한 이 태스크를 완료하십시오.

```
<storecent
storeent_id="@storeent_id_1"
catentry_id="@product_id_102"
/>
```

여기서

- storeent_id는 데이터베이스의 상점 엔티티 참조 번호입니다.
- catentry_id는 카탈로그 항목의 참조 번호입니다.

주: 상점 엔티티에 catentry_id를 추가할 때마다 참조 번호가 기존의 카탈로그 항목과 일치시키기 위해 변경됩니다.

3. 다음 ToolTech 견본 상점 예를 참조하여 상점 엔티티에 카탈로그 그룹을 추가하십시오. 카탈로그의 각 카탈로그 그룹에 대한 이 태스크를 완료하십시오.

```
<storecgrp
storeent_id="@storeent_id_1"
catgroup_id="@catgroup_id_1"
/>
```

여기서

- storeent_id는 데이터베이스의 상점 엔티티 참조 번호입니다.
- catgroup_id는 카탈로그 그룹의 참조 번호입니다.

주: 상점 엔티티에 catgroup_id를 추가할 때마다 참조 번호가 기존의 카탈로그 그룹과 일치시키기 위해 변경됩니다.

부분 12: 카탈로그에 세금 연관

1. 특정 상점의 카탈로그에 있는 상품과 서비스에 세금을 연관시켜야 합니다. CATENCALCD 테이블에 이 정보를 추가하여 세금 계산 코드와 카탈로그 항목을 연관시켜야 합니다. 자세한 정보는 174 페이지의 『WebSphere Commerce에서 세금 자원 작성』을 참조하십시오.

부분 13: 카탈로그에 운송 방법 연관

1. 카탈로그의 상품 및 서비스에 운송 방법을 연관시키려면, 운송 계산 코드를 카탈로그 항목과 연관시켜야 합니다. CATENCALCD 테이블에 이 정보를 추가하십시오. 자세한 정보는 153 페이지의 『WebSphere Commerce에서 운송 자원 작성』을 참조하십시오.

부분 14: 카탈로그에 서비스 센터 연관

1. 고객에게 상품을 운송하려면 서비스 센터와 카탈로그를 연관시켜야 합니다. 서비스 센터는 상점의 상품 재고 및 운송을 관리합니다. FFMCENTER 테이블에 이 정보를 추가하십시오. 자세한 정보는 124 페이지의 『WebSphere Commerce에서 이항 자원 작성』을 참조하십시오.

부분 15: 카탈로그 항목에 대한 가격 작성

1. 카탈로그 항목에 대한 가격 책정을 작성하십시오. 가격 책정은 해당 가격을 사용하기 위해 만족해야 하는 기준 및 카탈로그 항목의 가격 범위를 나타냅니다. 기능적인 카탈로그를 작성하려면 데이터베이스에 판매 설정 정보를 추가해야 합니다. TRADEPOSCN, TDPSCNCNTR, MGPTRDPSCN, OFFER 및 OFFERPRICE 테이블에 이 정보를 추가하십시오. 자세한 정보는 103 페이지의 『WebSphere Commerce에서 가격 책정 자원 작성』을 참조하십시오. 또는 WebSphere Commerce 액셀러레이터의 상품 관리 도구를 사용하여 카탈로그 항목에 대한 가격 책정을 작성 또는 갱신할 수 있습니다.

부분 16: XML 파일 로드






1. 데이터를 작성한 다음 로더 패키지를 사용하거나 상점 서비스의 공개 기능을 사용하여 XML 파일을 데이터베이스로 로드하십시오. 로더 패키지에 대한 자세한 정보는 247 페이지의 제 7 부 『상점 공개』를 참조하십시오.

주: 또한 WebSphere Commerce 액셀러레이터에서 상품 관리 도구를 사용하여 마스터 카탈로그의 카탈로그 자원을 작성할 수 있습니다. 상품 관리 도구에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오.

상점 카탈로그 자원 표시

카탈로그, 카탈로그 그룹 및 카탈로그 항목을 상점에 연관시킨 후에, 데이터베이스에서 이 관계를 작성하여 카탈로그 항목 및 카탈로그 그룹을 표시하기 위해 JSP 템플릿을 지정하십시오. 로더 패키지를 사용하여 데이터베이스로 로드할 수 있는 XML 파일 포맷으로 이러한 관계를 작성하십시오.

ToolTech 견본 상점에서 사용한 storecatalog.xml 파일은 자체 상점 아카이브 파일에 위치해 있습니다. storecatalog.xml 파일을 보려면, ZIP 프로그램을 사용하여 상점 아카이브의 압축을 푸십시오. storecatalog.xml 파일은 다음 데이터 디렉토리에 있습니다.

-  NT `drive:\WebSphere\CommerceServer\samplstores`
-  2000 `drive:\Program Files\WebSphere\CommerceServer\samplstores`
-  AIX `/usr/WebSphere/CommerceServer/samplstores`
-  Solaris  Linux `/opt/WebSphere/CommerceServer/samplstores`

- ▶ 400 /qibm/proddata/WebCommerce/samplestores

주: WebSphere Commerce 온라인 도움말에는 견본 상점에 포함된 데이터 자원 각각에 대한 정보가 포함됩니다.

store-catalog.dtd 파일은 다음 디렉토리에 있습니다.

- ▶ NT drive:\WebSphere\CommerceServer\xml\sar
- ▶ 2000 drive:\Program Files\WebSphere\CommerceServer\xml\sar
- ▶ AIX /usr/WebSphere/CommerceServer/xml/sar
- ▶ Solaris ▶ Linux /opt/WebSphere/CommerceServer/xml/sar
- ▶ 400 /qibm/proddata/WebCommerce/xml/sar

상점-카탈로그 관계를 작성하기 전에 상점 데이터 자원을 작성했는지 확인하십시오. 개별 태스크가 storecatalog.xml 파일에서 항목들을 작성하는 다음 태스크를 완료하십시오.

1. 상점에 카탈로그 그룹(카테고리)을 표시하려면 카탈로그 그룹에 JSP 템플리트를 지정해야 합니다. 특정 표시 페이지 템플리트를 카탈로그 그룹이나 기본 템플릿에 지정하여 모든 카탈로그 그룹을 표시할 수 있습니다. 다음 ToolTech 견본 상점 예를 참조하여 DISPCGPREL 테이블에 정보를 추가하여 카탈로그 그룹 템플리트를 지정해야 합니다. 카탈로그 그룹에 지정하고자 하는 각 템플릿에 대한 이 태스크를 완료하십시오.

```
<dispcgprel
catgroup_id="@catgroup_id_1"
devicefmt_id="-1"
dispcgprel_id="@dispcgprel_id_1"
mbrgrp_id="0"
pagename="/ToolTech/CategoryDisplay.jsp"
storeent_id="@storeent_id_1"
rank="0"/>
```

여기서,

- catgroup_id는 이 페이지 이름이 표시될 카탈로그 그룹의 참조 번호입니다. 0 값은 이 페이지가 모든 카탈로그 그룹에 대해 사용될 것임을 나타냅니다.
- devicefmt_id는 이 페이지가 표시될 신용 카드 유형의 참조 번호입니다. -1 값은 이 템플릿 페이지가 HTTP 브라우저에 의해 사용될 것임을 나타냅니다.
- dispcgprel_id는 이 항목의 참조 번호입니다.
- mbrgrp_id는 이 템플릿 페이지가 표시될 구성원 그룹의 참조 번호입니다. 0 값은 이 템플릿 페이지가 모든 구성원 그룹에 대해 사용될 것임을 나타냅니다.
- pagename은 표시 템플릿 페이지의 이름과 경로입니다.

- rank는 하나 이상의 페이지가 선택 기준을 충족시킬 때 연결을 중단시키기 위해 사용되는 순서 번호입니다.

주: JSP 템플리트를 카탈로그 그룹에 지정할 때마다 catentry_id가 기존의 카탈로그 항목과 일치시키기 위해 순서를 변경합니다.

2. 상품, 항목, 패키지, 번들 및 동적 제품군 등의 카탈로그 항목들을 상점에 표시하려면 카탈로그 항목에 JSP 템플리트를 지정해야 합니다. 기본 템플리트를 지정하여 모든 카탈로그 항목을 표시하거나 기본값을 지정하여 각 유형의 카탈로그 항목을 표시할 수 있습니다. 예를 들어, 상품에 대해 하나의 템플리트를 지정하고 항목에 다른 템플리트를 지정하거나, 특정 카탈로그 항목에 특정 템플리트를 지정하는 것입니다. 다음 ToolTech 견본 상점 예를 참조하여 DISPENTREL 테이블에 정보를 추가하여 템플리트를 지정해야 합니다. 카탈로그 그룹에 지정하고자 하는 각 템플리트에 대한 이 태스크를 완료하십시오.

```
<dispentrel
auctionstate="0"
catentry_id="0"
catenttype_id="ProductBean"
devicefmt_id="-1"
dispentrel_id="@dispentrel_id_1"
mbrgrp="0"
pagename="/ToolTech/ProductDisplay.jsp"
storeent_id="@storeent_id_1"
rank="0"/>
```

여기서

- auctionstate는 이 템플리트 페이지가 경매 중인 카탈로그 항목을 표시함을 나타냅니다.
 - 0=경매 템플리트가 아님
 - 1 = 경매 템플리트
- catentry_id는 이 페이지 이름이 표시될 카탈로그 항목의 참조 번호입니다. 0 값은 이 페이지 이름이 모든 카탈로그 항목에 대해 사용될 것임을 나타냅니다.
- catenttype_id는 이 페이지가 표시하기 위해 사용될 카탈로그 항목의 유형입니다.
 - ProductBean=상품 표시
 - ItemBean=항목 표시
 - PackageBean=패키지 표시
 - BundleBean=번들 표시
 - DynamicKitBean=동적 제품군 표시
- devicefmt_id는 이 페이지가 표시될 신용 카드 유형의 참조 번호입니다. -1 값은 이 템플리트 페이지가 HTTP 브라우저에 의해 사용될 것임을 나타냅니다.
- dispentrel_id는 카탈로그 항목의 참조 번호입니다.

- mbrgrp는 이 템플릿 페이지가 표시될 구성원 그룹의 참조 번호입니다. 0 값은 이 템플릿 페이지가 모든 구성원 그룹에 대해 사용될 것임을 나타냅니다.
- pagename은 표시 템플릿 페이지의 이름과 경로입니다.
- storeent_id는 이 페이지가 표시될 상점의 참조 번호입니다.
- rank는 하나 이상의 페이지가 선택 기준을 충족시킬 때 연결을 중단시키기 위해 사용되는 순서 번호입니다.

주: JSP 템플릿을 카탈로그 항목에 지정할 때마다 catentry_id가 기존의 카탈로그 항목과 일치시키기 위해 순서를 변경합니다.

탐색 가능 카탈로그

WebSphere Commerce 상점에는 두 가지 유형의 카탈로그(마스터 및 탐색 가능 카탈로그)가 허용됩니다. 탐색 가능 카탈로그는 마스터 카탈로그에 있는 구조적 제한 사항이 없습니다. 이 카탈로그는 유동적인 표시 구조를 제공하여 사용자의 상점 요구사항에 적합한 탐색 가능 카탈로그를 작성하도록 허용합니다.

특히 탐색 가능 카탈로그는 마스터 카탈로그에 있는 다음과 같은 제한 사항을 충족시킬 필요없습니다.

- 마스터 카탈로그는 적절한 트리에 위치해야 합니다. 이는 주기가 없으며 다음 구조를 사용할 수 없음을 의미합니다. 상위 카테고리 A에는 하위 카테고리 B가 있습니다. B와 B의 모든 서브카테고리는 A의 상위 카테고리가 아니라는 점이 중요합니다.
- 하나의 상품은 두 개 이상의 카탈로그에 속할 수 없습니다.







다음 태스크는 NewFashion 견본 상점 카탈로그를 수정하여 탐색 가능 카탈로그를 작성합니다. 결과 카탈로그는 다음 단계가 복수 카테고리 카테고리 주기 및 일부 제품의 카테고리화를 소개하므로 더 이상 마스터 카테고리로서 분류되지 않을 수 있습니다. 일반 탐색 가능 카탈로그는 정보를 카테고리 관계 테이블, CATGRPREL과 CATGPENREL에 추가함으로써 작성됩니다. 여기서, CATGRPREL에는 서브 카테고리 관계가 있고 CATGPENREL에는 각 카탈로그에 대한 카테고리 상품 관계가 있습니다. 이 예제는 NewFashion과 관련된 것이지만, 적절히 조정하여 카탈로그 정보, 구조 및 설계를 일치시킴으로써 사용자 고유의 카탈로그에서 이들 기본 단계를 수행할 수 있습니다.

카탈로그 주기 작성

카탈로그는 4개의 상위 카테고리가 있습니다(남성 의류, 여성 의류, 신상품 및 홈페이지 특별 판매). 이 예제에서는 신상품 카테고리를 남성의류로 복사하고 홈페이지 특별 판매를 여성의류로 복사한 다음 이름을 신상품으로 변경하는 방법을 보여줍니다.

카테고리 주기를 사용하여 NewFashion 견본 상점 마스터 카탈로그를 탐색 가능 카탈로그로 변경하려면, 다음을 수행하십시오.

1. NewFashion 상점 아카이브를 공개하여 NewFashion 견본 상점을 작성하십시오. NewFashion은 미국 영어 및 WebSphere Commerce로 출시되는 9개 자국어 중 하나로 사용 가능합니다. NewFashion_en_US_locale.sar 파일 중 공개할 파일을 하나 선택하십시오.
2. 편집기에서 catalog.xml 파일을 여십시오. 파일은 다음 WebSphere Commerce 디렉토리에 있습니다.

-  `drive:\WebSphere\CommerceServer\samplstores\NewFashion\locale\data`
-  `drive:\Program Files\WebSphere\CommerceServer\samplstores\NewFashion\locale\data`
-  `/usr/WebSphere/CommerceServer/samplstores/NewFashion/locale/data`
-   `/opt/WebSphere/Commerce/samplstores/NewFashion/locale/data`
-  `/qibm/proddata/WebCommerce/samplstores/NewFashion/locale/data`

3. catalog.xml 파일에서 CATGRPREL 데이터 절을 찾을 수 있습니다. 남성 의류 및 신상품 사이에 새 최상위 카테고리 관계를 작성하십시오. 현재, 두 개의 카테고리가 맨 위 레벨에 있습니다. 탐색 가능한 관계에 대하여, 원래 구조를 보존하면서 신상품을 남성 의류의 서브카테고리로 위치시키도록 새 절을 작성하십시오. CATGRPREL 절 아래, 다음 추출을 추가하십시오.

```
<catgrprel
catgroup_id_parent="@catgroup_id_11"
catgroup_id_child="@catgroup_id_21"
catalog_id="@catalog_id_1"
sequence="7"
/>
```

여기서,

- `catgroup_id_parent`는 NewFashion 견본 상점에 의해 정의되는 상위 카테고리의 카탈로그 그룹 내부 참조 번호입니다. 이 예제에서 `@catgroup_id_11`은 남성 의류 카테고리입니다.
- `catgroup_id_child`는 NewFashion 견본 상점에 의해 정의되는 하위 카테고리의 카탈로그 그룹 내부 참조 번호입니다. 이 예제에서 `@catgroup_id_21`은 신상품 카테고리입니다.
- `catalog_id`는 NewFashion 견본 상점에 의해 정의되는 카탈로그의 내부 참조 번호입니다.

- sequence는 NewFashion 견본 상점에 의해 정의되는 카탈로그 그룹의 콘텐츠의 표시 순서를 판별하는 번호입니다. 이 예제에서 처음 여섯 개의 남성의류 서브카테고리 뒤에 신상품 카테고리가 마지막으로 표시됩니다.
4. 위 단계를 반복하여 여성의류 및 홈페이지 특별 판매 사이에 새 최상위 카테고리 관계를 작성하십시오. 현재 두 개의 카테고리가 맨 위 레벨에 있습니다. 탐색 가능한 관계에 대하여, 원래 구조를 보존하면서 홈페이지 특별 판매를 여성의류의 서브카테고리로 위치시키도록 새 절을 작성하십시오. CATGRPREL 절 아래, 다음 추출을 추가하십시오.

```
<catgrprel
catgroup_id_parent="@catgroup_id_20"
catgroup_id_child="@catgroup_id_22"
catalog_id="@catalog_id_1"
sequence="9"
/>
```

여기서,

- catgroup_id_parent는 NewFashion 견본 상점에 의해 정의되는 상위 카테고리의 카탈로그 그룹 내부 참조 번호입니다. 이 예제에서 @catgroup_id_20은 여성의류 카테고리입니다.
 - catgroup_id_child는 NewFashion 견본 상점에 의해 정의되는 하위 카테고리의 카탈로그 그룹 내부 참조 번호입니다. 이 예제에서 @catgroup_id_22는 홈페이지 특별 판매 카테고리입니다.
 - catalog_id는 NewFashion 견본 상점에 의해 정의되는 카탈로그의 내부 참조 번호입니다.
 - sequence는 NewFashion 견본 상점에 의해 정의되는 카탈로그 그룹의 콘텐츠의 표시 순서를 판별하는 번호입니다. 이 예제에서 홈페이지 특별 판매 카테고리는 마지막으로 표시됩니다.
5. 현재 홈페이지 특별 판매가 여성의류의 서브카테고리로 되어, 카테고리 이름이 틀립니다. 카테고리 이름을 신상품으로 이름을 바꿔 남성의류의 새 서브카테고리와 일치시키십시오.
6. catalog.xml 파일을 저장하십시오.
7. 변경사항을 보려면, 다음 중 하나를 실행하십시오. 상점 서비스를 통해 수정된 NewFashion 상점 아카이브를 공개하거나 catalog.xml 파일을 316 페이지의 『데이터베이스 자원 그룹 로드』 지시사항을 사용하여 로더 패키지로 로드하십시오.



2차 카테고리에 상품을 추가

이 예제는 원래 구조를 보존하면서 하나의 카테고리에서 또 다른 카테고리로 상품을 복사하는 방법을 보여줍니다. 홈페이지 특별 판매 카테고리는 여름 잠옷 상품을 포함합니다.

다. 또한, 여름 잠옷 상품은 여성의류 맨 위 카테고리에 대한 잠옷 서브카테고리 밑에 속합니다. 이 지시사항은 여름 잠옷 상품과 SKU를 잠옷 카테고리로 복사하는 방법을 보여줍니다.

2차 카테고리에 상품을 추가하여 NewFashion 견본 상점 마스터 카탈로그를 탐색 가능 카탈로그를 변경하려면, 다음을 수행하십시오.

1. NewFashion 상점 아카이브를 공개하여 NewFashion 견본 상점을 작성하십시오. NewFashion은 미국 영어 및 WebSphere Commerce로 출시되는 9개 자국어 중 하나로 사용 가능합니다. NewFashion_en_US_locale.sar 파일 중 공개할 파일을 하나 선택하십시오.
2. 편집기에서 catalog.xml 파일을 여십시오. 파일은 다음 WebSphere Commerce 디렉토리에 있습니다.

-  `drive:\WebSphere\CommerceServer\samplstores\NewFashion\locale\data`
-  `drive:\Program Files\WebSphere\CommerceServer\samplstores\NewFashion\locale\data`
-  `/usr/WebSphere/CommerceServer/samplstores/NewFashion/locale/data`
-   `/opt/WebSphere/CommerceServer/samplstores/NewFashion/locale/data`
-  `/qibm/proddata/WebCommerce/samplstores/NewFashion/locale/data`

3. catalog.xml 파일에서 CATGPENREL 데이터 절을 찾으십시오. 원래 홈페이지 특별 상품 카테고리 밑에 있었던 여름 잠옷에 대해 새 상품 항목을 작성하십시오. CATGPENREL 절 아래, 다음 추출을 추가하여 상품을 포함하십시오.

```
<catgpenrel
  catgroup_id="@catgroup_id_18"
  catalog_id="@catalog_id_1"
  catentry_id="@product_id_2692"
  sequence="2"
/>
```

여기서,

- catgroup_id는 NewFashion 견본 상점에 의해 정의되는 카탈로그 그룹 내부 참조 번호입니다. 이 예제에서 @catgroup_id_18은 잠옷 카테고리입니다.
- catalog_id는 NewFashion 견본 상점에 의해 정의되는 카탈로그의 내부 참조 번호입니다.

- catentry_id는 NewFashion 견본 상점에 의해 정의되는 카탈로그 항목 내부 참조 번호입니다. 이 예제에서 @catentry_id_2692는 여름 잠옷 상품입니다.
 - sequence는 NewFashion 견본 상점에 의해 정의되는 카탈로그 그룹의 콘텐츠의 표시 순서를 판별하는 번호입니다. 이 예제에서 여름 잠옷 상품은 마지막으로 표시됩니다.
4. 여름 잠옷 상품 항목을 추가한 후, NewFashion 견본 상점에서 정의되는 CATGPENREL 아래의 상품에 대한 SKU 항목을 추가하십시오. 이제 여름 잠옷 상품이 10개의 정의된 SKU를 포함합니다. CATGPENREL 절 아래, 다음 추출을 추가하여 SKU를 포함하십시오.

```
<catgpenrel
catgroup_id="@catgroup_id_18"
catalog_id="@catalog_id_1"
catentry_id="@catentry_id_2695"
sequence="2"
/>
```

```
<catgpenrel
catgroup_id="@catgroup_id_18"
catalog_id="@catalog_id_1"
catentry_id="@catentry_id_2696"
sequence="2"
/>
```

```
<catgpenrel
catgroup_id="@catgroup_id_18"
catalog_id="@catalog_id_1"
catentry_id="@catentry_id_2697"
sequence="2"
/>
```

```
<catgpenrel
catgroup_id="@catgroup_id_18"
catalog_id="@catalog_id_1"
catentry_id="@catentry_id_2698"
sequence="2"
/>
```

```
<catgpenrel
catgroup_id="@catgroup_id_18"
catalog_id="@catalog_id_1"
catentry_id="@catentry_id_2699"
sequence="2"
/>
```

```
<catgpenrel
catgroup_id="@catgroup_id_18"
catalog_id="@catalog_id_1"
catentry_id="@catentry_id_2700"
sequence="2"
/>
```

```
<catgpenrel
catgroup_id="@catgroup_id_18"
```

```
catalog_id="@catalog_id_1"  
catentry_id="@catentry_id_2701"  
sequence="2"  
</>
```

```
<catgpenrel  
catgroup_id="@catgroup_id_18"  
catalog_id="@catalog_id_1"  
catentry_id="@catentry_id_2702"  
sequence="2"  
</>
```

```
<catgpenrel  
catgroup_id="@catgroup_id_18"  
catalog_id="@catalog_id_1"  
catentry_id="@catentry_id_2703"  
sequence="2"  
</>
```

```
<catgpenrel  
catgroup_id="@catgroup_id_18"  
catalog_id="@catalog_id_1"  
catentry_id="@catentry_id_2704"  
sequence="2"  
</>
```

여기서

- catgroup_id는 NewFashion 견본 상점에 의해 정의되는 카탈로그 그룹 내부 참조 번호입니다. 이 예제에서 @catgroup_id_18은 잠옷 카테고리입니다.
- catalog_id는 NewFashion 견본 상점에 의해 정의되는 카탈로그의 내부 참조 번호입니다.
- catentry_id는 NewFashion 견본 상점에 의해 정의되는 카탈로그 항목 내부 참조 번호입니다. 이 예제에서 @catentry_id_2695에서 @catentry_id_2704까지는 여름 잠옷 상품에 대해 정의된 SKU를 나타냅니다.
- sequence는 NewFashion 견본 상점에 의해 정의되는 카탈로그 그룹의 콘텐츠의 표시 순서를 판별하는 번호입니다. 이 예제에서 여름 잠옷 SKU는 마지막으로 표시됩니다.







5. catalog.xml 파일을 저장하십시오.

6. 변경사항을 보려면, 다음 중 하나를 실행하십시오. 상점 서비스를 통해 수정된 NewFashion 상점 아카이브를 공개하거나 catalog.xml 파일을 316 페이지의 『데이터베이스 자원 그룹 로드』 지시사항을 사용하여 로더 패키지로 로드하십시오.

WebSphere Commerce에서 카탈로그 자원 관리

시간이 지나면서 마스터 카탈로그로부터의 데이터베이스 자원 정보를 갱신할 필요가 있을 것입니다. 카탈로그를 유지보수하는 것은 지속적으로 상품을 추가 또는 제거하고, 카테고리 또는 카테고리 그룹을 작성하거나 연관시키며, 설명이나 가격같은 상품 정보를 갱신하는 것처럼 연속적인 과정입니다.

기존 데이터베이스 항목 및 사용자의 상점의 catalog.xml 파일을 사용하여 WebSphere Commerce XML 데이터를 편집함으로써 사용자 카탈로그 자원을 변경할 수 있습니다. 다음 데이터 디렉토리에 있는 WebSphere Commerce 기본 상점 XML 파일을 참조로 사용하십시오.

-  NT `drive:\WebSphere\CommerceServer\samplstores`
-  2000 `drive:\Program Files\WebSphere\CommerceServer\samplstores`
-  AIX `/usr/WebSphere/CommerceServer/samplstores`
-  Solaris  Linux `/opt/WebSphere/CommerceServer/samplstores`
-  400 `/qibm/proddata/WebCommerce/samplstores`

주: 이 예제는 NewFashion 기본 상점에서 작성되었으며, 어떤 XML 요소가 카탈로그 자원 정보를 변경하기 위해 수정되어야 하는지 식별합니다.

카탈로그 그룹

카탈로그 그룹이 CATGROUP 및 CATGRPDESC 데이터베이스 테이블을 사용하여 WebSphere Commerce 카탈로그에 작성됩니다. catalog.xml 파일에서 일반 카탈로그 그룹은 다음 추출과 비슷합니다.

```
<catgroup
  catgroup_id="@catgroup_id_1"
  member_id("&MEMBER_ID"
  identifier="Accessories"
  markfordelete="0"
/>
```

catgroup_id는 카탈로그 그룹의 내부 참조 번호입니다. 각 카탈로그 그룹은 WebSphere Commerce에 내부 참조 번호가 지정되며 이것은 카탈로그 항목 추가시 그룹을 식별합니다. identifier는 카탈로그 그룹에 대한 외부 이름입니다. 두 요소 모두 데이터베이스 자원 내에 고유하며 복제 작성될 수 없습니다.

이름 및 설명이 로케일 특정 catalog.xml 파일에 속하며, 이 중 하나가 사용자 상점을 지원하는 각 로케일에 대해 필요합니다. 변환 가능한 정보를 포함하는 일반 카탈로그 그룹은 다음 추출과 비슷합니다.

```
<catgrpdesc
  language_id("&en_US"
  catgroup_id="@catgroup_id_1"
```

```

name="Accessories"
shortdescription="Accessories"
longdescription="Accessories"
published="1"
/>

```

language_id는 카탈로그 정보의 언어를 식별합니다. 이 식별자는 사용자 상점을 지원하는 각 언어와 일치하기 위해 변경되어야 합니다. name은 고객에게 카탈로그 그룹의 간단한 설명 및 자세한 설명이 있는 shortdescription 및 longdescription 요소로서 표시됩니다.

새 카탈로그 그룹 작성시, 해당 정보를 위해 위의 구조를 따르십시오.

주:

1. identifier 및 name 요소가 위의 예와 동일하지만 내용은 다양할 수 있습니다. 예를 들어, 보완관으로 사용자 카탈로그 그룹 이름을 바꾸려고 선택할 수 있습니다. 이러한 경우, identifier의 정보는 변경할 필요가 없으며, name만 변경하면 됩니다.
2. 카탈로그 그룹 삭제시, catgroup_id 발생이 그에 따라 갱신되는지 확인하십시오. 예를 들어, 카탈로그 그룹 아래의 카탈로그 항목도 삭제하려는 경우, 전체 XML 항목을 삭제하게 됩니다. 그러나 카탈로그 항목을 유지하려는 경우, 올바른 그룹으로 catgroup_id를 변경해야 합니다.

카탈로그 항목

카탈로그 항목이 CATENTRY 및 CATENTDESC 데이터베이스 테이블에서 정보를 사용하여 WebSphere Commerce 카탈로그에 작성됩니다. 카탈로그 항목은 상품, 항목, 패키지, 번들 또는 동적 제품군이 될 수 있습니다. catalog.xml 파일에서 일반 카탈로그 항목은 다음 추출과 비슷합니다.

```

<catentry
catentry_id="@product_id_102"
baseitem_id="@baseitem_id_102"
member_id="&MEMBER_ID"
catenttype_id="ProductBean"
partnumber="product-sku-nf-102"
mfpartnumber="product-sku-nf-102"
mfname="NewFashion"
markfordelete="0"
buyable="1"/>

```

catentry_id는 상품 카탈로그 항목의 내부 참조 번호입니다. baseitem_id는 재고 목적을 위해 카탈로그 항목이 관련된 기본 항목입니다. partnumber는 카탈로그 항목의 부품 번호를 식별하는 참조 번호입니다. mfpartnumber는 카탈로그 항목을 식별하기 위해 제조업체가 사용하는 부품 번호입니다. 이 요소들은 데이터베이스 자원 내에서 고유해야 하며 복제 작성될 수 없습니다.

catenttype_id는 ItemBean, ProductBean, PackageBean, BundleBean 또는 DynamicKitBean과 같은 카탈로그 항목의 유형을 식별합니다.

이름 및 설명이 로케일 특정 catalog.xml 파일에 속하며, 이 중 하나가 사용자 상점을 지원하는 각 로케일에 대해 필요합니다. 또한, 상품 이미지가 이 파일에 포함되어 있습니다. 변환 가능한 정보가 있는 일반 카탈로그 그룹은 다음 추출과 비슷합니다.

```
<catentdesc
catentry_id="@product_id_102"
language_id("&en_US"
name="Belt"
shortdescription="Classic belt"
longdescription="This classic belt looks great with your favorite jeans,
or takes you to work in style. 1 1/2 inches wide in full-grain leather
with a solid nickel buckle."
thumbnail="images/mens_accessories_belt_sm.gif"
fullimage="images/mens_accessories_belt.gif"
available="1"
published="1"
/>
```

language_id는 카탈로그 정보의 언어를 식별합니다. 이 식별자는 사용자 상점을 지원하는 각 언어와 일치하기 위해 변경되어야 합니다. name은 고객에게 카탈로그 항목의 간단한 설명 및 자세한 설명을 포함하는 shortdescription 및 longdescription 요소로서 표시됩니다.

새 카탈로그 항목 작성시, 해당 정보를 위해 위의 구조를 따르십시오.

주:

1. 카탈로그 항목 삭제시, 고유 요소의 각 발생이 이에 맞게 갱신되는지 확인하십시오. 예를 들어, 카탈로그 그룹 아래의 카탈로그 항목도 삭제하려는 경우, 전체 XML 항목을 삭제하게 됩니다. 그러나 카탈로그 항목을 유지하려는 경우, 올바른 그룹으로 catgroup_id를 변경해야 합니다.
2. 상품은 카탈로그 항목의 다른 유형 이전에 작성되어야 합니다.

XML 파일을 수동으로 변경하려는 경우, 상품 관리 도구 또는 Catalog Manager 유틸리티 방법 중 하나를 선택하십시오.

상품 관리 도구

WebSphere Commerce 액셀러레이터에 있는 상품 관리 도구를 사용하면, 다양한 마법사 및 노트북을 사용하여 상점의 마스터 카탈로그에 있는 상품을 관리할 수 있습니다. 카탈로그의 콘텐츠를 갱신하거나 새로운 카탈로그 데이터를 작성할 수 있습니다.

- 마법사 또는 노트북을 사용하여 상품과 상품 정보를 작성, 갱신 및 삭제하십시오. 상품은 최종적으로 고객에 팔릴 개별 항목인 SKU에 대한 템플릿이 됩니다. 상품 정보에는 상품 코드(상품을 단일하게 식별함), 상품 이름 및 설명, 모든 판매 계획 옵션(예: 고객에게 상품을 보여주거나 상품이 특별 판매 대상인지를 알려주는 것), 상품 이미지, 세금 및 운송 스펙, 상품에 지정된 할인 및 제조업체 정보 등이 포함됩니다.

- 구매에 대한 SKU(또는 항목)를 생성, 갱신 및 삭제하십시오. SKU는 판매 계획에 있는 주문 가능한 항목을 표시합니다. 특정 상품과 연관된 모든 SKU들은 동일한 속성 세트를 보이며, 속성값에 의해 구별됩니다. SKU에 대한 추가 및 변경사항에는 주문 가능한 기초가 아니면, 상품과 동일한 정보가 포함됩니다.
- 상점에 의해 제공되는 상품 또는 서비스를 조직하는 데 사용되는 비슷한 특성을 가진 오브젝트의 그룹인 카테고리(또는 카탈로그 그룹)를 작성, 갱신 및 삭제하십시오. 상위 카테고리 및 이미지를 포함하는 카테고리 코드, 이름, 설명 등과 같은 카테고리 및 카테고리 정보를 작성, 변경 및 삭제하여 마스터 카탈로그의 카테고리 계층을 관리할 수 있습니다.
- 상위 카테고리를 선택하거나 상품과 SKU를 하나의 카테고리에서 다른 카테고리로 이동시켜 상품과 SKU를 연관시키십시오.
- 상품에 대한 속성 및 속성값을 작성하십시오. 가능한 각각의 속성 및 속성값 조합은 새로운 SKU와 같습니다. 이런 특성을 SKU에 지정하기 전에 속성값을 미리 정의해야 합니다. 속성 및 속성값을 작성한 후, 이름, 설명(텍스트, 자연수 또는 10진수)과 속성 및 속성값이 나타나는 순서 등과 같은 정보를 작성 또는 갱신할 수 있습니다.
- 카탈로그 가격 책정을 작성, 갱신 또는 삭제하고 상품과 연관시키십시오. 단일 또는 다량에 대한 가격을 설정하는 것처럼 가격을 사용함에 있어 꼭 충족되어야 하는 조건 세트와 함께 하나 이상의 통화로 상품 또는 SKU의 가격을 정의할 수 있습니다.

각 태스크에 대한 자세한 지시사항은 온라인 도움말의 상품 관리 절을 참조하십시오.

주:

1. 상품 관리자 도구는 변경사항이 적을 경우에만 사용이 권장됩니다. 계절 상품을 추가 또는 제거하거나, 재고 일소 판매를 준비하는 것과 같은 큰 카탈로그 갱신에 대하여는 로더 패키지를 사용하십시오.
2. 사용자가 캐시를 사용 가능하지 않게 하거나 현재 캐시된 JSP 페이지를 제거하지 않으면, 카탈로그 데이터의 어떤 변경 사항도 상점에서 표시되지 않습니다. 자세한 정보는 WebSphere Commerce 온라인 도움말의 CacheDelete 명령을 참조하십시오. CacheDelete 명령은 동적 페이지 캐시의 원격 정리를 시작하며 파일 시스템으로 직접 액세스하지 않고도 캐시를 관리할 수 있게 해줍니다. 이 명령을 사용하기 전에 자동 페이지 무효화를 사용할 수 있는지 확인하십시오. 이 명령을 사용하려면 사이트 관리자 또는 상점 개발자로 로그인해야 합니다.

Catalog Manager

Catalog Manager에서 사용할 수 있는 두 개의 유틸리티인 로더 패키지 또는 웹 편집기를 사용하여 카탈로그를 유지보수할 수도 있습니다. 로더 패키지는 많은 양의 기존 상품 정보를 데이터베이스로 반입하는 데 적합합니다. 로더 패키지는 WebSphere Commerce에서 카탈로그 정보를 작성 및 관리하는 기본 도구입니다. 이 패키지는 주로 데이터를 WebSphere Commerce 데이터베이스로 로드하고 준비하기 위한 명령 유틸리티들로 구성되어 있습니다. 로더 패키지를 사용하여 데이터베이스로부터 데이터를 XML

문서 형태로 추출하고, XML 데이터를 대체 XML 포맷으로 변환하고, 문자 분리 변수 포맷과 XML 데이터 포맷 사이에서 데이터를 변환할 수 있습니다.

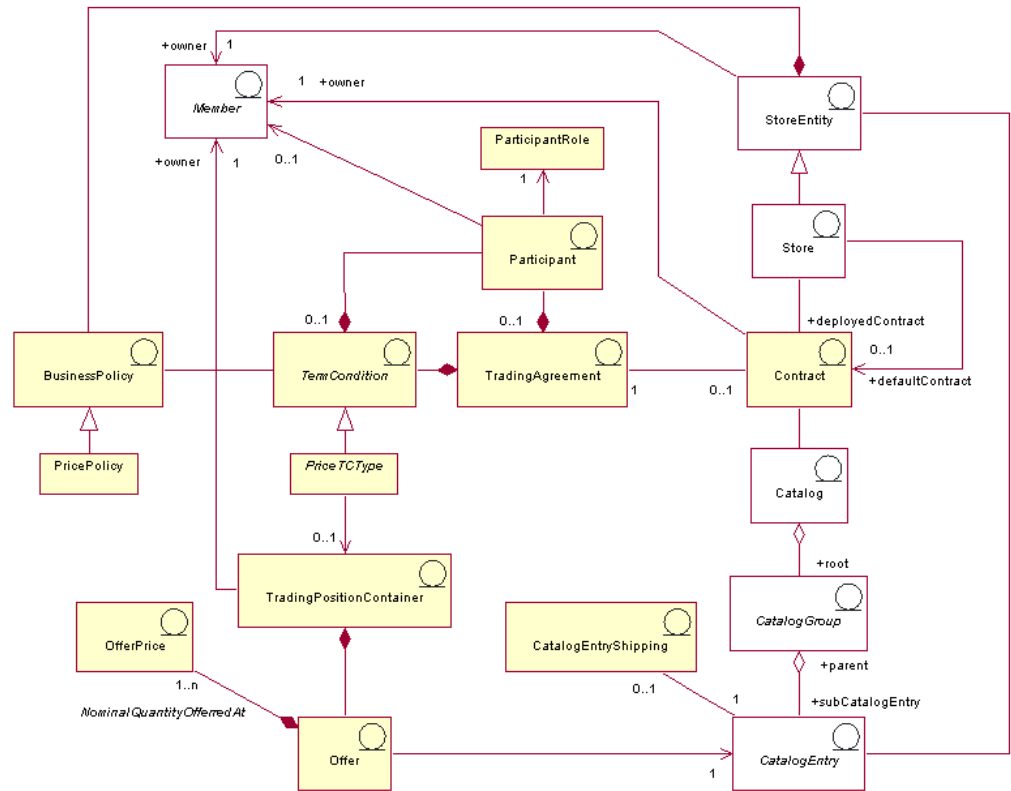
웹 편집기는 카탈로그 데이터를 작성, 편집 또는 삭제할 수 있는 웹 브라우저 인터페이스를 사용합니다. 정보 보기 또는 갱신에 대한 데이터 입력 양식이 웹 편집기의 핵심입니다. 가장 간단한 경우, 양식은 WebSphere Commerce 데이터베이스의 테이블에 해당합니다. 운영자는 제공된 기본 양식을 사용하거나 사용 가능한 양식을 사용자 정의하도록 선택할 수 있습니다. 자세한 정보는 *IBM WebSphere Commerce 버전 5.4 Catalog Manager 사용자 안내서*를 참조하십시오.

제 9 장 가격 책정 자원

가격 책정은 해당 가격을 사용하기 위해 만족해야 하는 기준 및 카탈로그 항목의 가격을 나타냅니다. 기능적인 카탈로그를 작성하려면 데이터베이스에 가격 책정 정보를 추가해야 합니다. 로더 패키지를 사용하여 데이터베이스에 로드할 수 있는 XML 파일 양식으로 가격 정보를 작성할 수 있습니다. 또는 가격 책정 데이터의 소 총계에 대해서는 WebSphere Commerce 액셀러레이터에서 상품 관리 도구를 사용할 수 있습니다.

WebSphere Commerce의 가격 책정

다음 도표는 WebSphere Commerce 서버의 가격 책정 자원을 보여줍니다.



이 도표와 상점 데이터 질의 모든 도표는 WebSphere Commerce 서버 정보 모델의 부분입니다. 정보 모델에 대한 자세한 정보는 29 페이지의 『상점 데이터 정보 모델』을 참조하십시오. 이 도표에서 사용된 규칙에 대한 자세한 정보는 359 페이지의 부록 A 『UML 범례』를 참조하십시오.

판매 설정

판매 설정 또는 가격 책정은 동일한 상품 또는 항목에 대해 다른 고객 또는 조직에 다른 가격을 제시하는 것입니다. 판매 설정은 고객이 해당 가격을 지불하는 데 만족하도록 구입 수량과 같은 카탈로그 항목 및 기준을 제시합니다. 예를 들어, 판매 계획 또는 서비스는 종종 어린이, 학생, 성인 및 노인에게 대해 다른 가격을 책정합니다. WebSphere Commerce에서 판매 설정은 거래 포지션이라고 하며 거래 포지션 컨테이너의 파트입니다.

판매 가격

판매 가격은 거래 계약이나 장기 구매 계약에 의해 상점에서 카탈로그 항목마다 제시한 가격입니다. 판매 설정은 복수 통화로 정의된 하나 이상의 판매 가격을 가질 수 있습니다.

거래 포지션 컨테이너

판매 설정은 구성원이 소유하고 있는 거래 포지션 컨테이너의 파트입니다. 거래 포지션 컨테이너에는 거래 포지션이 포함됩니다. 이것은 모든 고객 또는 거래 계약이나 장기 구매 계약, 장기 구매 계약의 규정을 통해 일정한 그룹의 고객들만 사용할 수 있습니다. 장기 구매 계약에서, 거래 포지션 컨테이너는 복수 가격 비즈니스 정책으로 참조되고 상점 그룹에 있는 상점이나 모든 상점에서 공유될 수 있는 가격 비즈니스 오브젝트입니다. 거래 포지션 컨테이너는 가격 목록이라고도 합니다.

규정

규정은 거래 계약의 행위 및 특성을 정의합니다. 상점 운영의 많은 면이 비즈니스 정책에 의해 정의되기 때문에 많은 규정이 비즈니스 정책을 참조합니다.

가격 책정 규정의 유형

가격 책정 규정은 장기 구매 계약 아래에서 어떤 상품이 사용 가능한지 정의하며 고객이 상품에 지불할 가격을 정의합니다. 다음 가격 책정 용어 중 최소한 하나가 장기 구매 계약에서 필요합니다.

- 사용자 정의 가격 목록은 판매용 상품과 그 가격 목록이 장기 구매 계약에서 판매용으로 사용자 정의되며 가격이 사용자 정의됨을 지정합니다. 항목은 상점 카탈로그의 어디에나 있을 수 있는 상점 카탈로그의 섹션으로 제한되지 않습니다.
- 조정 규정이 있는 전체 카탈로그는 상점 카탈로그에 정의된 대로 기본 가격에서 백분율 조정(가격 인상이나 할인)된 판매에 대해 상점 카탈로그에서 사용 가능한 모든 상품을 판매 설정합니다. 조정이 지정되지 않은 경우, 항목은 기본 가격으로 판매됩니다.
- 조정 규정이 있는 가격 목록은 상점 카탈로그에 정의된 대로 기본 가격에서 백분율 조정(가격 인상이나 할인)된 판매에 대해 가격 목록에서 사용 가능한 모든 상품을 판매 설정합니다. 조정이 지정되지 않은 경우, 항목은 기본 가격으로 판매됩니다.

- 선택적 조정 규정이 있는 가격 목록은 조정이 전체 가격 목록에 적용되는 것만 제외하고 조정된 가격 목록과 유사합니다. 가격 목록의 서브세트에 대한 조정이 작성됩니다. 가격 목록의 서브세트는 상품 세트 비즈니스 정책이나 사용자 정의된 상품 세트 중 하나일 수 있습니다.

거래 계약

거래 계약은 장기 구매 계약, RFQ, 비즈니스 계정 또는 경매일 수 있습니다. 거래 계약은 장기 구매 계약에서 규정된 비즈니스 정책과 지정된 규정으로 특정 항목을 구입할 수 있는 판매자와 구매자 간에 협상된 계약입니다. 예를 들어, 고객이 가격 책정 규정 아래에서 지정된 기간에 지정된 가격으로 상점에서 상품을 구입하게 합니다. WebSphere Commerce에서, 모든 고객은 장기 구매 계약 아래 상점에서 구매해야 하며, 상점은 하나 이상의 장기 구매 계약을 전개하고 그 중 하나는 기본 장기 구매 계약으로 지정될 수 있습니다. 기본 장기 구매 계약에는 상점 기본 정책 세트와 연관된 규정 세트가 들어 있습니다. 거래 계약에는 다른 역할을 갖는 0명 이상의 참석자가 있을 수 있습니다.

참석자

참석자는 거래 계약이나 규정 중 하나의 파트일 수 있습니다. 참석자는 구성원 그룹, 조직 등일 수 있는 구성원입니다. 구매자 역할의 참석자가 장기 구매 계약을 위해 지정된 경우, 구매자는 장기 구매 계약 아래에서 구매하기 위한 구매자 참석자의 구성원이어야 합니다. 장기 구매 계약의 규정은 0 또는 복수의 참석자를 포함할 수 있습니다.

참석자 역할

참석자는 다음과 같은 참석자 역할 중 하나를 가질 수 있습니다.

- 작성자
- 판매자
- 구매자
- 공급자
- 승인자
- 계정 보유자
- 구매자 연락처
- 판매자 연락처
- 대리인
- 운영자

장기 구매 계약

장기 구매 계약에는 상품에 대한 판매 가격이 들어 있습니다. WebSphere Commerce에서 모든 고객은 장기 구매 계약 아래에서 구매해야 합니다. 장기 구매 계약을 체결하면 고객은 장기 구매 계약에 명시된 규정 및 비즈니스 정책에 따라 지정된 기간 동안

지정된 가격으로 상점에서 상품을 구매할 수 있습니다. 상점은 0개 이상의 장기 구매 계약을 소유하고 있으며, 최소한 하나의 기본 장기 구매 계약을 소유하고 있습니다.

비즈니스 정책

비즈니스 정책은 비즈니스 프로세스, 업종 관례 및 상점 또는 상점 그룹 판매 설정의 범위 및 특성을 정의하는 상점 또는 상점 그룹이 따르는 규칙 세트입니다. 비즈니스 정책은 비즈니스 정책의 규칙들을 구현하는 하나 이상의 비즈니스 정책 명령의 조합, 규칙이 시행되는 비즈니스 오브젝트에 대한 참조 및 비즈니스 정책 명령의 운영을 구성하는 특성 세트와 함께 실행됩니다.

가격 책정 정책

가격 책정 정책에는 가격 목록에 대한 참조가 들어 있으며 가격 목록에서 비즈니스 정책이 구현될 방식을 정의하는 복수 비즈니스 정책 명령과 연관될 수 있습니다. 정책은 상점이나 상점 그룹에 대해 정의될 수 있습니다. 정책이 상점 그룹에 대해 등록되는 경우, 정책은 해당 그룹에서 모든 상점에 의해 사용될 수 있습니다.

카탈로그 항목 운송

운송을 위한 상품 포장 방법을 포함하는 카탈로그 항목 운송 정보. 각 카탈로그 항목은 정의된 운송 정보의 다른 유형을 가질 수 있습니다. 예를 들어, 포장시 상품의 높이, 중량 및 길이입니다.

기타 가격 책정 자원

다음 자원은 가격 책정과 연관되어 있습니다.

- 거래 포지션 컨테이너를 소유하는 구성원. 거래 포지션 컨테이너는 단 하나의 소유자만 갖습니다.
- 상점 엔티티는 WebSphere Commerce 서버 데이터베이스에 있는 상점을 나타냅니다.
- 카탈로그에는 장기 구매 계약에서 참조될 카탈로그 항목이 들어 있습니다. 카탈로그는 온라인 카탈로그에 대한 모든 계층 정보 및 탐색 정보를 포함하고 있으며, 온라인 상점에서 표시 및 구매를 위해 사용 가능한 카탈로그 그룹 및 카탈로그 항목의 컬렉션입니다.
- 카탈로그 그룹 또는 카테고리 탐색 및 카탈로그 파티션을 목적으로 작성된 카탈로그 항목들의 일반 그룹입니다. 카탈로그 그룹은 하나의 카탈로그에 속하며, 두 개 이상의 카탈로그 그룹이나 카탈로그 항목을 포함합니다. 카탈로그 그룹을 두 개 이상의 카탈로그에 연관시킬 수 있습니다.
- 카탈로그 항목은 온라인 카탈로그에서 주문할 수 있는 판매 계획을 나타냅니다. 카탈로그 항목은 카탈로그 그룹에 속합니다. 판매 설정은 항상 하나의 카탈로그 항목과 연관됩니다.



WebSphere Commerce 서버의 가격 책정 자원의 구조에 대한 자세한 정보는
WebSphere Commerce 온라인 도움말의 가격 책정 오브젝트 및 데이터 모델을 참
조하십시오.

WebSphere Commerce에서 가격 책정 자원 작성

가격 책정 자원을 작성하는 데에는 두 가지 옵션이 있습니다.

- WebSphere Commerce 액셀러레이터의 상품 관리를 사용하여 가격을 작성해야 합니다. WebSphere Commerce 액셀러레이터의 도구를 사용하는 것이 소형 카탈로그의 가격 작성에 가장 적합합니다.
- WebSphere Commerce 로더 패키지가 로드할 수 있는 XML 파일로 또는 상점 서비스를 통해 공개될 수 있는 상점 아카이브의 일부분으로 가격을 작성해야 합니다. 이 방법은 많은 양의 데이터를 작성하는 데 가장 적합합니다.







WebSphere Commerce 액셀러레이터의 상품 관리 도구를 사용하여 가격 작성에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오. XML 파일로 가격 작성에 대한 자세한 정보는 『XML 파일에 가격 책정 자원 작성』을 참조하십시오.

XML 파일에 가격 책정 자원 작성

로더 패키지를 사용하여 데이터베이스로 로드할 수 있는 XML 파일 포맷으로 가격 책정 자원을 작성하십시오. 로더 패키지에 대한 자세한 정보는 247 페이지의 제 7 부 『상점 공개』를 참조하십시오.

1. 견본 상점의 가격 책정 자원을 작성하는 데 사용된 XML 파일을 검토하십시오. 견본 상점의 모든 파일들은 해당 상점 아카이브 파일에 있습니다.

상점 아카이브 파일들은 다음 디렉토리에 있습니다.





-  NT drive:\WebSphere\CommerceServer\samplstores
-  2000 drive:\Program Files\WebSphere\CommerceServer\samplstores
-  AIX /usr/WebSphere/CommerceServer/samplstores
-  Solaris /opt/WebSphere/CommerceServer/samplstores
-  Linux /opt/WebSphere/CommerceServer/samplstores
-  400 /qibm/proddata/WebCommerce/samplstores

주: WebSphere Commerce 온라인 도움말은 견본 상점에 포함된 각 데이터 자원에 대한 정보를 포함합니다.

각 견본 상점에는 가격 책정 정보를 포함하는 두 개의 offering.xml 파일이 포함됩니다. 상점 아카이브에서 offering.xml 파일을 보려면 ZIP 프로그램을 사용하

여 파일을 압축 해제하십시오. offering.xml 파일은 데이터 디렉토리에 있습니다. 언어별 offering.xml은 데이터 디렉토리의 로케일 고유의 서브디렉토리에 있습니다.

2. 361 페이지의 부록 B 『데이터 작성』 정보를 검토하십시오.
3. 견본 상점 아카이브에서 offering.xml 파일들 중 하나를 복사하거나 새 파일을 작성하여 offering.xml 파일을 작성하십시오. 자세한 정보는 offering.xml에 해당 DTD 파일을 참조하십시오. DTD 파일은 다음 디렉토리에 있습니다.

-  NT drive:\WebSphere\CommerceServer\xml\sar
-  2000 drive:\Program Files\WebSphere\CommerceServer\xml\sar
-  AIX /usr/WebSphere/CommerceServer/xml/sar
-  Solaris /opt/WebSphere/CommerceServer/xml/sar
-  Linux /opt/WebSphere/CommerceServer/xml/sar
-  400 /qibm/proddata/WebCommerce/xml/sar

4. 거래 포지션 컨테이너를 작성하십시오. 상점의 상품에 대해 가격을 책정하려면, 먼저 거래 포지션 컨테이너를 작성해야 합니다. 거래 포지션 컨테이너를 작성하려면, TRADEPOSCN 테이블에 정보를 추가하십시오.

- a. 다음 예를 사용하여 XML 파일의 TRADEPOSCN 테이블에 거래 포지션 컨테이너를 작성하십시오.

```
<tradeposcn
tradeposcn_id="@tradeposcn_id_101"
member_id("&MEMBER_ID"
markfordelete="0"
name="ToolTech"
precedence="0"
/>
```

여기서

- tradeposcn_id는 생성된 고유 키입니다.
- member_id는 거래 포지션 컨테이너의 소유자입니다.
- markfordelete는 다음과 같습니다.
 - 0 = TradingPositionContainer를 사용할 수 있습니다.
 - 1 = TradingPositionContainer가 삭제를 위해 표시되었으며(DBClean 유틸리티라고도 함) 사용되어서는 안 됩니다.
- name은 특정 소유자에게 고유한 거래 포지션 컨테이너의 니모닉 이름입니다.
- precedence는 하나 이상의 거래 포지션 컨테이너가 특정한 시간에 규정될 때 최상위 PRECEDENCE를 가진 거래 포지션 컨테이너가 사용되는 것입니다.

5. CATGRPTPC 테이블에 정보를 추가하여 마스터 카탈로그를 거래 포지션 컨테이너와 연관시켜야 합니다. 마스터 카탈로그를 거래 포지션 컨테이너와 연관시킬 때, 마스터 카탈로그의 모든 카탈로그 항목은 표준 가격을 가져야 합니다. 마스터 카탈로그에 대한 자세한 정보는 84 페이지의 『상점 카탈로그 자원 표시』를 참조하십시오.

a. 다음 예를 사용하여 CATGRPTPC 테이블에 정보를 추가하여 마스터 카탈로그를 거래 포지션 컨테이너에 연관시켜야 합니다.

```
<catgrptpc
catalog_id="@catalog_id_1"
tradeposcn_id="@tradeposcn_id_101"
/>
```

여기서,

- catalog_id는 마스터 카탈로그입니다.
- tradeposcn_id는 거래 포지션 컨테이너입니다.

6. OFFER 및 OFFERPRICE 테이블에 정보를 추가하여 카탈로그 항목에 대한 판매 설정 및 판매 가격을 작성하십시오.

a. 다음 예를 사용하여 OFFER 테이블에 정보를 추가하여 카탈로그 항목에 대한 판매 설정을 작성하십시오. 가격을 작성하기 전에 카탈로그 항목을 작성해야 함에 유의하십시오. 카탈로그 항목 작성에 대한 자세한 정보는 84 페이지의 『상점 카탈로그 자원 표시』를 참조하십시오.

```
offer
offer_id="@offer_id_138"
startdate="2000-06-19 00:00:00.000000"
catentry_id="@product_id_102"
precedence="0"
published="1"
identifier="1"
flags="1"
tradeposcn_id="@tradeposcn_id_101"
/>
```

여기서

- offer_id는 생성된 고유 키입니다.
- startdate는 이 판매 설정이 효과가 있는 시간 범위의 시작입니다.
- catentry_id는 판매용으로 설정된 카탈로그 항목입니다.
- precedence는 하나 이상의 판매 설정이 특정한 시간에 효과적일 때 최상위 PRECEDENCE 를 가진 판매 설정이 사용되는 것입니다.
- published는
 - 0 = 공개되지 않음(일시적으로 사용 안함)
 - 1 = 공개
 - 2 = 삭제를 위해 표시됨(공개되지 않음)

- `identifier`는 지정된 카탈로그 항목 및 거래 포지션 컨테이너와 함께 이련 판매 설정을 고유하게 식별하는 번호입니다.
 - `flags`는
 - `1 = shiptoAddressRequired - 1`이면, `OrderItem`이 이러한 판매 설정을 참조하지만 운송 주소를 갖지 않을 경우 `OrderPrepare`가 오류를 반환합니다.
 - `tradeposcn_id`는 이 판매 설정이 일부분을 구성하는 거래 포지션 컨테이너입니다.
- b. 다음 예를 사용하여 `OFFERPRICE` 테이블에 정보를 추가하여 카탈로그 항목에 대한 판매 가격을 작성하십시오. 판매 가격은 카탈로그 항목에서 판매용으로 제공된 실제 가격입니다. 가격을 작성하기 전에 카탈로그 항목을 작성해야 함에 유의하십시오. 카탈로그 항목 작성에 대한 자세한 정보는 84 페이지의 『상점 카탈로그 자원 표시』를 참조하십시오.

```
<offerprice
offer_id="@offer_id_138"
currency="USD"
price="590.00"
/>
```

여기서

- `offer_id`는 이 가격과 연관된 판매 설정입니다.
- `currency`는 가격이 책정된 통화입니다.
- `price`는 판매 설정에 의해 언급된 상품의 정상 수량에 대한 가격입니다 (`CATENTSHIP.NOMINALQUANTITY` 참조).

주: 상점에서 복수 통화를 표시하려면, 각 통화에 대한 `OFFERPRICE` 테이블에서 별도의 XML 항목을 작성하십시오. 예를 들어, 캐나다 달러로 통화를 표시하려면, 새 XML 항목에서 `currency="CAD"`를 사용하십시오. `price` 값은 캐나다 달러로 가격을 반영하도록 변경됩니다. 아니면, 변환을 사용하여 고객이 선택한 통화에 근거하여 다른 비율을 표시하게 할 수 있습니다. 자세한 정보는 140 페이지의 『XML 파일을 사용하여 통화 자원 작성』을 참조하십시오.

- c. 카탈로그의 모든 카탈로그 항목에 대해 a 및 b 단계를 반복하십시오.



@ 및 &의 사용에 대한 자세한 정보는 361 페이지의 부록 B 『데이터 작성』을 참조하십시오.

제 10 장 장기 구매 계약 자원

WebSphere Commerce에서 모든 상점 고객은 장기 구매 계약에 따라 구매해야 합니다. 장기 구매 계약을 체결하면 고객은 특정 조건에 따라 지정된 기간 동안 지정된 가격으로 상점에서 상품을 구매할 수 있습니다. 상점 키탈로그를 볼 때, 고객은 상점과 체결한 장기 구매계약에 따른 상품들만을 보게 될 것입니다.

상점과 장기 구매 계약을 체결하지 않은 고객(예: 게스트 구매자)이 상점에서 구매를 원하거나, 고객이 체결한 장기 구매 계약에 해당하지 않는 상품을 구매하고 싶다면, 상점에 기본 장기 구매 계약이 있어야 합니다.

중요

WebSphere Commerce Professional Edition은 기본 장기 구매 계약만을 지원합니다.

상점 기본 장기 구매 계약 이외의 장기 구매 계약은 WebSphere Commerce Business Edition에 의해서만 지원됩니다.

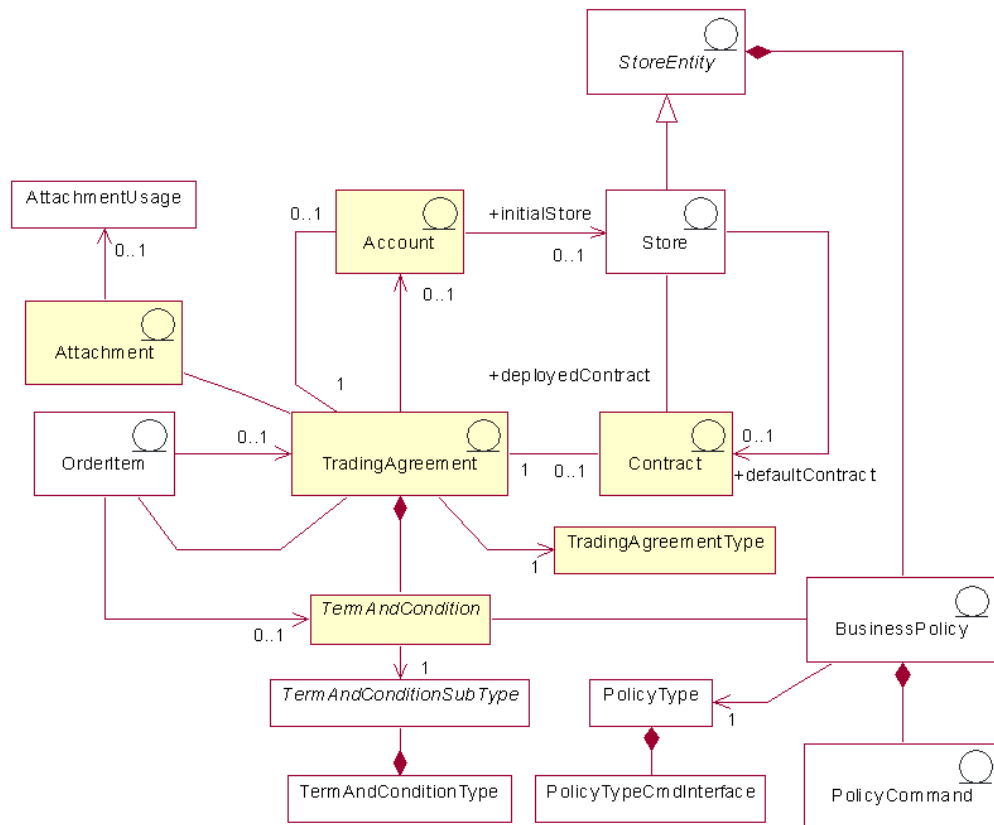
모든 고객이 상점에서 구매할 수 있게 하려면, WebSphere Commerce로 작성된 상점에 다음 사항이 있어야 합니다.

- 비즈니스 정책
- 기본 장기 구매 계약

비즈니스 정책은 기본 장기 구매 계약에 의해 참조되고 모든 고객이 상점에서 구매하도록 허용합니다.

WebSphere Commerce의 기본 장기 구매 계약 이해

다음 도표는 WebSphere Commerce의 장기 구매 계약 구조를 보여줍니다.



이 도표 및 상점 데이터 질의 기타 모두는 WebSphere Commerce 서버 정보 모델의 일부입니다. 정보 모델에 대한 자세한 정보는 29 페이지의 『상점 데이터 정보 모델』을 참조하십시오. 이 도표에서 사용된 규칙에 대한 자세한 정보는 359 페이지의 부록 A 『UML 범례』를 참조하십시오.

계정(비즈니스 계정)

비즈니스 계정은 구매자 조직과 판매자 조직 간의 관계를 나타냅니다. 비즈니스 계정은 여러 거래 계약을 조직하고 인보이스 사용자 정의, 구매 주문 확인 또는 판매자와 구매자의 신용 유지와 같이 구매자와 판매자 간의 관계와 관련 규정을 지정하는 데 사용할 수 있습니다.

장기 구매 계약은 구매자와 판매자 간의 계약을 나타내기 때문에 비즈니스 계정과 연관되어 있습니다. 이에 대한 예외는 비즈니스 계정과 연관되지 않는 상점 기본 장기 구매 계약입니다. 비즈니스 계정은 많은 장기 구매 계약이 계정과 연관되도록 할 수 있습니다.

비즈니스 계정은 거래 계약의 한 유형입니다. 거래 계약의 설명에 대해서는 109 페이지의 『거래 계약』을 참조하십시오.

중요: 비즈니스 계정은 WebSphere Commerce Business Edition에 의해서만 지원됩니다.

장기 구매 계약

상점과 연관된 실질적인 장기 구매 계약에는 전개 장기 구매 계약과 기본 장기 구매 계약이라는 두 가지 유형이 있습니다. 전개 장기 구매 계약은 구매자 조직이나 개별 구매자에 해당하는 것으로, 상점을 작성한 후 WebSphere Commerce 액셀러레이터를 사용하여 작성할 수 있습니다. 하나의 전개 장기 구매 계약은 하나의 비즈니스 계정과 연관이 있습니다. 기본 장기 구매 계약은 상점과 기타 모든 장기 구매 계약을 체결하지 않은 구매자에 대한 상점의 기본 행위를 정의합니다. 기본 장기 구매 계약은 XML 파일을 사용하여 작성되며 상점에 대하여 오직 하나의 기본 장기 구매 계약만이 정의됩니다. 장기 구매 계약에 대한 자세한 정보는 온라인 정보를 참조하십시오. 기본 장기 구매 계약 자원 작성에 대한 자세한 정보는 111 페이지의 『WebSphere Commerce에서 기본 장기 구매 계약 자원 작성』을 참조하십시오.

장기 구매 계약은 거래 계약의 한 유형입니다. 거래 계약의 설명에 대해서는 『거래 계약』을 참조하십시오.

거래 계약

WebSphere Commerce는 구매자와 판매자 간의 대화형 작업을 주관하는 많은 거래 메커니즘을 제공합니다. 다음 거래 메커니즘은 WebSphere Commerce의 서로 다른 판에서 지원됩니다.

- 경매(Business 및 Professional Edition에서 지원됨)
- 비즈니스 계정(Business Edition에서만 지원됨)
- 장기 구매 계약(이 장의 앞부분에서 논의된 제한사항 참조)
- RFQ(Business Edition에서만 지원됨)

이 모든 거래 메커니즘은 공통된 특성을 갖습니다. 예를 들어 모든 거래 메커니즘에는 참여자가 있고, 참여자들은 모두 거래 메커니즘의 행위를 주관하는 규칙을 갖고 있습니다. 거래 메커니즘의 행위를 주관하는 규칙은 WebSphere Commerce의 규정으로 알려져 있습니다.

하나의 거래 계약은 하나의 거래 메커니즘의 인스턴스를 나타내며, 하나의 거래 메커니즘의 해당 인스턴스의 특성을 기록합니다. WebSphere Commerce의 장기 구매 계약, 비즈니스 계정 및 RFQ는 각각 거래 계약으로 표시됩니다. WebSphere Commerce의 모든 경매를 주관하는 거래 계약은 단일합니다.

하나의 거래 계약은 TRADING 테이블에 저장된 하나의 프로파일, PARTICIPANT 테이블에 저장된 참여자, TERMCOND 테이블에 저장된 규정 및 ATTACHMENT 테이블에 URI(Universal Resource Identifiers)로 저장된 선택적 첨부로 구성되어 있습니다.

니다. 하나의 거래 계약이 여러 개의 첨부 파일을 가질 수 있기 때문에, 첨부 파일은 TRDATTACH 테이블을 사용하는 거래 계약과 관련되어 있습니다. 첨부 파일이 RFQ를 지원하지 않음에 주의하십시오.

일반 거래 계약 이외에도, 각 유형의 거래 계약은 자기 테이블의 거래 계약 유형에 고유한 추가 정보를 저장합니다. CONTRACT는 장기 구매 계약 특정 정보를, RFQ는 RFQ 특정 정보를 그리고 ACCOUNT는 비즈니스 계정 특정 정보를 저장합니다.

규정

규정은 거래 계약의 행위 및 특성을 정의합니다.

장기 구매 계약의 경우, 규정은 장기 구매 계약이 구매자 조직에서 구현되는 방법을 정의합니다. 규정은 장기 구매 계약에 따른 판매 항목, 판매 항목의 가격, 항목 운송 방법, 주문 지불 방법, 항목 반품 처리 방법, 주문 승인 방법 및 주문 운송 장소 등을 정의합니다.

상점 운영의 일부가 비즈니스 정책에 의해 정의되기 때문에 많은 규정이 비즈니스 정책을 참조합니다. 규정은 참조하는 비즈니스 정책에 매개변수를 제공합니다. 비즈니스 정책에 매개변수를 제공하면 각 장기 구매 계약에 대한 비즈니스 정책의 행위를 수정할 수 있습니다.

비즈니스 정책

비즈니스 정책은 상점 또는 상점 그룹이 따르는 규칙 세트입니다. 비즈니스 정책은 상점 또는 상점 판매 설정의 비즈니스 프로세스, 업종 관례 및 상점 또는 상점 그룹 판매 설정의 범위 및 특성 등을 정의합니다. 이것들은 상점 또는 상점 그룹 내에서 허용되고 지원되는 모든 관례에 대한 중앙 소스 및 참조 템플릿입니다.

WebSphere Commerce에서 비즈니스 정책은 비즈니스 정책의 규칙들을 구현하는 하나 이상의 비즈니스 정책 명령의 조합, 규칙이 시행되는 비즈니스 오브젝트에 대한 참조 및 비즈니스 정책 명령의 운영을 구성하는 특성 세트와 함께 실행됩니다. 규정은 참조하는 비즈니스 정책에 대해 매개변수를 제공합니다. 이것은 비즈니스 정책의 행위가 비즈니스 정책을 참조하는 규정에 따라 수정되는 것을 허용합니다.

첨부

첨부 파일은 거래 계약의 다른 요소에 의해 해당하지 않는 거래 계약에 대한 추가 정보를 제공합니다. 예로는 RFQ 요구사항 및 REG에 대한 일반 설명에 대한 추가 정보를 제공하는 파일이 있습니다. 거래 계약은 복수 첨부 파일을 가질 수 있습니다. 첨부 파일은 WebSphere Commerce의 외부에 저장되며 거래 계약은 URI를 첨부 파일에 저장합니다. URI의 예는 다음을 포함합니다.

- <http://www.mycompany.com/information/document1.txt>
- <file:///home/joeuser/mydocs/document1>

- <ftp://ftp.mycompany.com/information/attachment.txt>

모든 첨부는 어떤 것에 대한 첨부인지를 지시하는 첨부 사용법에 지정될 수 있습니다. 첨부 사용법은 첨부 선택적 특성입니다.

주문 항목

주문 항목은 주문과 함께 포함된 상품입니다. 서로 다른 장기 구매 계약 거래 계약에 따라 단일 주문에서 서로 다른 주문 항목이 구매될 수 있습니다. 구매자들은 상점 디자인에 따라 쇼핑 플로어가 시작할 때 또는 하나의 항목에 주문을 추가할 때 구매하는 장기 구매 계약 거래 계약을 선택할 수 있습니다. 서로 다른 장기 구매 계약 거래 계약에 따라 항목을 구매할 때, 다음 규칙이 적용됩니다.

- 하나의 주문에서 모든 항목에 대한 장기 구매 계약 거래 계약은 최소한 하나의 지불 방법을 공유해야 합니다. 만약 하나의 항목에 대한 장기 구매 계약이 하나의 지불 방법을 공유하지 않는다면, 구매자는 해당 항목을 주문에 추가할 수 없습니다. 하나의 주문에서 모든 항목이 공유하는 지불 방법만이 주문에 대한 지불로 사용될 수 있습니다.
- 하나의 주문의 모든 항목은 동일한 비즈니스 계정 또는 상점 기본 장기 구매 계약에 속하는 장기 구매 계약 거래로부터 와야 합니다.



WebSphere Commerce의 장기 구매 계약 자원 구조에 대한 자세한 정보는
WebSphere Commerce 온라인 도움말의 장기 구매 계약 데이터 모델을 참조하십시오.
시도.

WebSphere Commerce에서 기본 장기 구매 계약 자원 작성

기본 장기 구매 계약은 상점의 기본 행위를 정의합니다. 모든 장기 구매 계약과 같이 사용 가능한 상품, 가격, 지불 방법, 운송 방법 및 다른 상점 행위 등을 설정할 수 있습니다.

WebSphere Commerce 견본 상점에서 제공되는 상점 기본 장기 구매 계약에는 다음을 지정하는 규정이 포함되어 있습니다.






- 고객은 마스터 카탈로그에 설정된 표준 가격으로 상점의 마스터 카탈로그에서 사용할 수 있는 모든 상품을 구매(할인 또는 가격 인상 없음)할 수 있습니다.
- 모든 운송 경비는 판매자(상점)에게 지불됩니다.
- 고객은 일정한 날짜 이내에 위약금없이 구매 상품 반품 할 수 있습니다.
- 고객은 최초 구매시 사용된 동인한 지불 방법을 사용하여 환불 수령할 수 있습니다.

또한, 가장 일반적인 상점의 기본 장기 구매 계약 버전은 구매자가 사용할 수 있는 지불 또는 운송 방법을 제한하는 규정을 생략합니다. 규정을 생략하면 구매자는 상점에서 지원되는 모든 기본 지불 방법을 사용할 수 있고 상점에서 사용할 수 있는 모든 운송 방법을 사용할 수 있습니다.

기본 장기 구매 계약의 특성은 해당 규정에 정의되어 있습니다. 일부 규정은 비즈니스 정책을 참조합니다. 비즈니스 정책 및 규정에 대한 자세한 정보는 온라인 정보를 참조하십시오.

기본 장기 구매 계약 자원을 작성하려면, 다음을 수행하십시오.

1. 규정, 장기 구매 계약, 기본 장기 구매 계약 및 비즈니스 정책에 관한 온라인 정보를 검토하십시오.
2. `wcs.bootstrap.xml` 파일에서 정의된 비즈니스 정책을 검토하십시오. `wcs.bootstrap.xml` 파일에 대한 자세한 정보는 온라인 정보를 참조하십시오.
3. 견본 상점에 대한 기본 장기 구매 계약 자원을 작성하는 데 사용되는 파일을 검토하십시오. 모든 견본 상점 파일은 해당 상점 아카이브 파일에 있습니다. 각 견본 상점은 추가 비즈니스 정책 정보 및 기본 장기 구매 계약 정보를 포함하는 `businesspolicy.xml` 및 `contract.xml` 파일을 포함합니다. 상점 아카이브 파일들은 다음 디렉토리에 있습니다.

-  `/usr/WebSphere/CommerceServer/samplestores`
-  `/qibm/proddata/WebCommerce/samplestores`
-  `/opt/WebSphere/CommerceServer/samplestores`
-  `/opt/WebSphere/CommerceServer/samplestores`
-  `drive:\Program Files\WebSphere\CommerceServer\samplestores`
-  `drive:\WebSphere\CommerceServer\samplestores`

주:

- a. WebSphere Commerce 온라인 도움말에는 견본 상점에 포함된 데이터 자원 각각에 대한 정보가 포함됩니다.
 - b. 상점 아카이브에서 `businesspolicy.xml` 및 `contract.xml` 파일을 보려면 ZIP 프로그램을 사용하여 압축을 푸십시오. 파일은 `data` 디렉토리에 있습니다.
 - c. WebSphere Commerce Business Edition에서 제공되는 ToolTech 견본 상점에 대한 장기 구매 계약 자원 파일은 상점 기본 장기 구매 계약이 아닌 장기 구매 계약에 대한 정보를 포함합니다.
4. 361 페이지의 부록 B 『데이터 작성』 정보를 검토하십시오.
 5. 견본 상점 아카이브의 `businesspolicy.xml` 파일 중 하나를 복사하거나 새 파일을 작성해서 `businesspolicy.xml` 파일을 작성하십시오. 새 파일 작성에 대한 지시사항은 113 페이지의 『비즈니스 정책 XML 파일 작성』에 있습니다. 논의된 것들과 다른 비즈니스 정책을 작성하려면, `businesspolicy.xml`에 해당하는 DTD 파일을 보십시오. DTD 파일은 다음 디렉토리에 있습니다.

-  `/usr/WebSphere/CommerceServer/xml/sar`

- ▶ 400 /qibm/proddata/WebCommerce/xml/sar
 - ▶ Linux /opt/WebSphere/CommercServer/xml/sar
 - ▶ Solaris /opt/WebSphere/CommercServer/xml/sar
 - ▶ 2000 drive:\Program Files\WebSphere\CommerceServer\xml\sar
 - ▶ NT drive:\WebSphere\CommerceServer\xml\sar
6. 로더 패키지를 사용하여 businesspolicy.xml 파일을 로드하십시오. 로더 패키지에 대한 자세한 정보는 247 페이지의 제 7 부 『상점 공개』를 참조하십시오. 다국어 지원 상점을 작성하는 경우에는 상점이 지원하는 로케일 각각에 대한 별도 XML 파일을 작성할 수 있습니다. 로케일 고유의 파일은 쉽게 번역할 수 있도록 모든 설명 정보를 지정해야 합니다.
7. 견본 상점 아카이브의 contract.xml 파일 중 하나를 복사하거나 새 파일을 작성해서 contract.xml 파일을 작성하십시오. 새 파일 작성에 대한 지시사항은 115 페이지의 『기본 장기 구매 계약 XML 파일 작성』에 있습니다. 보다 더 복잡한 기본 장기 구매 계약을 작성하려면 contract.xml 파일 구조를 정의하는 B2BTrading.dtd 파일을 검토하십시오. B2BTrading.dtd는 다음 디렉토리에 있습니다.
- ▶ AIX /usr/WebSphere/CommerceServer/xml/trading
 - ▶ 400 /qibm/proddata/WebCommerce/xml/trading
 - ▶ Linux /opt/WebSphere/CommerceServer/xml/trading
 - ▶ Solaris /opt/WebSphere/CommerceServer/xml/trading
 - ▶ 2000 drive:\Program Files\WebSphere\CommerceServer\xml\trading
 - ▶ NT drive:\WebSphere\CommerceServer\xml\trading
8. ContractImportApprovedVersion 명령을 사용하여 장기 구매 계약을 공개하십시오. 자세한 정보는 323 페이지의 제 29 장 『비즈니스 계정 및 장기 구매 계약 공개』를 참조하십시오. 또한 ContractImportApprovedVersion 명령에 대한 정보는 온라인 정보에서 사용 가능합니다.

WebSphere Commerce Business Edition 사용자는 WebSphere Commerce 액셀러레이터를 사용하여 특정 고객에 대한 장기 구매 계약을 정의할 수 있습니다. 특정 고객에 대한 장기 구매 계약에 관한 자세한 정보는 온라인 정보를 참조하십시오.

비즈니스 정책 XML 파일 작성

WebSphere Commerce가 상점의 기본 장기 구매 계약에서 규정이 참조할 수 있는 많은 비즈니스 정책을 제공하는 반면, 일부 비즈니스 정책은 사용자가 정의해야 합니다. 사용자는 반품 비용, 반품 승인 및 상점 기본 장기 구매 계약 기간이 참조하는 가격 책

정 비즈니스 정책을 정의해야 합니다. 이러한 비즈니스 정책에 대한 명령은 수정없이 제공되며 또 사용될 수 있습니다. 자신만의 비즈니스 정책을 작성하고 싶다면, *IBM WebSphere Commerce 프로그래머 안내서*를 참조하십시오.

상점에 대해 비즈니스 정책을 작성하려면, 비즈니스 정책을 작성하여 하나 이상의 명령을 비즈니스 정책과 연관시켜야 합니다. 비즈니스 정책을 작성하려면, POLICY 테이블에 정보를 추가해야 합니다. 하나의 명령을 비즈니스 정책에 연관시키려면, POLICYCMD 테이블에 정보를 추가해야 합니다.

비즈니스 정책을 작성하고 명령을 정책에 연관시키려면, 다음을 수행하십시오.

1. POLICY 테이블에 정보를 추가하여 비즈니스 정책 XML 파일에 비즈니스 정책을 작성해야 합니다. 참조로 다음 예를 사용하십시오.

```
<policy
policy_id="@policy_id_10"
policyname="MasterCatalogPriceList"
policytype_id="Price"
storeent_id="@storeent_id_1"
properties="name=InFashion&member_id=&MEMBER_ID"
/>
```

여기서,

- policy_id는 비즈니스 정책에 대한 고유한 수치 식별자입니다.
- policyname은 비즈니스 정책에 대한 고유한 이름입니다.
- policytype_id는 정의되는 정책 유형입니다. 올바른 policytype_id는 다음과 같습니다.
 - InvoiceFormat
 - Payment
 - Price
 - ProductSet
 - ReturnApproval
 - ReturnCharge
 - ReturnPayment
 - ShippingCharge
 - ShippingPayment
- storeent_id는 상점 또는 상점 그룹입니다.
- properties는 비즈니스 정책 명령으로 보내진 이름 값 쌍의 목록입니다.

2. POLICYCMD 테이블에 정보를 추가하여 비즈니스 정책 XML 파일의 비즈니스 정책과 명령을 연관시켜야 합니다. 참조로 다음 예를 사용하십시오.

```
<policycmd
  policy_id="@policy_id_10"
  businesscmdclass=
    "com.ibm.com.commerce.price.commands.CalculateContractPricesCmdImpl"
/>
```

여기서,

- policy_id는 명령을 연관시키는 데 사용하는 비즈니스 정책의 수치 식별자입니다.
- businesscmdclass는 비즈니스 정책을 구현하는 Java 클래스의 이름입니다.

businesscmdclass 속성에 있는 행 바꾸기는 표시 전용입니다.



@ 및 &의 사용에 대한 자세한 정보는 361 페이지의 부록 B 『데이터 작성』을 참조하십시오.

기본 장기 구매 계약 XML 파일 작성

기본 장기 구매 계약을 작성하려면, 장기 구매 계약, 장기 구매 계약 소유자, 장기 구매 계약 설명, 장기 구매 계약 참여자 및 장기 구매 계약의 규정을 정의해야 합니다. 장기 구매 계약 정보는 CONTRACT, PARTICIPNT, TRADING, TERMCOND 등 네 개의 테이블에 저장됩니다.

기본 장기 구매 계약은 STOREDEF 데이터베이스 테이블을 사용한 상점과 연관됩니다. WebSphere Commerce Business Edition 사용자에게 있어서, 기본 장기 구매 계약이 아닌 장기 구매 계약은 STORECNTR 데이터베이스 테이블을 사용하여 상점과 연관됩니다.

기본 장기 구매 계약을 작성하려면, 다음을 수행하십시오.

1. XML 파일에서 기본 장기 구매 계약을 정의해야 합니다. 기본 장기 구매 계약은 다음과 같이 XML 파일의 처음에 정의되어 있습니다.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Trading SYSTEM "B2BTrading.dtd">
<Trading>
<Contract state="Active" origin="Manual"
  name="&STORE_IDENTIFIER; Default Contract" majorVersionNumber="1"
  minorVersionNumber="0" contractUsage="Default">
```

Contract 구성요소의 행 바꾸기는 표시 전용임에 유의하십시오.

2. 장기 구매 계약 소유자를 정의해야 합니다. 참조로 다음 예를 사용하십시오.

```
<ContractOwner>
  <Member>
    <Organization distinguishName="&MEMBER_IDENTIFIER;" />
  </Member>
</ContractOwner>
```

여기서, `distinguishName`은 LDAP 인식 이름 포맷에 있는 장기 구매 계약을 소유하고 있는 사용자의 이름입니다. 예를 들면, `uid=erickoeck, ou=People, dc=ibm, dc=com`이 있습니다.

3. 장기 구매 계약 XML 파일에서 장기 구매 계약 설명을 정의해야 합니다. 참조로 다음 예를 사용하십시오.

```
<ContractDescription title="This is a store default contract." languageId="-1">
</ContractDescription>
```

여기서

- `title`은 장기 구매 계약의 텍스트 설명입니다.
- `languageId`는 직위가 있는 언어입니다. 다음 값들은 `languageId`에 대하여 사전정의되어 있습니다.

- -1 (영어 - 미국)
- -2 (불어)
- -3 (독어)
- -4 (이탈리아어)
- -5 (스페인어)
- -6 (포르투갈어)
- -7 (중국어)
- -8 (대만어)
- -9 (한국어)
- -10 (일본어)

상점에 대한 언어 자원을 갱신하여 `languageId`에 대해 추가 값을 정의할 수 있습니다. 언어 자원에 대한 추가 정보는 135 페이지의 제 14 장 『언어 자원』을 참조하십시오.

4. 장기 구매 계약 XML 파일에서 장기 구매 계약 참여자를 정의해야 합니다. 참조로 다음 예를 사용하십시오.

```
<Participant role="Buyer">
</Participant>
<Participant role="Seller">
  <Member>
    <Organization distinguishName="&MEMBER_IDENTIFIER;" />
  </Member>
</Participant>
```

여기서, `distinguishName`은 LDAP 인식 이름 포맷에 있는 장기 구매 계약에 대한 판매자인 사용자의 이름입니다. 예를 들면, `uid=erickoeck, ou=People, dc=ibm, dc=com`이 있습니다. 많은 경우, 이는 장기 구매 계약 소유자와 동일합니다.

주: 장기 구매 계약은 구매자 역할을 하는 모든 사용자에게 사용 가능하기 때문에 구매자 참여 역할에는 어떤 구성원도 지정되지 않습니다.

5. 장기 구매 계약 XML 파일에서 규정을 정의해야 합니다. XML 구성요소 및 속성은 규정 유형이 다양하기 때문에 다릅니다. B2BTrading.dtd 파일을 사용하여 XML 구성요소 및 속성이 각 유형의 규정을 사용하는 방법에 대해 배워야 합니다. 규정을 정의할 때 다음 속성들이 공통적으로 사용됩니다.

policyName

규정이 참조하는 비즈니스 정책의 이름. 이 이름은 POLICY.POLICYNAME에 저장됩니다.

policyType

규정이 참조하는 비즈니스 정책의 유형. 올바른 값은 다음과 같습니다.

- Price
- ProductSet
- InvoiceFormat
- Payment
- ReturnApproval
- ReturnCharge
- ReturnPayment
- ShippingCharge
- ShippingMode

storeIdentity

규정에 대한 상점 또는 상점 그룹.

distinguishName

상점 또는 상점 그룹을 소유하는 사용자의 이름. 이름은 LDAP 인식 이름 포맷이어야 합니다. 예를 들면, uid=wcsadmin,o=Root Organization입니다.

다음 견본 규정은 규정이 정의하는 것의 설명에 선행합니다.

- 모든 구매자는 마스터 카탈로그에 설정된 가격으로 상점의 마스터 카탈로그에 있는 모든 항목을 구매할 수 있습니다.

```
<TermCondition>
  <PriceTC>
    <PriceTCMasterCatalogWithOptionalAdjustment>
    </PriceTCMasterCatalogWithOptionalAdjustment>
  </PriceTC>
</TermCondition>
```

- 구매자는 판매자에게 모든 운송 경비를 지불합니다.

```

<TermCondition>
  <ShippingTC>
    <ShippingTCShippingCharge>
      <PolicyReference policyName="StandardShippingChargeBySeller"
        policyType="ShippingCharge" storeIdentity="&STORE_IDENTIFIER;">
        <Member>
          <User distinguishName="&MEMBER_IDENTIFIER;">
        </Member>
      </PolicyReference>
    </ShippingTCShippingCharge>
  </ShippingTC>
</TermCondition>

```

PolicyReference 요소에 있는 행 바꾸기는 표시 전용입니다.

- 구매자는 반품 비용 없이 상품을 반품할 수 있습니다. 상품은 ApprovalByDays 비즈니스 정책에 정의된 날짜 이내에 반품되어야 합니다.

```

<TermCondition>
  <ReturnTC>
    <ReturnTCReturnCharge>
      <ReturnChargePolicyReference>
        <PolicyReference policyName="NoCharges"
          policyType="ReturnCharge"
          storeIdentity="&STORE_IDENTIFIER;">
          <Member>
            <Organization distinguishName="&MEMBER_IDENTIFIER;">
          </Member>
        </PolicyReference>
      </ReturnChargePolicyReference>
      <ReturnApprovalPolicyReference>
        <PolicyReference policyName="ApprovalByDays"
          policyType="ReturnApproval"
          storeIdentity="&STORE_IDENTIFIER;">
          <Member>
            <Organization distinguishName="&MEMBER_IDENTIFIER;">
          </Member>
        </PolicyReference>
      </ReturnApprovalPolicyReference>
    </ReturnTCReturnCharge>
  </ReturnTC>
</TermCondition>

```

PolicyReference 요소에 있는 행 바꾸기는 표시 전용입니다.

WebSphere Commerce Business Edition 사용자 주:

상점 기본 장기 구매 계약에서 이 규정을 생략하는 것은 기본적으로, 상점이 반품을 받지 않는다는 것을 표시합니다. 하지만 다른 장기 구매 계약은 반품 규정을 정의하므로써 구매자가 반품하게 할 수 있습니다.

WebSphere Commerce Professional Edition 사용자 주:

상점 기본 장기 구매 계약에서 이 규정을 생략하는 것은 상점이 반품을 받지 않는다는 것을 표시합니다.

- 환불은 주문을 완료할 때 구매자가 사용한 것과 동일한 지불 방법을 사용하여 지불됩니다.


```
<TermCondition>
  <ReturnTC>
    <ReturnTCRefundPaymentMethod>
      <PolicyReference policyName="UseOriginalPayment"
        policyType="ReturnPayment" storeIdentity="&STORE_IDENTIFIER;">
        <Member>
          <User distinguishName="&MEMBER_IDENTIFIER;">
          </Member>
        </PolicyReference>
      </ReturnTCRefundPaymentMethod>
    </ReturnTC>
  </TermCondition>
```

PolicyReference 구성요소의 행바꾸기는 표시 전용임에 유의하십시오.



@ 및 &의 사용에 대한 자세한 정보는 361 페이지의 부록 B 『데이터 작성』을 참조하십시오.

제 11 장 이행 지원

서비스 센터는 상점에서 재고 창고 및 운송과 수령 센터의 두 가지로 사용됩니다. 한 상점은 연관된 하나 이상의 서비스 센터를 가질 수 있습니다. 서비스 센터는 상점의 상품 재고 및 운송을 관리합니다. 이행은 오더피킹, 포장 및 운송을 포함합니다. 오더피킹은 서비스 센터에서 하나 이상의 출고 중에서 상품을 선택하는 것이고, 포장은 이런 상품을 운송 컨테이너에 넣는 것이며, 운송을 이를 고객에게 전달하는 것입니다.

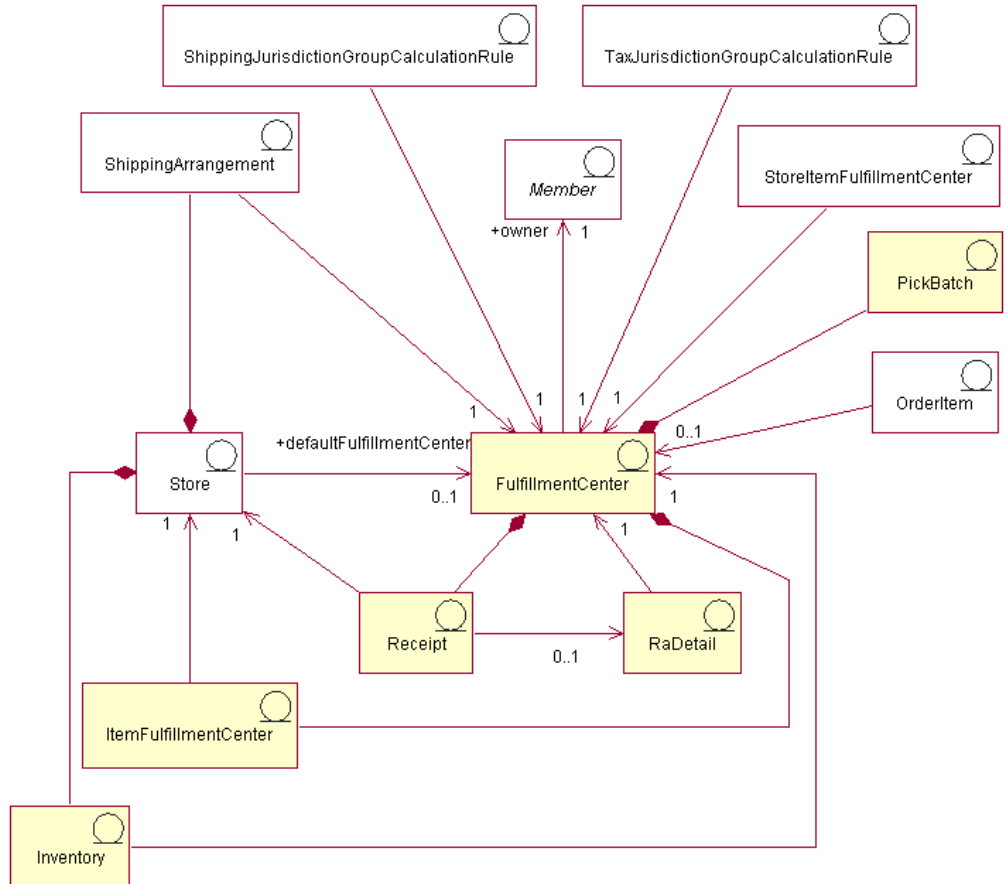
상품은 상품 마법사와 상품 노트북의 이행으로 구성됩니다. 상품 구성은 재고 추적, 백 오더 허용, 백오더 강제 실행, 별도 출고 및 상품을 반품할 수 없음을 지정하는 옵션을 제공합니다.

일반적으로 서비스 센터에서는 많은 사람이 동시에 작업하지만, 그들이 수행하는 태스크는 모두 다릅니다. WebSphere Commerce 액셀러레이터는 대부분의 공통 태스크를 역할로 구분하고, 이런 역할을 사용자에게 지정합니다. WebSphere Commerce 액셀러레이터에서 서비스 센터와 관계되는 하나 이상의 역할을 지정받은 경우, 로그인시 하나의 서비스 센터를 선택해야 합니다.

주: 이행, 서비스 센터 및 역할에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오.

WebSphere Commerce에서 이행 자원 이해

이행 자원을 이해하려면, 이행과 상점 사이의 관계를 이해할 필요가 있습니다. 이는 정보 모델을 사용하여 설명할 수 있습니다. 다음 절은 재고가 상점 및 다른 자원에 대해 갖는 관계에 대해 설명합니다.



이 도표 및 상점 데이터 절의 기타 모두는 WebSphere Commerce 서버 정보 모델의 일부입니다. 정보 모델에 대한 자세한 정보는 29 페이지의 『상점 데이터 정보 모델』을 참조하십시오. 이 도표에 사용된 규칙에 대한 자세한 정보는 359 페이지의 부록 A 『UML 범례』를 참조하십시오.

서비스 센터

이전 도표에서 *서비스 센터*는 이행 처리의 중심입니다. 서비스 센터는 소유자가 있으며 MEMBER 테이블에 정의됩니다. 개별 상점은 한 서비스 센터와 연관되고, 기본 서비스 센터를 가집니다. 한 서비스 센터는 여러 상점과 연관될 수 있습니다. 도표에서 표시된 것처럼, 상점과 서비스 센터 사이에는 여러 상호 작용이 있습니다. 상점 자원에 대한 자세한 정보는 45 페이지의 『WebSphere Commerce의 상점 자원 이해』를 참조하십시오.

수량증

서비스 센터는 항목의 재고를 매일, 매주 또는 매월 기준으로 받습니다. 항목의 재고를 받을 때, 재고를 소유하는 상점과 받은 수량에 대한 정보를 기록하는 RECEIPT 테이블에 수량증을 작성합니다. 주문을 처리함에 따라 RECEIPT 테이블을 갱신하여 현재 사용 가능한 재고 레벨을 반영합니다. 수량증 작성에 대한 자세한 정보는 201 페이지의 『WebSphere Commerce에서 재고 자원 작성』을 참조하십시오.

RaDetail

*RaDetail*은 예상 재고 레코드 항목에 대한 추가 정보입니다. 이 정보를 사용하여 서비스 센터에서 재고를 받을 것으로 예상되는 시기를 추정하고, 고객에게 백오더된 항목에 대한 예상 운송 날짜를 제공합니다.

재고

상점에는 서비스 센터와 연관된 재고가 있습니다. 재고에는 서비스 센터에서 실제 고려할 수 있는 모든 것이 포함됩니다. 재고는 한 상점 및 한 서비스 센터와 연관됩니다. 서비스 센터에서 한 상점이 소유한 재고에 대한 정보에는 예약된 수량, 백오더 총계 및 백오더에 할당된 총계와 같은 것도 기록됩니다. 이 정보는 ITEMFFMCTR 테이블에 저장됩니다. 재고 및 재고 자원에 대한 자세한 정보는 197 페이지의 제 21 장 『재고 자원』을 참조하십시오.

운송 협정

상점과 서비스 센터 사이의 마지막 관계는 운송 협정입니다. 운송 협정은 서비스 센터가 운송 방법을 사용하여 상점을 대신하여 상품을 운송할 수 있음을 나타냅니다. 개별 상점은 서비스 센터와 운송 협정을 갖고, 개별 서비스 센터는 상점과 운송 협정을 가집니다. 운송 협정은 SHPARRANGE 테이블에서 설정됩니다. 운송 협정 작성에 자세한 정보는 165 페이지의 『운송 이행 자원 작성』을 참조하십시오.

기타 이행 자원

서비스 센터에는 상점과 직접 관계되지 않는 기타 관계가 있습니다. 오더피킹 일괄처리표는 한 서비스 센터와 연관되는 한 가지입니다. 오더피킹 일괄처리표는 처리할 주문 출고를 서비스 센터에서 한 단위로 함께 그룹화하고, 오더피킹 전표와 출고 전표를 작성합니다. 항목을 오더피킹하고 포장하면, 주문 출고를 운송할 수 있으며 운송을 확인할 수 있습니다. 오더피킹 일괄처리표 정보는 PICKBATCH 테이블에 저장됩니다. 주문 항목도 한 서비스 센터와 연관됩니다. 항목은 속성에 의해 정의되는 상품의 특정 인스턴스입니다. 주문에서 개별 항목에 대한 정보는 ORDERITEM 테이블에 저장됩니다. 주문 자원에 대한 자세한 정보는 203 페이지의 제 22 장 『주문 자원』을 참조하십시오.

다른 엔티티와 마찬가지로 서비스 센터에도 일부 조치를 통제하는 규칙이 있습니다. 개별 서비스 센터는 세금 비용 및 운송 경비에 대한 규칙을 가집니다. 이것은 TAXJCRULE 및 SHPJCRULE 테이블에서 별도로 정의됩니다. 세금 비용 및 운송 경

비에 대한 자세한 정보는 151 페이지의 제 18 장 『운송 자원』, 및 171 페이지의 『WebSphere Commerce에서 세금 자원 이해』를 참조하십시오.



WebSphere Commerce 서버의 이행 자원 구조에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말의 이행 데이터 모델을 참조하십시오.

WebSphere Commerce에서 이행 자원 작성

상점이 고객에게 상품을 운송하기 전에, 이런 상품을 공급할 서비스 센터를 정의해야 합니다. 로더 패키지를 사용하여 데이터베이스에 로드할 수 있는 XML 파일 양식으로 이 정보를 작성하십시오. 로더 패키지에 대한 자세한 정보는 247 페이지의 제 7 부 『상점 공개』를 참조하십시오.

XML 파일을 사용하여 상점의 이행 자원을 작성하려면, 다음을 수행하십시오.

1. 견본 상점의 자원을 작성하는 데 사용된 XML 파일을 검토하십시오. 견본 상점의 모든 파일은 해당 상점 아카이브 파일에 있습니다.

점 아카이브 파일은 다음 디렉토리에 있습니다.

- **NT** drive:\WebSphere\CommerceServer\samplstores
- **2000** drive:\Program Files\WebSphere\CommerceServer\samplstores
- **AIX** /usr/WebSphere/CommerceServer/samplstores
- **Solaris** /opt/WebSphere/CommerceServer/samplstores
- **Linux** /opt/WebSphere/CommerceServer/samplstores
- **400** /qibm/proddata/WebCommerce/samplstores

주: WebSphere Commerce 온라인 도움말에는 견본 상점에 포함된 데이터 자원 각각에 대한 정보가 포함됩니다.

개별 견본 상점에는 이행 정보를 포함하는 fulfillment.xml 파일이 들어 있습니다. 상점 아카이브의 fulfillment.xml 파일을 보려면, ZIP 프로그램을 사용하여 이 파일의 압축을 푸십시오. fulfillment.xml 파일은 데이터 디렉토리에 있습니다.

2. 361 페이지의 부록 B 『데이터 작성』 정보를 검토하십시오.
3. 견본 상점 아카이브의 fulfillment.xml 파일 중 하나를 복사하거나 새 파일을 작성해서 fulfillment.xml 파일을 작성하십시오. 자세한 정보는 fulfillment.xml 에 해당 DTD 파일을 참조하십시오. DTD 파일은 다음 디렉토리에 있습니다.

- **NT** drive:\WebSphere\CommerceServer\xml\sar
- **2000** drive:\Program Files\WebSphere\CommerceServer\xml\sar
- **AIX** /usr/WebSphere/CommerceServer/xml/sar

- **Solaris** /opt/WebSphere/CommerceServer/xml/sar
- **Linux** /opt/WebSphere/CommerceServer/xml/sar
- **400** /qibm/proddata/WebCommerce/xml/sar

4. 상점이 지원하는 센터 또는 서비스 센터를 정의하십시오.

- a. 다음 예를 참조하여 XML 파일의 FFMCENTER 테이블에 서비스 센터를 정의하십시오.

```
<ffmcenter
  ffmcenter_id="@ffmcenter_id_1"
  member_id("&MEMBER_ID"
  name="ToolTech Home"
  defaultshipoffset="0"
  markfordelete="0"
  />
```

여기서,

- ffmcenter_id는 생성된 고유 키입니다.
- member_id는 서비스 센터의 소유자입니다.
- name은 소유자와 함께 서비스 센터를 고유하게 식별하는 문자열입니다.
- defaultshipoffset는 이 서비스 센터에서 한 항목을 운송하는 데 걸리는 추정 시간(초)입니다. 이 값은 STORITMFFC 테이블에서 무시될 수 있습니다.
- markfordelete는 서비스 센터가 다음과 같이 삭제될 지 여부를 나타냅니다. 0 = 삭제되지 않음. 1 = 더 이상 사용하지 않으면 삭제됨. 자세한 정보는 WebSphere Commerce 온라인 도움말의 데이터베이스 정리에 대한 정보를 참조하십시오.

- b. 다음 예를 참조하여 XML 파일의 FFMCENTDS 테이블에 서비스 센터를 설명하십시오. 다국어 지원 상점을 작성하는 경우에는 이 정보를 로케일 고유의 XML 파일에 포함시켜야 합니다.

```
<ffmcentds
  ffmcenter_id="@ffmcenter_id_1"
  description="The fulfillment center that supplies products to ToolTech."
  language_id("&en_US"
  displayname="ToolTech Fulfillment"
  staddress_id="@staddress_id_en_US_1"
  />
```

여기서

- ffmcenter_id는 생성된 고유 키입니다.
- description은 고객에게 적절히 표시할 서비스 센터에 대한 설명입니다.
- language_id는 이 정보가 표시할 언어입니다.
- displayname은 고객에게 적절히 표시할 서비스 센터의 이름입니다.

- staddress_id는 서비스 센터의 실제 위치입니다.
- c. 상점이 지원하는 모든 서비스 센터에 대해 a 및 b단계를 반복하십시오.



@ 및 & 사용에 대한 자세한 정보는 361 페이지의 부록 B 『데이터 작성』을 참조하십시오.

상점 이행 자원 작성

상점의 상품을 공급할 센터 또는 서비스 센터를 정의한 다음에는, 개별 상품에 서비스 센터를 연관시켜야 합니다. 즉, 어떤 서비스 센터가 어떤 상품을 공급할지 식별해야 합니다. 이 관계를 작성하려면 INVENTORY 테이블에 정보를 추가하십시오. 로더 패키지를 사용하여 데이터베이스에 로드할 수 있는 XML 파일 양식으로 이 정보를 작성하십시오. 로더 패키지에 대한 자세한 정보는 247 페이지의 제 7 부 『상점 공개』를 참조하십시오.

주:

1. 상점을 서비스 센터와 연관시키기 전에 상점 자원을 작성해야 합니다. 상점 자원 작성에 대한 자세한 정보는 47 페이지의 『XML 파일에 상점 데이터 자원 작성』을 참조하십시오. 상점 이행 자원을 작성하기 전에 카탈로그 자원도 작성해야 합니다. 자세한 정보는 84 페이지의 『상점 카탈로그 자원 표시』를 참조하십시오.
2. ATP를 이행하지 않고 구현하려는 경우에만 상점 이행 자원을 작성하십시오. INVENTORY 테이블은 ATP 기능을 포함하는 상점에서 사용되지 않습니다.

XML 파일을 사용하여 상점 이행 관계를 작성하려면, 다음을 수행하십시오.

1. 견본 상점의 상점 이행 자원을 작성하는 데 사용된 XML 파일을 검토하십시오. 견본 상점의 모든 파일은 해당 상점 아카이브 파일에 있습니다.



상점 아카이브 파일은 다음 디렉토리에 있습니다.

- **NT** drive:\WebSphere\CommerceServer\samplstores
- **2000** drive:\Program Files\WebSphere\CommerceServer\samplstores
- **AIX** /usr/WebSphere/CommerceServer/samplstores
- **Solaris** /opt/WebSphere/CommerceServer/samplstores
- **Linux** /opt/WebSphere/CommerceServer/samplstores
- **400** /qibm/proddata/WebCommerce/samplstores

주: WebSphere Commerce 온라인 도움말에는 견본 상점에 포함된 데이터 자원 각각에 대한 정보가 포함됩니다.

각각의 견본 상점은 상점 이행 정보가 있는 storefulfill.xml 파일을 포함합니다. 상점 아카이브의 storefulfill.xml 파일을 보려면, ZIP 프로그램을 사용하여 이 파일의 압축을 푸십시오. storefulfill.xml 파일은 데이터 디렉토리에 있습니다.

2. 361 페이지의 부록 B 『데이터 작성』 정보를 검토하십시오.
3. 견본 상점 아카이브의 storefulfill.xml 파일 중 하나를 복사하거나 새 파일을 작성해서 storefulfill.xml 파일을 작성하십시오. 자세한 정보는 storefulfill.xml에 해당 DTD 파일을 참조하십시오. DTD 파일은 다음 디렉토리에 있습니다.

-  NT drive:\WebSphere\CommerceServer\xml\sar
-  2000 drive:\Program Files\WebSphere\CommercServer\xml\sar
-  AIX /usr/WebSphere/CommerceServer/xml/sar
-  Solaris /opt/WebSphere/CommerceServer/xml/sar
-  Linux /opt/WebSphere/CommerceServer/xml/sar
-  400 /qibm/proddata/WebCommerce/xml/sar

4. 다음 예를 참조하여 INVENTORY 테이블에 정보를 추가해서 XML 파일에 상점 서비스 센터 관계를 작성하십시오.

```
<inventory
catentry_id="@catentry_id_1470"
quantity="100"
ffmcenter_id="@ffmcenter_id_1"
store_id="@storeent_id_1"
quantitymeasure="C62"
inventoryflags="0"
/>
```

여기서

- catentry_id는 이 서비스 센터에서 공급할 카탈로그 항목입니다.
- quantity는 이 서비스 센터에서 사용 가능한 QUANTITYMEASURE에서 지정한 단위의 수량 총계입니다.
- ffmcenter_id는 재고를 공급하는 서비스 센터입니다.
- store_id는 재고가 공급되는 상점입니다.
- quantitymeasure는 QUANTITY에 대한 측정 단위입니다.
- inventoryflags는 QUANTITY가 사용되는 방법을 나타내는 비트 플래그입니다.
 - 1 = noUpdate. 기본 UpdateInventory 태스크 명령이 QUANTITY를 갱신하지 않았습니다.

- 2 = noCheck. 기본 CheckInventory 및 UpdateInventory 태스크 명령이 QUANTITY를 확인하지 않았습니다.

5. 상점의 개별 카탈로그 항목에 대해 3단계를 반복하십시오.



@ 및 & 사용에 대한 자세한 정보는 361 페이지의 부록 B 『데이터 작성』을 참조하십시오.

제 12 장 캠페인 자원

캠페인은 마케팅 노력을 구성하기 위해 사용됩니다. 캠페인은 보통 마케팅 관리자나 판매 계획 관리자에 의해 작성됩니다. 캠페인은 종종 특정 목표 세트와 연관됩니다. 예를 들어, "신학기" 캠페인은 캠페인 동안 어린이의 옷 판매를 증가시키는 목적을 가질 수 있습니다.

WebSphere Commerce의 캠페인 이해

WebSphere Commerce에서 캠페인은 조건을 정의하는 몇 가지 캠페인 행사를 포함합니다. 정의된 조건이 true로 확인되면 캠페인 행사는 고객에 대한 대상 콘텐츠를 생성합니다. 결과는 캠페인이 행사를 구성하는 상위 레벨 마케팅 요소라는 것입니다.

캠페인 행사는 행사의 콜렉션이 들어 있는 캠페인과 연관됩니다. 예를 들어 사무 용품 공급 상점이 "신학기" 캠페인을 가진 경우, 행사는 펜 할인을 광고하거나, 학생으로 직업을 등록 및 표시한 고객에게 줄을 친 용지를 제안하는 것과 같은 하위 레벨 조치와 관련됩니다.

캠페인 행사는 세 가지 유형의 동적 콘텐츠를 표시할 수 있습니다.

- 제안 판매 행사
- 유사 그룹 필터링 기반 추천 상품
- 인지 광고

제안 판매 콘텐츠는 고객의 프로필 및 다른 고객의 행위에 근거하여 특정 고객을 대상으로 규칙 기반 카테고리 및 추천 상품을 제공하도록 설계됩니다. 이러한 유형의 콘텐츠를 표시하는 행사는 비슷한 모델 소개와 나은 모델 소개 기회를 만들기 위해 작성됩니다.

유사 그룹 필터링 기반 추천 상품은 추천 상품을 만들기 위한 것이지만, 사전 정의된 규칙보다는 고객의 전체적인 행위에 근거한 항목을 대상으로 하는 다른 추천 상품 알고리즘을 사용합니다.

인지 광고는 제안 판매에 사용되는 동일한 기준에 근거하여 특정 고객을 대상으로 광고 콘텐츠를 제공하도록 설계되지만, 온라인 상점에서 활동에 대한 고객 인지를 증가시키고, 특별 판매 설정을 강조하고 브랜드 인지를 강화시키기 위해 사용됩니다.

행사는 사이트에서 임의 페이지로 결합될 수 있습니다. 사이트 설계시, e-Marketing 지점이라는 특별한 플레이스홀더가 사이트에 배치됩니다. 고객에게 표시될 때, 이 플레이스홀더는 특정 대상 콘텐츠로 바뀝니다. 대상 위치는 원하는 위치에서 e-Marketing 지

점을 표시하도록 행사를 스케줄하여 지정됩니다. 상점에 e-Marketing 지점 추가에 대한 자세한 정보는 351 페이지의 제 32 장 『상점에 e-Marketing 지점 추가』를 참조하십시오.

캠페인 행사에는 캠페인을 표시하는 시기와 대상을 결정하는 조건이 들어 있습니다. 이 조건은 행사가 작성될 때 정의되고 행사의 가시성 및 표시된 콘텐츠를 조정하기 위해 행사의 활동 기간 중 변경될 수 있습니다.

캠페인 행사는 사용에 대한 통계를 생성합니다. 이 통계는 판매자, 마케팅 관리자 및 판매 계획 관리자에 의해 WebSphere Commerce 액셀러레이터를 사용하여 볼 수 있습니다. 통계는 구현되는 각 e-Marketing 지점에 대한 행사의 방문 페이지 기록을 예시합니다. 이 통계는 행사의 효율성에 대한 피드백과, 피드백이 표시하는 다양한 위치 간의 비교 성공 비율을 제공합니다.

WebSphere Commerce의 캠페인 자원 작성

캠페인 및 캠페인 행사는 보통 마케팅 관리자나, WebSphere Commerce 액셀러레이터에서 캠페인과 캠페인 행사 마법사를 사용하는 판매 계획 관리자에 의해 작성됩니다. 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오.

상점에 e-Marketing 지점 추가에 대한 자세한 정보는 351 페이지의 제 32 장 『상점에 e-Marketing 지점 추가』를 참조하십시오.

제 13 장 지불 자원

WebSphere Commerce는 IBM Payment Manager를 지원합니다. 상점의 지불 자원을 작성하려면, 상점에서 Payment Manager를 사용할지 여부를 지정하고, 사용할 경우에는 상점이 승인하는 지불 카세트 유형과 브랜드 유형을 지정하십시오.

이러한 정보를 지정하려면 다음을 수행하십시오.

- 상점 서비스를 사용하여 상점 공개 중 로드되는 XML 파일(paymentinfo.xml)의 양식으로 지불 데이터를 작성하십시오. 이것은 공개 중인 상점에 대해 지정된 판매자 및 브랜드 유형으로 Payment Manager를 구성합니다. 자세한 정보는 『XML 파일을 사용하여 지불 자원 작성』을 참조하십시오.

주: paymentinfo.xml은 WebSphere Commerce 서버 데이터베이스에 테이블을 대량 자료 반입하지 않습니다. Payment Manager를 구성합니다. paymentinfo.xml은 지불 방법으로 오프라인 신용 카드를 사용 중인 경우에만 적용 가능합니다. 다른 지불 방법을 구성하려면, 다음 글머리표를 참조하십시오.







- 관리 콘솔이나 Payment Manager 사용자 인터페이스를 사용하여 상점에 대한 Payment Manager의 설정을 완료하십시오. 관리 콘솔을 사용하는 경우, 메뉴 항목이 Payment Manager 메뉴에 표시됩니다. Payment Manager 사용자 인터페이스를 사용하는 경우, 메뉴 항목이 탐색 프레임의 관리 아래에 표시됩니다. 자세한 정보는 WebSphere Commerce 온라인 도움말의 "상점의 Payment Manager 설정"을 참조하십시오.

XML 파일을 사용하여 지불 자원 작성

XML 파일을 사용하여 상점의 지불 자원을 작성하려면, 다음을 수행하십시오.

1. 견본 상점의 지불 자원을 작성하는 데 사용된 XML 파일을 검토하십시오. 견본 상점의 모든 파일은 해당 상점 아카이브 파일에 있습니다.

상점 아카이브 파일은 다음 디렉토리에 있습니다.

-  NT drive:\WebSphere\CommerceServer\samplstores
-  2000 drive:\Program Files\WebSphere\CommerceServer\samplstores
-  AIX /usr/WebSphere/CommerceServer/samplstores
-  Solaris /opt/WebSphere/CommerceServer/samplstores
-  Linux /opt/WebSphere/CommerceServer/samplstores
-  400 /qibm/proddata/WebCommerce/samplstores

주: WebSphere Commerce 온라인 도움말에는 견본 상점에 포함된 데이터 자원 각각에 대한 정보가 포함됩니다.

개별 견본 상점에는 지불 정보를 포함하는 paymentinfo.xml 파일이 들어 있습니다. 상점 아카이브의 paymentinfo.xml 파일을 보려면, ZIP 프로그램을 사용하여 이 파일의 압축을 푸십시오. paymentinfo.xml 파일은 데이터 디렉토리에 있습니다.

2. 견본 상점 아카이브의 paymentinfo.xml 파일 중 하나를 복사하거나 새 파일을 작성해서 paymentinfo.xml 파일을 작성하십시오. 자세한 정보는 paymentinfo.xml에 해당하는 DTD 파일을 참조하십시오. DTD 파일은 다음 디렉토리에 있습니다.

-  NT drive:\WebSphere\CommerceServer\xml\sar
-  2000 drive:\Program Files\WebSphere\CommerceServer\xml\sar
-  AIX /usr/WebSphere/CommerceServer/xml/sar
-  Solaris /opt/WebSphere/CommerceServer/xml/sar
-  Linux /opt/WebSphere/CommerceServer/xml/sar
-  400 /qibm/proddata/WebCommerce/xml/sar

3. Payment Manager를 사용 또는 사용 안함으로 설정하십시오.

a. 다음 예를 참조하여 XML 파일에서 Payment Manager를 사용하거나 사용하지 않게 하고 상점에서 승인하는 지불 카세트, 통화 및 브랜드의 유형을 지정하십시오.

```
<paymentinfo>
<PaymentManager enable="yes"/>
<Cassette type="OfflineCard">
<Account currency="USD">
<Brand type="MasterCard"/>
<Brand type="VISA"/>
<Brand type="American Express"/>
</Account>
<Account currency="EUR">
<Brand type="MasterCard"/>
<Brand type="VISA"/>
<Brand type="American Express"/>
</Account>
</Cassette>
</paymentinfo>
```

여기서

- enable은 Payment Manager를 사용할지 아니면 사용하지 않을지의 여부입니다.
- Cassette type은 지원되는 카세트의 유형입니다.

- Account currency는 상점이 지원하는 통화입니다. 계정 통화는 OfflineCard 카세트 유형을 사용 중인 경우 필요합니다. 통화는 ISO 4217 표준과 일치하는 세 문자 코드로 식별되어야 합니다. 예를 들어, 미국 달러의 경우 "USD"입니다.
- Brand type은 계정과 통화에서 지원되는 신용 카드의 유형입니다.

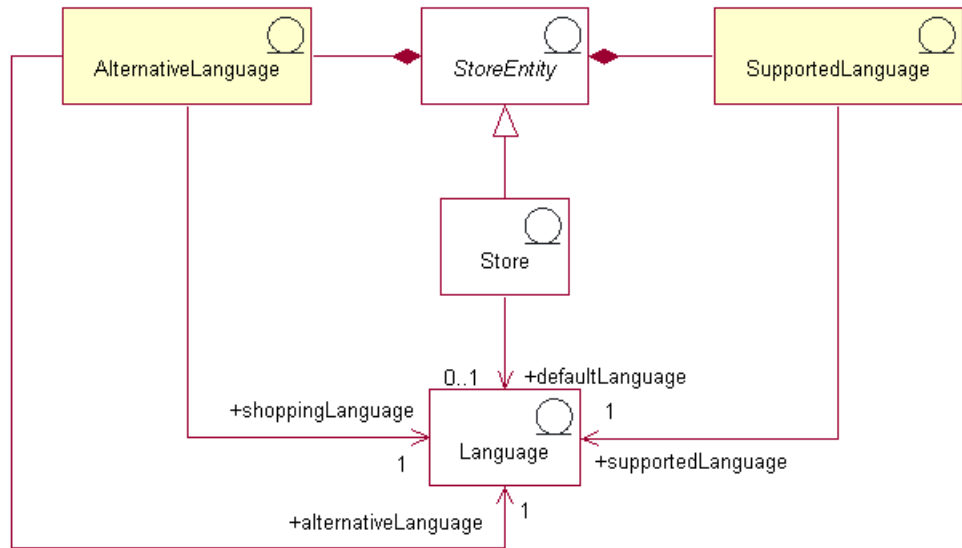
제 14 장 언어 지원

WebSphere Commerce에서, 사용자 사이트는 그 안에서 사용할 수 있는 여러 언어를 정의할 수 있습니다. LANGUAGE 테이블에는 독일어, 대만어, 중국어, 일본어, 한국어, 이태리어, 프랑스어, 스페인어, 포르투갈어 및 영어를 포함하여 10개의 언어가 들어 있습니다. 사이트에서 추가 언어나 기존 언어의 방언을 정의하여 다른 문화나 인적사항에서 고객에게 정보를 표시하는 방법을 조정할 수 있습니다.

WebSphere Commerce의 언어 자원 이해

언어 자원을 이해하려면, 언어와 상점간의 관계를 이해해야 합니다. 이것은 아래의 정보 모델을 사용하여 설명할 수 있습니다. 다음 절은 상점과 다른 자원들에 대해 관련 언어가 갖고 있는 관계에 대해 설명합니다.

아래의 도표는 언어 자원 정보 모델을 설명합니다.



WebSphere Commerce에는 4개의 언어 분류가 있습니다. 이것은 기본 언어, 지원 언어, 대체 언어 및 구매 언어입니다. 이렇게 분류된 각 언어는 상점에서 서로 다른 역할을 수행합니다. 모든 언어는 LANGUAGE 테이블에 저장됩니다.

기본 언어

기본 언어는 각 상점과 연관됩니다. 이것은 주 언어로 사용하기 위해 상점이 선택한 언어이며, 구매 언어를 명시적으로 선택하지 않은 고객에게 표시되는 언어입니다. 상점에 대한 기본 언어는 상점에서 묵시적으로 지원되므로, 상점은 항상 기본 언어나,

LANGPAIR 테이블에 정의된 대체 언어 중 하나로 정보를 표시할 수 있어야 합니다. 정보가 지원되는 언어나 대체 언어 중 하나로 사용 가능하지 않으면, 정보는 기본 언어로 표시됩니다.

지원 언어

STORELANG 테이블은 각 상점에서 지원하는 언어를 나타냅니다. 상점은 지원되는 언어나 LANGPAIR 테이블에 정의된 대체 언어 중 하나로 정보를 표시할 수 있어야 합니다. 또한 상점은 상점 그룹에서 지원하는 모든 언어를 지원합니다.

대체 언어

지원되는 언어 중 하나로 정보를 사용할 수 없을 경우, 상점은 사용 가능한 경우 대체 언어로 정보를 표시하려고 합니다. 상점은 각 대체 언어에 시도하려는 순서를 지정할 수 있습니다. 상점의 대체 언어는 상점 그룹에 대한 대체 언어를 포함합니다. 일부 정보가 오직 한 가지 언어에서 사용 가능할 때 대체 언어는 유용할 수 있지만, 관련된 다른 언어로 고객이 구매할 수 있어야 합니다. 이것은 모든 정보가 지원되는 모든 언어로 아직 변환되지 않았거나 동일한 정보를 갖는 동일한 언어의 매우 유사한 두 개의 방언이 지원되는 경우입니다.



WebSphere Commerce 서버의 언어 자원 구조에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말의 언어 오브젝트 및 데이터 모델을 참조하십시오.

WebSphere Commerce에서 언어 자원 작성

상점이 다음 방법 중 하나를 지원하는 언어를 정의할 수 있습니다.

- 상점 서비스의 도구 사용
- 로더 패키지 또는 상점 서비스의 공개 도구가 로드할 XML 파일

주: 상점 서비스 도구는 상점 아카이브 포맷으로 미리 저장된 XML 파일과 함께 작업할 수 있습니다.

상점 서비스를 사용한 상점 지원 언어 정의에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오. XML 파일의 상점 지원 언어 정의에 대한 자세한 정보는 47 페이지의 『XML 파일에 상점 데이터 자원 작성』을 참조하십시오.

제 15 장 통화 지원

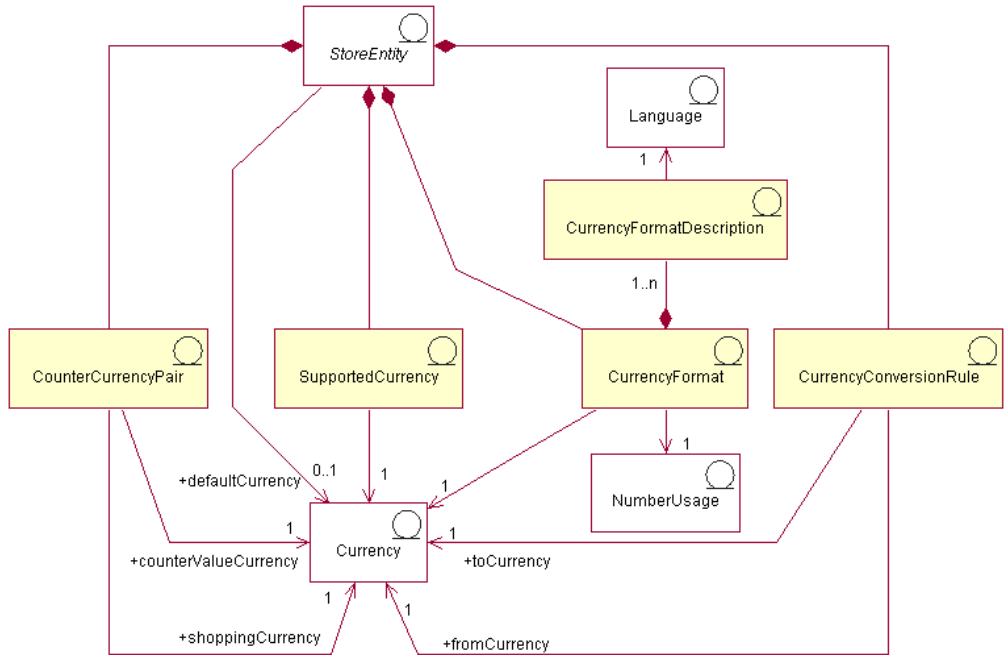
사이트에 하나의 통화로 가격을 표시하거나, 유로화를 제공한 지침에 따라 여러 가지 통화를 사용할 수 있습니다. 하나의 사이트에 여러 개의 상점이 있는 경우, 상점 별로 다른 통화를 사용하거나 상점 그룹에 통화를 지정할 수 있습니다. 작성하는 사이트의 본질에 따라, 사용하고자 하는 통화와 통화 표시 방법을 지정할 수 있습니다.

WebSphere Commerce에서 고객은 구매 통화를 선택할 수 있습니다. 구매 통화는 고객이 특정 상점에서 상품에 대해 지불하는 통화입니다. 상점 페이지의 모든 통화 총계는 이 통화로 표시됩니다. 고객이 자신들의 구매 통화를 변경할 때, 고객이 장바구니에 추가한 항목에 대한 가격과 주문 총계는 자동으로 새 구매 통화로 변환되고 다시 계산되어 표시됩니다.

고객은 유로화를 포함해 여러 통화로 구매할 수 있습니다. 유로화는 1999년 1월 1일 EU의 공식 통화가 되었으며, 현재 금융 시장에서 사용되고 있습니다. 유로화와 모든 참여 국가의 통화 간의 변환 비율은 고정 가격입니다. EU의 미래 단일 통화에 대한 지폐 및 동전은 2002년 1월 1일부터 사용 가능합니다. 6개월 후면 기존의 국가별 통화는 영구적으로 유통되지 않습니다. 1999년부터 2001년까지의 과도기 동안 판매자는 국가별 통화와 유로화를 모두 받아야 합니다.

WebSphere Commerce의 통화 지원 이해

다음 도표는 WebSphere Commerce 서버의 통화 구조를 보여줍니다.



이 도표와 상점 데이터 질의 모든 도표들은 WebSphere Commerce 서버 정보 모델의 부분입니다. 정보 모델에 대한 자세한 정보는 29 페이지의 『상점 데이터 정보 모델』을 참조하십시오. 이 도표에서 사용된 규칙에 대한 자세한 정보는 359 페이지의 부록 A 『UML 범례』를 참조하십시오.

위의 도표에서 통화는 정보 모델의 중앙에 있습니다. 각 상점 또는 상점들의 그룹은 기본 통화를 가지고 있습니다.

통화 포맷

상점 엔티티는 여러 통화 포매팅 규칙을 가질 수 있습니다. 하나의 상점이 하나의 특정 통화에 대한 포매팅 규칙을 가지고 있지 않다면, 그 상점은 상점 그룹의 포매팅 규칙을 사용합니다. 통화 포맷은 CURFORMAT 테이블에 설정되어 있습니다.

숫자 사용법

포맷된 각 통화 규칙은 하나의 숫자 사용법과 연관되어 있습니다. 수량 및 통화량과 같은 숫자는 연관된 사용법에 따라 다르게 포맷되고 반올림될 수 있습니다. 상점들은 숫자 사용법에 따라 표시하는 숫자에 대해 서로 다른 반올림 및 포매팅 규칙을 지정할 수 있습니다. 그래서 한 상점에서는 단위 가격 사용법을 지정하여 소수점 네 번째 자리로 단위 가격을 반올림하지만, 다른 상점에서는 기본 사용법을 지정하여 소수점 두 번째 자리로 통화 총계를 반올림합니다. 숫자 사용법은 NUMBRUSG 테이블에 저장됩니다.

통화 포맷 설명

통화 포맷 규칙은 여러 통화 포맷 설명을 가질 수 있습니다. 통화 포맷 설명은 특정 언어에서 특정 통화의 통화 총계를 포맷하는 방법(목적 표시)을 설명합니다. 각각의 설명은 LANGUAGE 테이블의 언어와 연관되어 있습니다. 언어 자원에 관한 자세한 정보는 135 페이지의 제 14 장 『언어 자원』을 참조하십시오. 통화 포맷 설명은 CURFMTDESC 테이블에 저장됩니다.

지원 통화

상점 엔티티는 여러 지원 통화를 가질 수 있습니다. 지원 통화는 지불이 승인된 통화입니다.

통화 변환 규칙

모든 통화는 다른 통화에서 다른 통화로 변환시 통제하는 규칙을 가지고 있습니다. 각 통화 변환 규칙은 고객이 지원 구매 통화로 지불해야 할 총계를 데이터베이스에서 특정 통화로 저장된 가격으로 환산할 때 사용할 수 있습니다.

카운터 통화

카운터 통화는 지원되는 통화와 함께 표시되는 통화 총계입니다. 구매용으로는 사용될 수 없으며 정보용으로 사용됩니다. 고객이 유로화로 구매하기로 결정한 경우, 고객은 European Monetary Union 통화 총계를 볼 수 있으며, 다른 통화 총계가 상점에 표시되기도 합니다. 구매 통화 총계는 해당 구매 통화에 대한 모든 카운터 값 통화로 변환됩니다. 카운터 통화는 네덜란드 길더, 유로화와 같이 지원 통화와 쌍을 이룹니다. 카운터 통화 쌍은 CURCVLIST 테이블에 저장됩니다



WebSphere Commerce 서버의 통화 지원 구조에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말의 통화 데이터 모델을 참조하십시오.

WebSphere Commerce에서 통화 자원 작성

WebSphere Commerce의 상점 서비스 도구를 사용하여 지원되는 통화를 상점에 추가하고 상점에 대한 기본 통화를 선택할 수 있습니다. 상점 서비스 도구를 사용하여 편집할 수 있는 자원에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말 주제 "상점 데이터베이스 자원 변경"을 참조하십시오.

주: 상점 서비스 도구는 상점 아카이브 양식의 사전 대량 자료 반입된 XML 파일을 사용하여 작업합니다.

로더 패키지를 사용하여 데이터베이스에 로드할 수 있는 XML 파일을 사용하여 지원되는 통화 및 기본 통화를 상점에 추가할 수도 있습니다. 이 방법은 정의하는 통화 변환 비율, 카운터 값 통화를 포함하여 통화 자원의 다른 유형을 작성하도록 허용합니다.

기존 상점 아카이브의 통화 자원 편집에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오. XML 파일 양식으로 새 통화 자원 작성에 대한 자세한 정보는 『XML 파일을 사용하여 통화 자원 작성』을 참조하십시오.






XML 파일을 사용하여 통화 자원 작성

로더 패키지를 사용하여 데이터베이스에 로드할 수 있는 XML 파일 양식으로 상점의 통화 자원을 작성하십시오. 로더 패키지에 대한 자세한 정보는 247 페이지의 제 7 부 『상점 공개』를 참조하십시오.

XML 파일을 사용하여 상점의 통화 자원을 작성하려면 다음과 같이 하십시오.

1. 견본 상점의 통화 자원을 작성하는 데 사용된 XML 파일을 검토하십시오. 견본 상점의 모든 파일들은 해당 상점 아카이브 파일에 있습니다.

상점 아카이브 파일들은 다음 디렉토리에 있습니다.

-  NT drive:\WebSphere\CommerceServer\samplstores
-  2000 drive:\Program Files\WebSphere\CommerceServer\samplstores
-  AIX /usr/WebSphere/CommerceServer/samplstores
-  Solaris /opt/WebSphere/CommerceServer/samplstores
-  Linux /opt/WebSphere/CommerceServer/samplstores
-  400 /qibm/proddata/WebCommerce/samplstores

주: WebSphere Commerce 온라인 도움말은 견본 상점에 포함된 각 데이터 자원에 대한 정보를 포함합니다.

각 견본 상점은 통화 정보가 들어 있는 currency.xml 파일을 갖고 있습니다. 상점 아카이브에서 currency.xml 파일을 보려면 ZIP 프로그램을 사용하여 파일을 압축 해제하십시오. currency.xml 파일들은 데이터 디렉토리에 있습니다.

2. 361 페이지의 부록 B 『데이터 작성』 정보를 검토하십시오.
3. 견본 상점 아카이브에서 currency.xml 파일들 중 하나를 복사하거나 새 파일을 작성하여 currency.xml 파일을 작성하십시오. 자세한 정보는 currency.xml에 대한 해당 DTD 파일을 참조하십시오. DTD 파일은 다음 디렉토리에 있습니다.

-  NT drive:\WebSphere\CommerceServer\xml\sar
-  2000 drive:\Program Files\WebSphere\CommerceServer\xml\sar
-  AIX /usr/WebSphere/CommerceServer/xml/sar
-  Solaris /opt/WebSphere/CommercServer/xml/sar
-  Linux /opt/WebSphere/CommercServer/xml/sar
-  400 /qibm/proddata/WebCommerce/xml/sar

4. 상점이 지원하는 통화를 정의하십시오.

- a. 다음 예를 사용하여 CURLIST 테이블에 대한 XML 파일에서 상점이 지원하는 통화를 정의해야 합니다.

```
<curlist currstr="USD" storeent_id="@storeent_id_1" />
```

여기서,

- currstr은 지원되는 통화를 나타내는 세 문자 ISO 4217 통화 코드입니다. 이 코드는 SETCURR 테이블의 SETCCURR 열에 표시되어야 합니다. 상점은 모든 지원 통화로 지불을 승인할 수 있어야 합니다.
- storeent_id는 상점 엔티티입니다.

- b. 상점이 지원하는 각 통화를 반복하십시오.



상점의 기본 통화는 STOREENT 테이블에 정의됩니다. 자세한 정보는 47 페이지의 『XML 파일에 상점 데이터 자원 작성』을 참조하십시오.

5. (선택적) 상점의 통화 가격이 표시되는 방식은 가격 설정 방식에 의해 좌우됩니다. 상점에서 사용되는 모든 통화의 가격을 정의하거나 기본 통화에 대해서만 가격을 정의할 수 있습니다. 가격 설정에 대한 자세한 정보는 103 페이지의 『WebSphere Commerce에서 가격 책정 자원 작성』을 참조하십시오.

가격 설정시, 기본 통화에 대해서만 가격을 정의하고 지원되는 다른 통화로 상점 가격을 표시하려는 경우, 상점에 대한 변환 비율을 추가해야 합니다. 이 변환 비율을 사용하여 기본 통화에서 지원되는 통화로 변환하십시오.

- a. 변환될 원본 통화(예: 미국 달러(USD))를 결정하고, 변환하려는 대상 통화(예: 엔(JPY))를 결정하십시오. 각 통화에 대한 ISO 통화를 결정하려면, 국제 통화 에 대해 ISO 4217 코드를 참조하십시오.
- b. 다음 예를 사용하여, 변환 정보를 CURCONVERT 테이블에 추가하십시오.

```
<curconvert  
storeent_id="@storeent_id_1"  
fromcurr="USD"  
tocurr="JPY"  
factor="105.10"  
multiplyordivide="M"  
bidirectional="Y"  
updatable="Y"  
curconvert_id="@curconvert_id_1" />
```

여기서,

- storeent_id는 상점 엔티티입니다.
- fromcurr은 변환하려는 원본 통화입니다. FROMCURR 통화의 총계는 보통 판매 설정된 상품과 연관된 유사한 총계, 운송 경비, 할인, 가격을 결정하는 데 사용되는 규칙이나 기타 정보의 파트입니다.

- tocurr은 변환하려는 대상 통화입니다. TOCURR은 보통 고객이 지불하려는 통화입니다. 이 통화의 총계는 단가, 운송 경비 또는 세금 총계와 같은 주문 항목 부분입니다.
- factor는 변환 인수입니다.
- multiplyordivide는 다음과 같습니다. FROMCURR에서 TOCURR로 변환하려면 다음과 같이 하십시오.
 - M = FACTOR로 곱하기
 - D = FACTOR로 나누기

양방향 규칙의 경우, TOCURR에서 FROMCURR로의 변환은 역조작을 사용해야 합니다.
- bidirectional은 규칙이 양방향인지 아니면 단일 방향인지 여부를 나타냅니다.
 - Y = 양방향
 - N = 단일 방향
- updatable은 통화 변환 규칙을 관리하는 사용자 인터페이스에서 사용하려는 플래그입니다. 올바른 값은 다음과 같습니다.
 - N = 변환 비율을 취소할 수 없음 - 변경할 수 없음
 - Y = 변환 비율을 변경할 수 있음
- curconvert_id는 생성된 고유 키입니다.

c. 가격을 표시하려는 모든 통화에 대해 a 및 b단계를 반복하십시오.



가격 정보에서 지원되는 모든 통화에 대해 가격을 정의했어도, 상점에서 지원되는 통화에 대한 통화 변환 비율을 정의하려고 할 수 있습니다.

6. (선택적) 구매 통화와 카운터 통화(예를 들어, 네덜란드 길더와 유로 둘 다 가격 표시) 둘 다 표시 가격이 있는 경우, 정보를 CURCVLIST 테이블에 추가해야 합니다.

a. 다음 예를 사용하여 변환 정보를 CURCVLIST 테이블에 추가하십시오.

```
<curcvlist
storeent_id="@storeent_id_1"
currstr="NLG"
countervaluecurr="EUR"
displayseq="1" />
```

여기서,

- storeent_id는 상점 엔티티입니다.
- currstr은 통화를 나타내는 세 문자 ISO 4217 통화 코드입니다. 이 코드는 SETCURRE 테이블의 SETCCURRE 열에 표시되어야 하며 변환하려는 원

본 통화입니다. FROMCURR 통화의 총계는 보통 판매 설정된 상품과 연관된 유사한 총계, 운송 경비, 할인, 가격을 결정하는 데 사용되는 규칙이나 기타 정보의 파트입니다.

- countervaluecurr은 카운터 값 통화를 나타내는 세 문자 ISO 4217 통화 코드입니다. 이 코드는 SETCURR 테이블의 SETCCURR 열에 표시되어야 합니다.
- displayseq는 카운터 값 통화의 표시 순서를 나타내는 숫자입니다. 카운터 값 통화는 CURCVLIST 테이블에서 DISPLAYSEQ에 지정된 카운터 값 표시 순서에 근거한 오름차순으로 표시됩니다.



@ 및 &의 사용에 대한 자세한 정보는 361 페이지의 부록 B 『데이터 작성』을 참조하십시오.

기타 통화 태스크

일반적으로 통화와 다른 통화 태스크에 대한 자세한 정보는 다음을 포함합니다.

- WebSphere Commerce에 의해 현재 지원되지 않는 새 통화 추가
- 기존 통화 포맷 변경
- 새 통화나 특정 상점에 대한 새 포맷 규칙 추가

WebSphere Commerce 온라인 도움말을 참조하십시오.

제 16 장 측정 자원 단위

킬로그램, 인치, 리터 등의 다양한 수량 단위로 상품을 판매할 수 있고, 재고를 추적할 수 있습니다. 이런 단위 중에서 상품을 최소 수량 및 여러 특정 수량으로 주문할 수 있습니다.

제어기 명령은 UOM(측정 단위)을 사용하여 수량 단위를 지정합니다. UOM 매개변수를 지정하지 않으면, 고객이 지정한 수량은 CATENTSHIP 데이터베이스 테이블에 있는 카탈로그 항목의 기본 수량으로 곱해집니다. 그 결과를 요청한 수량으로 알게 됩니다.

요청 수량은 반올림하여 카탈로그 항목의 수량 배수의 다음 값 중 가장 높은 값으로 합니다. 예를 들어, 곱이 2kg이고 요청 수량이 4.1kg이면, 반올림 결과는 6kg이 됩니다. 반올림한 수량은 자체 수량 단위가 있는 재고를 확인할 때 사용됩니다. 재고 수량 단위와 카탈로그 항목 수량 단위가 다르면, 두 단위를 변환해야 합니다.

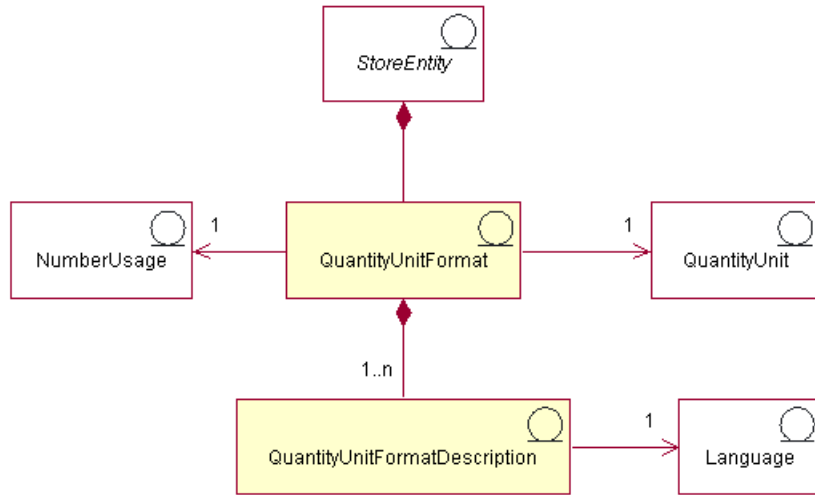
ATP 재고가 사용될 때(STORE 테이블의 ALLOCATIONGOODFOR 열 참조), 재고 수량 단위는 BASEITEM 테이블의 QUANTITYMEASURE 열에서 정의된 수량 단위입니다. 그렇지 않으면, 이것은 INVENTORY 테이블의 QUANTITYMEASURE 열에서 정의됩니다.

카탈로그 항목의 기본 수량으로 나누어 반올림한 수량이 정상 수량으로 알려져 있습니다. 정상 수량은 실행하는 명령에 따라 주문 항목이나 이해 항목에 저장됩니다. 예를 들어, 반올림한 수량이 6kg이고 기본 수량이 2kg이면, 정상 수량은 3입니다.

카탈로그 항목의 판매 설정을 찾을 때, 요청 수량이 판매 설정이 주는 최선의 가격에 영향을 미칠 수 있으므로 어떤 판매 설정을 사용해야 할지 결정해야 합니다. 예를 들어, 반올림한 수량이 6kg이고 기본 수량 2kg에 대해 가격을 \$4.00로 최소 수량을 10kg으로 지정한 판매 설정과 기본 수량 2kg에 대해 가격을 \$4.50로 최소 수량을 2kg으로 지정한 다른 판매 설정으로 된 두 개의 판매 설정이 있는 경우, 두 번째 판매 설정만 사용할 수 있습니다.

WebSphere Commerce에서 측정 단위 이해

다음 도표는 WebSphere Commerce 서버에서의 측정 단위 구조를 보여줍니다.



이 도표 및 상점 데이터 절의 기타 모두는 WebSphere Commerce 서버 정보 모델의 일부입니다. 정보 모델에 대한 자세한 정보는 29 페이지의 『상점 데이터 정보 모델』을 참조하십시오. 이 도표에 사용된 규칙에 대한 자세한 정보는 359 페이지의 부록 A 『UML 범례』를 참조하십시오.

수량 단위 및 수량 단위 포맷

수량 단위는 상점에서 사용하는 측정 단위입니다(예: 킬로그램, 파운드, 미터, 인치 및 리터 등). 수량 단위 포맷은 상점에서 이 수량 단위를 포맷한 방법입니다(예를 들면 수량 단위를 표현할 때, 사용된 10진수 자리의 수입니다).

개별 수량 단위 포맷은 한 상점 엔티티만의 일부이지만, 개별 상점 엔티티는 여러 수량 단위 포맷을 가질 수도 있습니다.

수량 단위 포맷은 개별 수량 단위와 번호 사용법을 가지며, 상점이 지원하는 언어의 수에 따라 하나 이상의 수량 단위 포맷 설명을 가질 수 있습니다.

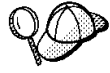
수량 단위 포맷 설명

수량 단위 포맷 설명은 특정 언어에서 특정 수량 단위의 수량 총계를 포맷하는 방법(목적 표시)을 설명합니다.

번호 사용법

번호 사용법은 한 응용프로그램에서 번호를 사용하는 방법을 정의합니다. 예를 들어, WebSphere Commerce 코드의 번호 사용법 코드를 사용해서, 해당 번호(통화나 수량)를 포맷하거나 반올림하고자 하는 방식을 선택할 수 있습니다. 이런 코드(NUMBRUSG 테이블에 정의됨)를 사용하면 CURFORMAT, CURFMTDESC, QTYFORMAT 및

QTYFMTDESC 테이블에서 번호 사용법의 해당 유형에 대해 지정된 규칙에 따라 번호를 포맷할 수 있습니다. 이는 상점이 다양한 상황의 요구사항에 부합하여 다른 방식으로 번호 수를 포맷할 수 있게 합니다.



WebSphere Commerce 서버의 측정 단위 자원의 구조에 대한 자세한 정보는
WebSphere Commerce 온라인 도움말의 수량 단위 데이터 모델을 참조하십시오.

WebSphere Commerce에서 측정 단위 작성

인스턴스가 작성될 때, 측정 단위는 WebSphere Commerce 서버 데이터베이스에 사전에 대량 자료 반입됩니다. 자세한 정보는 41 페이지의 제 5 장 『사이트 자원』을 참조하십시오.

상점에 사용할 새 측정 단위를 WebSphere Commerce에 정의하거나 상점에서 사용하지 않으려는 측정 단위를 삭제할 수 있습니다.

상점에서 사용할 새 측정 단위를 정의하려면, 다음 데이터베이스 테이블에 정보를 추가하십시오.

- QTYUNIT
- QTUNITDSC
- QTYFORMAT
- QTYFMTDESC
- QTYUNITMAP
- QTYCONVERT

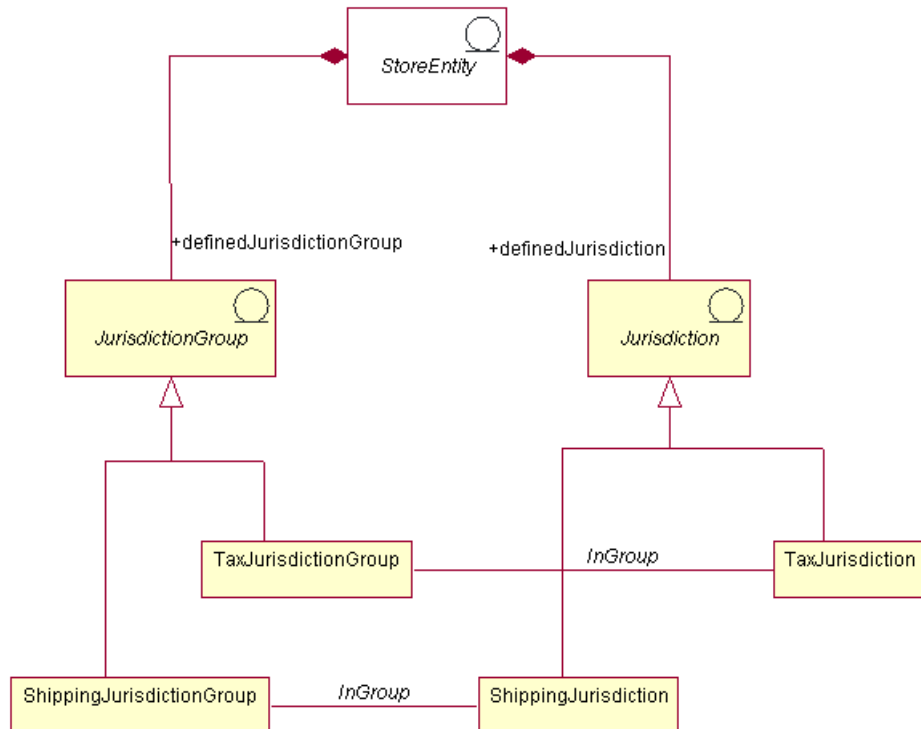
제 17 장 관할 구역 지원

관할 구역은 상품을 판매하는 지리적인 지역 또는 국가나 지역, 지방이나 영토를 나타내는 지역 또는 우편번호 범위입니다. 관할 구역은 함께 그룹화되어 관할 구역 그룹을 형성할 수 있습니다.

관할 구역 그룹은 주문에 대한 운송 경비 및 세금 비용을 계산하는 데 사용됩니다. 즉, 관할 구역 그룹은 사용되는 운송 경비 및 세금 계산을 규정하기 위해 사용할 수 있습니다. 이 규정된 계산 규칙은 계산 규칙과 연관된 관할 구역 그룹에서 관할 구역 중 하나의 주소로 항목을 운송 중인 경우에만 주문 항목에 적용 가능합니다. 결과적으로, 운송 경비와 세금 총계는 주문에서 다른 항목에 대한 운송 주소에 따라 다르게 계산될 수 있습니다.

WebSphere Commerce의 관할 구역 지원 이해

다음 도표는 관할 구역과 관할 구역 그룹을 WebSphere Commerce 서버에 최적화하는 방법을 보여줍니다.





이 도표와 상점 데이터 절의 모든 도표는 WebSphere Commerce 서버 정보 모델의 부분입니다. 정보 모델에 대한 자세한 정보는 29 페이지의 『상점 데이터 정보 모델』을 참조하십시오. 이 도표에서 사용된 규칙에 대한 자세한 정보는 359 페이지의 부록 A 『UML 범례』를 참조하십시오.

WebSphere Commerce에서 관할 구역이나 관할 구역 그룹은 상점의 부분이며, 관할 구역이 작성되는 상점이나 상점 그룹에 독점적입니다. 예를 들어, 상점에 대해 세 개의 관할 구역을 작성한 다음 상점을 삭제하면 관할 구역도 삭제됩니다. 삭제된 관할 구역은 기존의 다른 상점 또는 앞으로 작성할 상점이 사용할 수 없습니다.

그러나 상점 그룹에 대한 관할 구역을 작성하는 경우, 해당 그룹에 있는 상점이 삭제될 때 관할 구역은 삭제되지 않습니다. 해당 상점 그룹에서 작성된 새로운 상점에 대해 관할 구역이 사용 가능합니다.

WebSphere Commerce는 운송 관할 구역 및 세금 관할 구역이라는 두 가지 유형의 관할 구역을 지원합니다. 운송 관할 구역은 함께 그룹화되어 운송 경비 계산 규칙을 규정하는 운송 관할 구역 그룹을 형성할 수 있습니다. 유사하게 세금 관할 구역은 함께 그룹화되어 세금 계산 규칙을 규정하는 세금 관할 구역 그룹을 형성할 수 있습니다.



WebSphere Commerce 서버의 관할 구역 자원의 구조에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말의 관할 구역 데이터 모델을 참조하십시오.

WebSphere Commerce에서 관할 구역 자원 작성

세금 및 운송 경비를 적용하기 위해 상점에 대한 관할 구역 자원을 작성해야 합니다. 관할 구역 작성에 대한 자세한 정보는 174 페이지의 『WebSphere Commerce에서 세금 자원 작성』 또는 153 페이지의 『WebSphere Commerce에서 운송 자원 작성』을 참조하십시오.

일단 상점에 대한 관할 구역이 작성되면, 상점 서비스의 세금 및 운송 노트북을 사용하여 관할 구역을 편집하거나 새로운 관할 구역을 작성할 수 있습니다.

주: 상점 서비스는 작성하는 모든 관할 구역에 대한 관할 구역 그룹을 자동으로 작성합니다. 상점 서비스는 상점에 대한 관할 구역을 작성하지만 상점 그룹에 대해서는 작성하지 않습니다.

주: 상점 서비스는 상점 아카이브의 형태로 미리 공개된 데이터에 대해서만 작업할 수 있음에 주의하십시오.



이 도표 및 상점 데이터 절의 기타 모두는 WebSphere Commerce 서버 정보 모델의 일부입니다. 정보 모델에 대한 자세한 정보는 29 페이지의 『상점 데이터 정보 모델』을 참조하십시오. 이 도표에서 사용된 규칙에 대한 자세한 정보는 359 페이지의 부록 A 『UML 범례』를 참조하십시오.

운송 방법

운송 모드는 상품을 운송하는 방법입니다. 특히 운송 방법은 운송 회사(서비스 센터에서 고객에게로 운송 서비스를 제공하는 회사) 및 해당 운송 회사에서 제공하는 운송 서비스의 조합입니다. 예를 들어, ABC 운송 회사, 운송 회사, 야간 서비스 및 ABC 운송 회사, 특급 배송은 운송 방법입니다.

운송 방법은 상점 엔티티에 속합니다. 상점 엔티티가 삭제된 경우, 해당 상점 엔티티 내에 정의된 운송 방법도 삭제됩니다. 상점은 기본값 운송 모드를 갖도록 요구되지 않지만, 운송 모드를 갖는 것이 좋습니다.

운송 협정

운송 협정은 상점과 서비스 센터 간의 준비로 서비스 센터에서 지정된 운송 방법을 사용하여 특정 상점의 상품을 운송할 것을 나타냅니다. 운송 협정이 유효한 기간과 운송 협정이 이루어지는 운송 관할 구역을 포함하는 특정 제한이 운송 협정에 적용될 수 있습니다.

운송 협정이 운송 방법과 연관되는 경우, 해당 운송 방법에만 적용됩니다. 그렇지 않으면, 운송 협정은 사용 가능한 모든 운송 방법에 적용됩니다. 운송 협정은 상점의 일부 분으로서 상점을 삭제하면 삭제됩니다.

계산 코드

계산 코드는 운송 비용을 계산하는 데 사용됩니다. 즉, 운송 계산 코드는 주문 항목의 운송을 계산하는 방법을 나타냅니다. 주문 항목에 대한 운송 경비를 계산하려면, 카탈로그 항목이나 카탈로그 항목의 그룹 중 하나에 운송 계산 코드를 지정해야 합니다.

계산 코드는 상점 엔티티의 일부입니다. 계산 코드는 한 상점 엔티티와만 연관될 수 있지만, 상점 엔티티에는 여러 계산 코드가 있을 수 있습니다. 상점 엔티티가 삭제되면, 해당 상점 엔티티와 연관된 계산 코드도 삭제됩니다.



계산 코드에 관한 자세한 정보는 IBM WebSphere Commerce 계산 프레임워크 안내서를 참조하십시오.

계산 규칙

각 계산 코드는 계산 규칙 세트를 가집니다. 주문 항목에 대한 운송 경비는 운송 방법, 서비스 센터 및 운송 관할 구역에 따라 다를 수 있습니다.

ShippingJurisdictionGroupCalculationRules은 각 주문 항목에 대해 어떤 계산 규칙을

사용해야 하는지 여부를 결정하기 위해 운송 계산 규칙을 관할 구역, 서비스 센터 및 운송 방법과 연관시키는 관계 오브젝트입니다.

계산 규칙이나 ShippingJurisdictionGroupCalculationRules에 의해 참조되는 다른 오브젝트 중 하나가 삭제되면, ShippingJurisdictionGroupCalculation 규칙도 삭제됩니다.



계산 규칙에 관한 자세한 정보는 *IBM WebSphere Commerce 계산 프레임워크* 안내서를 참조하십시오.

관할 구역 및 관할 구역 그룹

관할 구역은 상품을 판매하는 지리적인 지역 또는 국가나 지역, 지방이나 영토를 나타내는 지역 또는 우편번호 범위입니다. 관할 구역들은 함께 그룹으로 묶여 관할 구역 그룹을 형성합니다.

WebSphere Commerce는 두 가지 유형의 관할 구역 즉, 운송 관할 구역 및 세금 관할 구역을 지원합니다. 이런 개별 관할 구역은 해당 그룹의 일부분입니다. 예를 들어, 운송 관할 구역은 운송 관할 구역 그룹에 속하고 세금 관할 구역은 세금 관할 구역 그룹에 속합니다.

관할 구역 그룹은 계산 규칙과 연관됩니다. 계산 규칙은 운송 경비 총계를 결정하는 계산의 일부분으로서 관할 구역 그룹을 사용합니다.

관할 구역 및 관할 구역 그룹은 상점 엔티티의 일부분입니다. 상점 엔티티를 삭제하면, 해당 상점과 연관된 관할 구역 및 관할 구역 그룹도 삭제됩니다.

하나의 운송 주소는 여러 운송 관할 구역으로 분해될 수 있습니다. 예를 들어, 미국 뉴욕의 운송 주소는 "미국, 뉴욕", "미국" 및 "세계"의 운송 관할 구역으로 적용됩니다. 운송 주소가 여러 운송 관할 구역에 적용되면, 여러 운송 계산 규칙을 적용할 수 있습니다. 그러한 경우, 연관된 ShippingJurisdictionGroupCalculationRules의 우선순위는 어떤 규칙이 사용될지를 결정하는 데 사용됩니다.



WebSphere Commerce 서버의 운송 자원 구조에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말의 세금 오브젝트 및 데이터 모델을 참조하십시오.

WebSphere Commerce에서 운송 자원 작성

WebSphere Commerce의 상점 서비스 도구를 사용하면, 상점 이카이브의 특정 운송 자원(예: 운송 방법 및 관할 구역)을 작성하고 편집할 수 있지만, 모든 운송 자원을 작성하고 편집할 수는 없습니다. 상점 서비스 도구를 사용하여 편집할 수 있는 자원에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말 주제 "상점 데이터베이스 자원 변경"을 참조하십시오.

주: 상점 서비스 도구는 상점 아카이브 양식의 사전 대량 자료 반입된 XML 파일을 사용하여 작업합니다.

로더 패키지를 사용하여 데이터베이스에 로드할 수 있는 XML 파일 양식으로 운송 자원을 작성할 수도 있습니다. 결과적으로 운송 자원 작성에 대해 다음과 같이 두 가지 옵션이 있습니다.

- WebSphere Commerce와 함께 제공된 견본 상점 중 하나 또는 기존 상점 아카이브에서 기존 운송 자원 편집
- XML 파일 양식으로 새 운송 자원 작성

기존 상점 아카이브의 운송 자원 편집에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오. XML 파일 양식으로 새 운송 자원 작성에 대한 자세한 정보는 『XML 파일을 사용하여 운송 자원 작성』을 참조하십시오.






XML 파일을 사용하여 운송 자원 작성

로더 패키지를 사용하여 데이터베이스에 로드할 수 있는 XML 파일 양식으로 운송 자원을 작성합니다. 로더 패키지에 대한 자세한 정보는 247 페이지의 제 7 부 『상점 공개』를 참조하십시오. 다국어 지원 상점을 작성하는 경우에는 상점이 지원하는 로케일 각각에 대한 별도 XML 파일을 작성하십시오. 로케일 고유의 파일은 쉽게 번역할 수 있도록 모든 설명 정보를 지정해야 합니다.

이들 태스크에 사용되는 견본 상점은 번역할 필요가 없는 모든 정보에 대해서는 하나의 shipping.xml 파일을 사용하고, 번역할 필요가 있는 정보에 대해서는 상점이 지원하는 로케일 각각에 대한 다른 shipping.xml 파일을 사용합니다. 로케일 고유의 파일은 쉽게 번역할 수 있도록 모든 설명 정보를 포함합니다.

XML 파일을 사용하여 상점의 운송 자원을 작성하려면, 다음과 같이 하십시오.

1. IBM WebSphere Commerce 계산 프레임워크 안내서를 검토하십시오. WebSphere Commerce 계산 프레임워크는 고객이 구매하려고 선택한 제품이나 서비스와 연관된 재정적 양(예: 운송 비용)을 계산합니다.
2. 견본 상점의 운송 자원을 작성하는 데 사용된 XML 파일을 검토하십시오. 견본 상점의 모든 파일은 해당 상점 아카이브 파일에 있습니다. 개별 견본 상점에는 운송 정보를 포함하는 두 개 이상의 shipping.xml 파일이 들어 있습니다. 상점 아카이브 파일은 다음 디렉토리에 있습니다.

-  NT drive:\WebSphere\CommerceServer\samplstores
-  2000 drive:\Program Files\WebSphere\CommerceServer\samplstores
-  AIX /usr/WebSphere/CommerceServer/samplstores
-  Solaris /opt/WebSphere/CommerceServer/samplstores
-  Linux /opt/WebSphere/CommerceServer/samplstores

- ▶ 400 /qibm/proddata/WebCommerce/samplestores

주: WebSphere Commerce 온라인 도움말에는 견본 상점에 포함된 데이터 자원 각각에 대한 정보가 포함됩니다.

상점 아카이브의 shipping.xml 파일을 보려면 ZIP 프로그램을 사용하여 이 파일의 압축을 푸십시오. shipping.xml 파일은 데이터 디렉토리에 있습니다. 언어별 shipping.xml은 데이터 디렉토리의 로케일 고유의 서브디렉토리에 있습니다.

- 361 페이지의 부록 B 『데이터 작성』 정보를 검토하십시오.
- 견본 상점 아카이브의 shipping.xml 파일 중 하나를 복사하거나 새 파일을 작성해서 shipping.xml 파일을 작성하십시오. 자세한 정보는 shipping.xml에 대한 해당 DTD 파일을 참조하십시오. DTD 파일은 다음 디렉토리에 있습니다.

- ▶ NT drive:\WebSphere\CommerceServer\xml\sar
- ▶ 2000 drive:\Program Files\WebSphere\CommerceServer\xml\sar
- ▶ AIX /usr/WebSphere/CommerceServer/xml/sar
- ▶ Solaris /opt/WebSphere/CommercServer/xml/sar
- ▶ Linux /opt/WebSphere/CommercServer/xml/sar
- ▶ 400 /qibm/proddata/WebCommerce/xml/sar

5. 상품 및 서비스를 운송할 관할 구역 및 관할 구역 그룹을 정의하십시오. 모든 관할 구역은 관할 구역 그룹에 속해야 합니다.

- a. 다음 예를 참조하여 XML 파일의 JURSTGROUP 테이블에 관할 구역 그룹을 정의하십시오.

```
< jurstgroup
  jurstgroup_id="@jurstgroup_id_1"
  description="Jurisdiction Group1 for Shipping"
  subclass="1"
  storeent_id="@storeent_id_1"
  code="World"/>
```

여기서,

- jurstgroup_id는 생성된 고유 키입니다.
- description은 관할 구역 그룹을 관리하는 사용자 인터페이스에 적절히 표시할 관할 구역 그룹에 대한 간략한 설명입니다.
- subclass는 다음과 같은 관할 구역 그룹 서브클래스입니다.
 - 1 = ShippingJurisdictionGroup
 - 2 = TaxJurisdictionGroup

- storeent_id는 이 관할 구역 그룹과 연관된 상점 엔티티입니다.
 - code는 상점 엔티티 및 서브클래스와 함께 이 관할 구역 그룹을 고유하게 식별합니다.
- b. 다음 예를 참조하여 XML 파일의 JURST 테이블에 관할 구역을 정의하십시오.

```
< jurst
jurst_id="@jurst_id_1"
storeent_id="@storeent_id_1"
code="World"
subclass="1"/>
```

여기서,

- jurst_id는 생성된 고유 키입니다.
 - storeent_id는 이 관할 구역 그룹과 연관된 상점 엔티티입니다.
 - code는 상점 엔티티 및 서브클래스와 함께 이 관할 구역 그룹을 고유하게 식별합니다.
 - subclass는 다음과 같은 관할 구역 서브클래스입니다.
 - 1 = ShippingJurisdiction
 - 2 = TaxJurisdiction
- c. 다음 예를 참조하여 JURSTGRPREL 테이블에 정보를 추가해서, b단계에서 작성한 관할 구역을 a단계에서 정의한 관할 구역 그룹에 연관시키십시오.

```
< jurstgprel
jurst_id="@jurst_id_1"
jurstgroup_id="@jurstgroup_id_1"
subclass="1"/>
```

여기서,

- jurst_id는 생성된 고유 키입니다.
 - jurstgroup_id는 관할 구역 그룹입니다.
 - subclass는 관할 구역 및 관할 구역 그룹의 서브클래스로 이 둘은 서로 일치해야 합니다.
 - 1 = ShippingJurisdiction[Group]
 - 2 = TaxJurisdiction[Group]
- d. 상점이 지원하는 모든 관할 구역 및 관할 구역 그룹에 대해 a부터 c단계까지를 반복하십시오.
6. 상점이 사용할 운송 방법을 정의하십시오.
- a. 다음 예를 참조하여 XML 파일의 SHIPMODE 테이블에 운송 방법을 정의하십시오.

```

<shipmode
shipmode_id="@shipmode_id_1"
field1
storeent_id="@storeent_id_1"
code="Ground 1 week"
carrier="XYZ Carrier" />

```

여기서,

- shipmode_id는 생성된 고유 키입니다.
 - field1은 사용자 정의에 사용 가능한 필드입니다.
 - storeent_id는 이 운송 방법과 연관된 상점 엔티티입니다.
 - code는 판매자 지정 코드로 상점 엔티티에 대해 고유합니다.
 - carrier는 운송 회사의 이름 또는 식별자입니다.
- b. 다음 예를 참조하여 SHPMODEDSC 테이블에 운송 방법에 대한 정보를 추가하십시오. 다국어 지원 상점을 작성하는 경우에는 로케일 고유의 XML 파일에 이 정보를 포함시켜야 합니다.

```

< shpmodedsc
description="International mail"
field1="USD$5.00 per order plus USD$1.00 for each item"
field2="5 business days"
shipmode_id="@shipmode_id_1"
language_id="&en_US;" />

```

여기서,

- description은 고객이 선택할 수 있도록 적절히 표시할 ShippingMode에 대한 간략한 설명입니다.
 - field1 및 field2는 사용자 정의에 사용 가능한 필드입니다.
 - shipmode_id는 생성된 고유 키입니다.
 - language_id는 사용된 언어입니다.
- c. 상점의 모든 운송 방법에 대해 a부터 b단계까지 반복하십시오.
7. 상점에서 사용할 계산 코드를 정의하십시오.
- a. 다음 예를 참조하여 XML 파일 CALCODE 테이블에 계산 코드를 정의하십시오.

```

< calcode
calcode_id="@calcode_id_1"
code="shipping Code 1- per/order"

```

```

calusage_id="-2"
storeent_id="@storeent_id_1"
groupby="0"
published="1"
sequence="+0.00E+000"
calmethod_id="-23"
calmethod_id_app="-24"
calmethod_id_qfy="-22"
flags="0" />

```

여기서,

- calcode_id는 생성된 고유 키입니다.
- code는 이 CalculationCode, 주어진 특정 CalculationUsage 및 StoreEntity를 고유하게 식별하는 문자열입니다.
- calusage_id는 이 CalculationCode를 사용할 계산 종류를 나타냅니다. 예를 들어, CalculationCode를 사용하여 다음과 같은 재정 총계 중 하나를 계산할 수 있습니다.
 - 할인(-1)
 - 운송 경비(-2)
 - 판매세(-3)
 - 운송세(-4)
 - 쿠폰(-5)
- storeent_id는 계산 코드와 연관된 상점 엔티티입니다.
- groupby는 계산을 수행할 때 OrderItems을 그룹으로 묶는 방법을 CalculationCodeCombineMethod에 나타내는 비트 플래그입니다. 0 = 그룹 없음. 적용 가능한 모든 OrderItems를 단일 그룹에 두십시오. 자세한 정보는 WebSphere Commerce 온라인 도움말의 *CALCODE* 테이블: 정보를 참조하십시오.
- published는 계산 코드가 공개되었는지 여부를 지정합니다.
 - 0 = 공개되지 않음(일시적으로 사용 안함)
 - 1 = 공개됨
 - 2 = 삭제 표시됨(및 공개되지 않음)
- sequenceCalculationCodes는 가장 낮은 것에서 가장 높은 것 순서로 계산되거나 적용됩니다. 두 개의 계산 코드가 동일한 순서 번호를 갖는 경우, 하위 calcode_id를 갖는 계산 코드가 우선 계산됩니다.

- calmethod_id는 이 CalculationCode의 재정 총계를 계산하는 방법을 정의하는 CalculationCodeCalculateMethod입니다. 운송에 대한 CalculationCodeCalculateMethod, calmethod_id="-23"은 WebSphere Commerce와 함께 제공되는 유일한 운송 계산 방법입니다.
- calmethod_id_app는 연관된 OrderItems의 계산 총계를 저장하는 CalculationCodeApplyMethod입니다. 운송에 대한 CalculationCodeApplyMethod, calmethod_id_app="-24"는 WebSphere Commerce와 함께 제공되는 유일한 운송 적용 방법입니다.
- calmethod_id_qfy는 이 CalculationCode와 연관되는 OrderItems를 정의하는 CalculationCodeQualifyMethod입니다. 운송에 대한 CalculationCodeQualifyMethod, calmethod_id_qfy="-22"는 WebSphere Commerce와 함께 제공되는 유일한 운송 조건 방법입니다.
- flags는 이 CalculationCode의 CalculationCodeQualifyMethod가 호출되었는지 여부를 지정합니다.
 - 0 = 제한되지 않음. 방법이 호출되지 않습니다.
 - 1 = 제한됨. 방법이 호출됩니다.

b. 다음 예를 참조하여, XML 파일의 CALCODEDSC 테이블에 계산 코드 설명 정보를 추가하십시오. 다국어 지원 상점을 작성하는 경우에는 이 정보를 로케일 고유의 XML 파일에 포함시켜야 합니다.

```
<calcodedsc
calcode_id="@calcode_id_3"
description="5.00USD per order"
language_id="@&en_US"
longdescription="This shipping calculation code charges 5.00USD
per order."
/>
```

여기서,

- calcode_id는 이 정보를 적용할 계산 코드입니다.
- description은 계산 코드에 대한 간단한 설명입니다.
- language_id는 이 정보를 적용할 언어입니다.
- longdescription는 계산 코드에 대한 자세한 설명입니다.

c. 상점에서 사용된 개별 계산 코드에 대해 a부터 b단계까지를 반복하십시오.

8. 상점의 계산 규칙을 정의하십시오.

a. 다음 예를 참조하여 XML 파일의 CALRULE 테이블에 계산 규칙을 설정하십시오.

```
<calrule
calrule_id="@calrule_id_1"
calcode_id="@calcode_id_1"
```

```

startdate="1900-01-01 00:00:00.000000"
enddate="2100-01-01 00:00:00.000000"
sequence="+1.0000000000000000E+000"
combination="2"
calmethod_id="-27"
calmethod_id_qfy="-26"
flags="1"
identifier="1" />

```

여기서,

- calrule_id는 생성된 고유 식별자입니다.
- calcode_id는 이 계산 규칙이 일부분이 되는 계산 코드입니다.
- startdate는 이 계산 규칙이 유효하게 되는 시점입니다.
- enddate는 이 계산 규칙이 유효하지 않게 되는 시점입니다.
- sequence는 이 계산 규칙이 처리되는 순서입니다. 같은 계산 코드에 대해 계산 규칙은 낮은 값에서 높은 값 순서로 처리됩니다.
- combination은 기본 CalculationRuleCombineMethod 구현이 수행할 특별 처리에 대한 비트 플래그를 지정합니다. 자세한 정보는 WebSphere Commerce 온라인 도움말의 CALRULE 테이블을 참조하십시오.
- calmethod_id는 OrderItems 세트의 재정 결과를 계산하는 CalculationRuleCalculateMethod입니다.
- calmethod_id_qfy는 CalculationRuleCalculateMethod에 보내야 하는 OrderItems 세트를 결정하는 CalculationRuleQualifyMethod입니다.
- flags는 CalculationRuleCombineMethod에 의해 사용되어 이 계산 규칙이 다른 계산 규칙과 결합되는 방법을 결정합니다. 자세한 정보는 CALRULE 테이블을 참조하십시오.
- identifier는 계산 코드와 조합하여 이 계산 규칙을 식별합니다.

자세한 정보는 WebSphere Commerce 온라인 도움말의 CALRULE 테이블을 참조하십시오.

- b. 상점에서 사용된 개별 계산 규칙에 대해 a단계를 반복하십시오. 개별 계산 코드는 여러 계산 규칙을 가질 수 있습니다. 예를 들어, calcode_id="@calcode_id_1"는 여러 calrule_ids와 연관됩니다.

9. 상점의 계산 스케일을 정의하십시오.

계산 스케일은 계산에 적용될 범위의 세트입니다. 예를 들어, 운송 경비의 경우 각각 특별 비용에 해당하는 중량 범위 세트가 있습니다. 중량이 0-5kg인 상품은 운송할 때 비용이 \$10.00입니다. 중량이 5-10kg인 상품은 운송할 때 비용이 \$15.00입니다. 이런 범위가 스케일을 작성합니다.

- a. 다음 예를 참조하여 XML 파일의 CALSCALE 테이블에 계산 스케일을 설정하십시오.

```
<calscale
calscale_id="@calscale_id_1"
code="Scale Code 1 per order USD"
storeent_id="@storeent_id_1"
calusage_id="-2"
setccurr="USD"
calmethod_id="-28"/>
```

여기서,

- calscale_id는 생성된 고유 식별자입니다.
- code는 이 계산 스케일, 주어진 특정 계산 사용법 및 상점 엔티티를 고유하게 식별하는 문자열입니다.
- storeent_id는 이 계산 스케일이 일부가 되는 상점 엔티티입니다.
- calusage_id는 이 CalculationScale이 사용되는 계산 종류를 나타냅니다. 예를 들어, CalculationScale을 사용하여 다음과 같은 재정 총계 중 하나를 계산할 수 있습니다.
 - 할인(-1)
 - 운송 경비(-2)
 - 판매세(-3)
 - 운송세(-4)
 - 쿠폰(-5)
- setccurr를 지정하면, 이 계산 스케일에 대해 계산 범위 오브젝트의 범위 시작 값에 대해 선택한 통화를 나타냅니다. CalculationScaleLookupMethod는 이 통화의 "찾아보기 번호"를 리턴해야 합니다.
- calmethod_id는 주어진 주문 항목 세트가 찾아보기 값, 기본 재정값, 결과 곱셈 및 재정 총계를 계산하는 데 계산 스케일이 사용할 수학적 중량 세트를 결정하는 CalculationScaleLookupMethod입니다. 사용할 CalculationScaleLookupMethod를 결정하려면 다음을 수행하십시오.
 - WebSphere Commerce 온라인 도움말의 CALMETHOD 테이블을 참조하십시오. SUBCLASS 열의 설명을 참조하십시오. CALMETHOD 테이블

블: 정보에 대한 링크를 누르십시오. 이 테이블은 사용 가능한 계산 방법의 유형을 나열합니다. MonetaryCalculationScaleLookupMethod 방법은 9번입니다.

- wcs.bootstrap_xx_XX.xml 부트스트랩 파일을 여십시오. 여기서, xx_XX는 로케일 코드입니다. 부트스트랩 파일은 다음 디렉토리에 있습니다.
 -  NT drive:\WebSphere\CommerceServer\schema\xml
 -  2000 drive:\Program Files\WebSphere\CommerceServer\schema\xml
 -  AIX /usr/WebSphere/CommerceServer/schema/xml
 -  Solaris /opt/WebSphere/CommerceServer/schema/xml
 -  Linux /opt/WebSphere/CommerceServer/schema/xml
 -  400 /qibm/proddata/WebCommerce/schema/xml
- 사용 가능한 계산 방법(CALMETHOD)을 나열하는 절을 위치 지정하십시오.
- 세금(판매세는 -3이고 운송세는 -4)의 calusage_ID 값과 함께 계산 방법을 위치 지정하십시오.
- 서브클래스가 7인 계산 방법을 찾으십시오. 몇 가지가 있습니다. 사용자 필요에 맞게 하나를 오더피킹하십시오.

자세한 정보는 WebSphere Commerce 온라인 도움말의 CALSCALE 테이블을 참조하십시오.

- b. 상점에서 사용된 개별 계산 스케일에 대해 a단계를 반복하십시오. 예를 들어, 운송의 경우 InFashion이 주문별 비용 스케일 및 항목별 비용 스케일을 작성합니다.







10. 계산 스케일의 계산 범위를 정의하십시오.

- a. 다음 예를 참조하여 XML 파일의 CALRANGE 테이블에 계산 범위를 설정하십시오.

```
<calrange  
calrange_id="@calrange_id_1"  
calscale_id="@calscale_id_1"  
calmethod_id="-33"  
rangestart="0.00000"  
cumulative="0"/>
```

여기서

- calrange_id는 생성된 고유 식별자입니다.

- calscale_id는 이 계산 범위가 일부만이 되는 계산 스케일입니다.
- calmethod_id는 CalculationRangeLookupResult로부터 재정 총계를 결정하는 CalculationRangeMethod입니다. 예를 들어, FixedAmountCalculationRangeCmd, PerUnitAmountCalculationRangeCmd 또는 PercentageCalculationRangeCmd입니다. CalculationRangeMethod를 결정하려면, 다음을 수행하십시오.
 - WebSphere Commerce 온라인 도움말의 CALMETHOD 테이블을 참조하십시오. SUBCLASS 열의 설명을 참조하십시오. CALMETHOD 테이블: 정보에 대한 링크를 누르십시오. 이 테이블은 사용 가능한 계산 방법의 유형을 나열합니다. CalculationRangeMethod는 10번입니다.
 - wcs.bootstrap_xx_XX.xml 부트스트랩 파일을 여십시오. 여기서, xx_XX는 로케일 코드입니다. 부트스트랩 파일은 다음 디렉토리에 있습니다.
 -  NT drive:\WebSphere\CommerceServer\schema\xml
 -  2000 drive:\Program Files\WebSphere\CommerceServer\schema\xml
 -  AIX /usr/WebSphere/CommerceServer/schema/xml
 -  Solaris /opt/WebSphere/CommerceServer/schema/xml
 -  Linux /opt/WebSphere/CommerceServer/schema/xml
 -  400 /qibm/proddata/WebCommerce/schema/xml
 - 사용 가능한 계산 방법(CALMETHOD)을 나열하는 절을 위치 지정하십시오.
 - 세금(판매세는 -3이고 운송세는 -4)의 calusage_ID 값과 함께 계산 방법을 위치 지정하십시오.
 - 서브클래스가 9인 계산 방법을 찾으십시오. 몇 가지가 있습니다. 사용자 필요에 맞게 하나를 오더피킹하십시오.
- cumulative은 올바른 값입니다.
 - 0 = 가장 높은 RANGESTART 값과 일치하는 CalculationRange만 사용됩니다.
 - 1 = 일치하는 모든 CalculationRanges가 사용됩니다. 계산된 재정 총계는 최종 결과에 합계됩니다.

자세한 정보는 WebSphere Commerce 온라인 도움말의 CALRANGE 테이블을 참조하십시오.

- b. 상점에서 사용된 계산 스케일과 연관된 개별 계산 범위에 대해 a단계를 반복하십시오.

11. 계산 스케일에 대해 계산 찾아보기 값을 정의하십시오. 계산 찾아보기 값은 계산 스케일과 연관된 값입니다. 예를 들어, 계산 스케일에는 다음 중량 범위 및 운송에 대해 연관된 가격이 포함됩니다.

- 0 - 5kg까지 비용: \$10.00
- 5 -10kg까지 비용: \$15.00

찾아보기 값은 \$10.00 및 \$15.00입니다.

a. 다음 예를 참조하여, XML 파일의 CALRLOOKUP 테이블에 계산 찾아보기 값을 설정하십시오. 다국어 지원 상점을 작성하는 경우에는 상점이 지원하는 로케일별로 파일이 하나인 로케일 고유의 XML 파일에 이 정보를 포함시켜야 합니다. 예를 들어, 상점이 미국 및 일본에 있는 고객에게 운송하는 경우에는 하나의 XML 파일에 미국 달러 찾아보기 값을 추가하고, 다른 XML 파일에 엔 찾아보기 값을 추가시켜야 합니다.

```
<calrlookup  
calrlookup_id="@calrlookup_id_1"  
setccurr="USD"  
calrange_id="@calrange_id_1"  
value="5.00"/>
```

여기서,

- calrlookup_id는 생성된 고유 식별자입니다.
- calrange_id는 이 계산 범위 찾아보기 결과가 일부분이 되는 계산 범위입니다.
- value은 재정 결과를 결정하는 데 계산 범위의 계산 범위 방법이 사용한 계산 범위 찾아보기 결과의 값입니다.

자세한 정보는 WebSphere Commerce 온라인 도움말의 CALRLOOKUP 테이블을 참조하십시오.

b. 상점에서 사용된 계산 스케일과 연관된 개별 찾아보기 값에 대해 a 단계를 반복하십시오.

12. 계산 규칙과 계산 스케일을 연관시키십시오.

a. 다음 예를 참조하여 XML 파일의 CRULESCALE 테이블에 있는 계산 규칙을 계산 스케일과 연관시키십시오.

```
< crulescale  
calrule_id="@calrule_id_1"  
calscale_id="@calscale_id_1" />
```

여기서

- calrule_id는 계산 규칙입니다.

- calscale_id는 계산 스케일입니다.
- b. 개별 계산 스케일 및 규칙 연관에 대해 a단계를 반복하십시오.



@ 및 & 사용에 대한 자세한 정보는 361 페이지의 부록 B 『데이터 작성』을 참조하십시오.

운송 이행 자원 작성




운송 자원이 상점에서 올바르게 작업되도록 하려면 운송 관할 구역 그룹을 계산 규칙에, 서비스 센터를 상점에서 사용하는 운송 방법에 연관시켜야 합니다.

운송 자원을 서비스 센터에 연관시키기 전에 이행 자원을 작성해야 합니다. 이행 자원 작성에 대한 자세한 정보는 124 페이지의 『WebSphere Commerce에서 이행 자원 작성』을 참조하십시오.

이행 자원을 작성한 후, SHPJCRULE 및 SHPARRANGE 테이블에 정보를 추가해서 운송 자원을 이행 자원과 연관시키십시오. 다음을 수행하십시오.

1. *IBM WebSphere Commerce 계산 프레임워크 안내서*를 검토하십시오. WebSphere Commerce 계산 프레임워크는 고객이 구매하려고 선택한 제품이나 서비스와 연관된 재정적 양(예: 운송 비용)을 계산합니다.
2. 견본 상점의 운송 이행 자원을 작성하는 데 사용된 XML 파일을 검토하십시오. 견본 상점의 모든 파일은 해당 상점 아카이브 파일에 있습니다. 개별 견본 상점에는 운송 이행 정보를 포함하는 shipfulfill.xml 파일이 들어 있습니다. 상점 아카이브의 shipfulfill.xml 파일을 보려면, ZIP 프로그램을 사용하여 이 파일의 압축을 푸십시오. shipfulfill.xml 파일은 데이터 디렉토리에 있습니다.







상점 아카이브 파일은 다음 디렉토리에 있습니다.

-  NT drive:\WebSphere\CommerceServer\samplstores
-  2000 drive:\Program Files\WebSphere\CommerceServer\samplstores
-  AIX /usr/WebSphere/CommerceServer/samplstores
-  Solaris /opt/WebSphere/CommerceServer/samplstores
-  Linux /opt/WebSphere/CommerceServer/samplstores
-  400 /qibm/proddata/WebCommerce/samplstores

주: WebSphere Commerce 온라인 도움말에는 견본 상점에 포함된 데이터 자원 각각에 대한 정보가 포함됩니다.

3. 361 페이지의 부록 B 『데이터 작성』 정보를 검토하십시오.

4. 견본 상점 아카이브의 shipfulfill.xml 파일 중 하나를 복사하거나 새 파일을 작성해서 shipfulfill.xml 파일을 작성하십시오. 자세한 정보는 shipfulfill.xml에 대한 해당 DTD 파일을 참조하십시오. DTD 파일은 다음 디렉토리에 있습니다.

-  NT drive:\WebSphere\CommerceServer\xml\sar
-  2000 drive:\Program Files\WebSphere\CommerceServer\xml\sar
-  AIX /usr/WebSphere/CommerceServer/xml/sar
-  Solaris /opt/WebSphere/CommerceServer/xml/sar
-  Linux /opt/WebSphere/CommerceServer/xml/sar
-  400 /qibm/proddata/WebCommerce/xml/sar

5. SHPJCRULE 테이블에 정보를 추가해서 운송 관할 구역 그룹에 계산 규칙을 연관시키십시오. 다음 예를 참조하십시오. 다국어 지원 상점을 작성하는 경우에는 상점이 지원하는 개별 로케일에 대해서도 XML 파일을 작성하십시오.

```
<shpjcrule
  calrule_id="@calrule_id_1"
  ffmcenter_id="@ffmcenter_id_1"
  jurstgroup_id="@jurstgroup_id_1"
  precedence="0"
  shipmode_id="@shipmode_id_1"
  shpjcrule_id="@shpjcrule_id_1"
```

여기서,

- calrule_id는 사용된 계산 규칙입니다.
- ffmcenter_id는 서비스 센터입니다. 이것이 널(NULL)값이면 이 연관은 모든 서비스 센터에 적용됩니다.
- jurstgroup_id는 운송 관할 구역 그룹입니다. 이것이 널(NULL)값이면 이 연관은 모든 운송 관할 구역 그룹에 적용됩니다.
- precedence는 같은 서비스 센터 및 운송 방법에 대해 하나 이상의 특정 운송 관할 구역 그룹에 해당됨을 의미합니다. 가장 높은 SHPJCRULE.PRECEDENCE 값을 갖는 계산 규칙만 사용됩니다.
- shipmode_id는 운송 방법입니다.
- shpjcrule_id는 생성된 고유 식별자입니다.

6. 상점의 개별 관할 구역 그룹, 서비스 센터 및 규칙 연관에 대해 3단계를 반복하십시오.

7. SHPARRANGE 테이블에 정보를 추가해서 운송 방법 및 서비스 센터를 상점과 연관시키십시오. 다음 예를 안내서로 사용하십시오.

```
<shparrange
  shparrange_id="@shparrange_id_2"
  store_id="@storeent_id_1"
  ffmcenter_id="@ffmcenter_id_1"
```



```

shipmode_id= "@shipmode_id_2"
startdate="1970-06-22 23:00:00.000000"
enddate= "2008-06-22 23:00:00.000000"
precedence= "0"
flags="0"
/>

```

여기서,

- shparrange_id는 생성된 고유 식별자입니다.
- store_id는 상점입니다.
- ffmcenter_id는 서비스 센터입니다.
- shipmode_id는 운송 방법입니다. 널(NULL)값은 운송 방법에 관계없이 이 운송 협정을 사용할 수 있음을 나타냅니다.
- startdate는 이 운송 협정이 유효하게 되는 시점입니다.
- enddate는 이 운송 협정이 유효하지 않게 되는 시점입니다.
- precedence는 특정 시간에 둘 이상의 운송 협정(같은 상점 및 운송 방법에 대해)이 유효할 때, 가장 높은 PRECEDENCE 값을 갖는 것만 사용됩니다.
- flags는 비트 플래그를 포함합니다.
 - 1 = 제한 - 이 운송 협정은 운송 주소가 이 운송 협정과 연관된 (SHPARJURGP 테이블을 통해) 운송 관할 구역 그룹 중 하나와 일치하는 주문 항목에 대해서만 적용됩니다.

8. 상점에서 사용된 모든 운송 방법에 대해 5단계를 반복하십시오.



@ 및 & 사용에 대한 자세한 정보는 361 페이지의 부록 B 『데이터 작성』을 참조하십시오.

상점-카탈로그-운송 자원 작성

운송 방법을 상점에 연관시키려면, 상점이 포함하는 개별 장기 구매 계약에 대해 계산 코드를 상점의 카탈로그 항목과 연관시켜야 합니다.







상점-카탈로그-운송 자원을 작성하기 전에 상점 및 카탈로그 자원을 작성해야 합니다. 상점 자원 작성에 대한 자세한 정보는 47 페이지의 『XML 파일에 상점 데이터 자원 작성』을 참조하십시오. 카탈로그 자원 작성에 대한 자세한 정보는 84 페이지의 『상점 카탈로그 자원 표시』를 참조하십시오.

상점-카탈로그-운송 자원을 작성하려면, 다음을 수행하십시오.

1. *IBM WebSphere Commerce 계산 프레임워크 안내서*를 검토하십시오. WebSphere Commerce 계산 프레임워크는 고객이 구매하려고 선택한 제품이나 서비스와 연관된 재정적 양(예: 운송 비용)을 계산합니다.

2. 견본 상점의 운송 이행 자원을 작성하는 데 사용된 XML 파일을 검토하십시오. 견본 상점의 모든 파일은 해당 상점 아카이브 파일에 있습니다.






상점 아카이브 파일은 다음 디렉토리에 있습니다.

-  NT drive:\WebSphere\CommerceServer\samplstores
-  2000 drive:\Program Files\WebSphere\CommerceServer\samplstores
-  AIX /usr/WebSphere/CommerceServer/samplstores
-  Solaris /opt/WebSphere/CommerceServer/samplstores
-  Linux /opt/WebSphere/CommerceServer/samplstores
-  400 /qibm/proddata/WebCommerce/samplstores

주: WebSphere Commerce 온라인 도움말에는 견본 상점에 포함된 데이터 자원 각각에 대한 정보가 포함됩니다.

개별 견본 상점에는 운송 이행 정보를 포함하는 store-catalog-shipping.xml 파일이 들어 있습니다. 상점 아카이브의 store-catalog-shipping.xml 파일을 보려면, ZIP 프로그램을 사용하여 이 파일의 압축을 푸십시오. store-catalog-shipping.xml 파일은 데이터 디렉토리에 있습니다.

3. 361 페이지의 부록 B 『데이터 작성』 정보를 검토하십시오.
4. 견본 상점 아카이브의 store-catalog-shipping.xml 파일 중 하나를 복사하거나 새 파일을 작성해서 store-catalog-shipping.xml 파일을 작성하십시오. 자세한 정보는 store-catalog-shipping.xml에 대한 해당 DTD 파일을 참조하십시오. DTD 파일은 다음 디렉토리에 있습니다.

-  NT drive:\WebSphere\CommerceServer\xml\sar
-  2000 drive:\Program Files\WebSphere\CommerceServer\xml\sar
-  AIX /usr/WebSphere/CommerceServer/xml/sar
-  Solaris /opt/WebSphere/CommerceServer/xml/sar
-  Linux /opt/WebSphere/CommerceServer/xml/sar
-  400 /qibm/proddata/WebCommerce/xml/sar

5. CATENCALCD 테이블에 정보를 추가해서 상점-카탈로그-운송 관계를 작성하십시오. 다음 예를 안내서로 사용하십시오.

```
<catencalcd
  calcode_id="@calcode_id_1"
  catencalcd_id="@catencalcd_id_1"
  store_id="@storeent_id_1"
/>
```

여기서,

- calcode_id는 계산 코드입니다.
- catencalcd_id는 생성된 고유 식별자입니다.
- store_id는 상점입니다.



@ 및 & 사용에 대한 자세한 정보는 361 페이지의 부록 B 『데이터 작성』을 참조하십시오.

기본 운송 방법 작성

상점의 기본 운송 방법을 작성하려면, STOREDEF 테이블에 정보를 추가해야 합니다. STOREDEF 테이블에 정보를 추가하려면, 다음을 수행하십시오.

1. 견본 상점의 상점 기본 자원을 작성하는 데 사용된 XML 파일을 검토하십시오. 견본 상점의 모든 파일은 해당 상점 아카이브 파일에 있습니다.

상점 아카이브 파일은 다음 디렉토리에 있습니다.




- **NT** drive:\WebSphere\CommerceServer\samplstores
- **2000** drive:\Program Files\WebSphere\CommerceServer\samplstores
- **AIX** /usr/WebSphere/CommerceServer/samplstores
- **Solaris** /opt/WebSphere/CommerceServer/samplstores
- **Linux** /opt/WebSphere/CommerceServer/samplstores
- **400** /qibm/proddata/WebCommerce/samplstores

주: WebSphere Commerce 온라인 도움말에는 견본 상점에 포함된 데이터 자원 각각에 대한 정보가 포함됩니다.

개별 견본 상점에는 기본 운송 정보를 포함하는 store-defaults.xml 파일이 들어 있습니다. 상점 아카이브의 store-defaults.xml 파일을 보려면, ZIP 프로그램을 사용하여 이 파일의 압축을 푸십시오. store-defaults.xml 파일은 데이터 디렉토리에 있습니다.

2. 361 페이지의 부록 B 『데이터 작성』 정보를 검토하십시오.
3. 견본 상점 아카이브의 store-defaults.xml 파일 중 하나를 복사하거나 새 파일을 작성해서 store-defaults.xml 파일을 작성하십시오. 자세한 정보는 store-defaults.xml에 대한 해당 DTD 파일을 참조하십시오. DTD 파일은 다음 디렉토리에 있습니다.

- **NT** drive:\WebSphere\CommerceServer\xml\sar
- **2000** drive:\Program Files\WebSphere\CommerceServer\xml\sar
- **AIX** /usr/WebSphere/CommerceServer/xml/sar

-  /opt/WebSphere/CommerceServer/xml/sar
-  /opt/WebSphere/CommerceServer/xml/sar
-  /qibm/proddata/WebCommerce/xml/sar

4. 다음 예를 참조하여 STOREDEF 테이블에 정보를 추가해서 XML 파일에 상점의 기본 운송 방법을 지정하십시오.

```
<storedef
  store_id="@storeent_id_1"
  shipmode_id="@shipmode_id_1"
/>
```

여기서,

- store_id는 상점입니다.
- shipmode_id는 상점의 기본 운송 방법입니다.



@ 및 & 사용에 대한 자세한 정보는 361 페이지의 부록 B 『데이터 작성』을 참조하십시오.

제 19 장 세금 지원

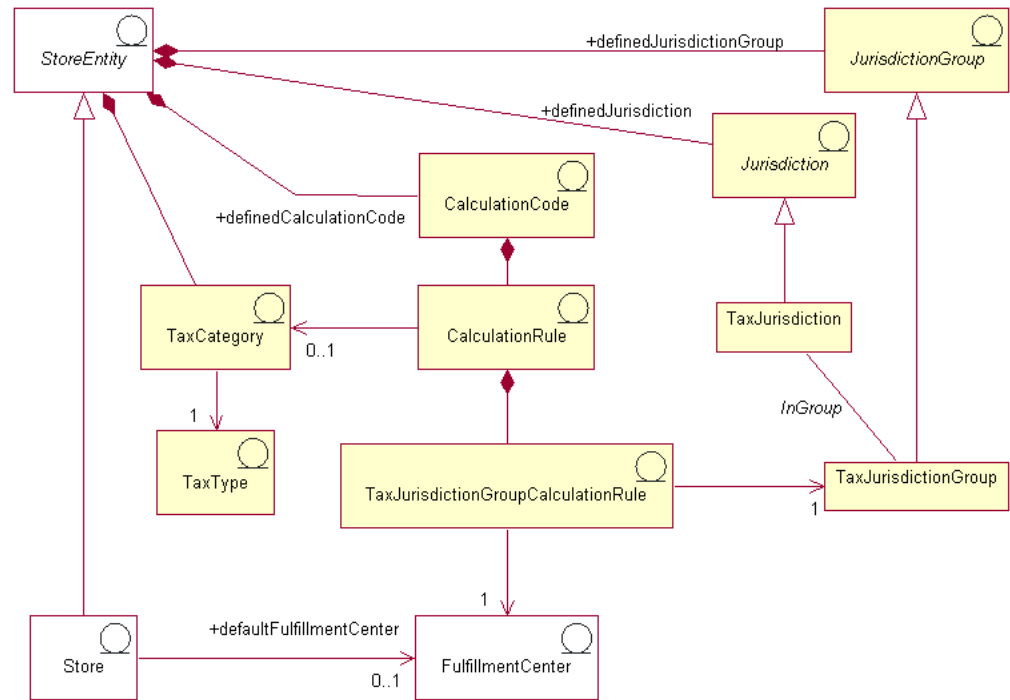
상점에서 제공하는 상품 및 서비스에 대한 세금을 부과하고 징수하려면, WebSphere Commerce로 작성된 상점이 다음을 포함해야 합니다.

- 세금 카테고리
- 계산 코드
- 관할 구역 및 관할 구역 그룹

세금 카테고리, 계산 코드, 관할 구역 및 관할 구역 그룹을 조합하여 상점의 세금 비용을 작성합니다.

WebSphere Commerce에서 세금 지원 이해

다음 도표는 WebSphere Commerce 서버의 과세 구조를 나타냅니다.



이 도표 및 상점 데이터 절의 기타 모두는 WebSphere Commerce 서버 정보 모델의 일부입니다. 정보 모델에 대한 자세한 정보는 29 페이지의 『상점 데이터 정보 모델』을 참조하십시오. 이 도표에 사용된 규칙에 대한 자세한 정보는 359 페이지의 부록 A 『UML 범례』를 참조하십시오.

세금 카테고리

세금 카테고리는 연방, 주 또는 지방, 시와 같이 상점이 징수할 필요가 있는 세금의 여러 종류에 해당합니다.

상점 엔티티가 여러 세금 카테고리를 가진다 해도 세금 카테고리는 한 상점 엔티티의 일부분입니다. 상점 엔티티를 삭제하면, 해당 상점과 연관된 세금 카테고리도 삭제됩니다.

세금 유형

상점은 보통 두 가지 유형의 세금 즉, 판매세나 사용세 그리고 운송세를 징수합니다. 개별 세금 카테고리는 하나의 세금 유형을 가집니다. 개별 세금 카테고리가 오직 하나의 세금 유형(예: 연방 세금 카테고리는 판매세 유형임)이라 해도, 다른 여러 세금 카테고리가 같은 세금 유형(예: 판매세 유형의 세금은 연방, 지방 및 시 카테고리에 적용 가능)에 속할 수 있습니다.

계산 코드

계산 코드는 세금 비용을 계산하는 데 사용됩니다. 즉, 세금 계산 코드는 주문 항목의 세금을 계산하는 방법을 나타냅니다. 주문 항목에 대한 세금을 계산하려면, 카탈로그 항목이나 카탈로그 항목의 그룹 중 하나에 판매 세금 및 운송 세금 계산 코드를 지정해야 합니다. 개별 세금 유형의 오직 하나의 계산 코드만 특정 주문 항목이나 카탈로그 항목 그룹에 적용할 수 있습니다. 판매세나 사용세는 보통 정가에 대해 징수하고, 운송세는 운송 경비에 대해 징수합니다.

계산 코드는 상점 엔티티의 일부분입니다. 계산 코드는 한 상점 엔티티와만 연관될 수 있지만, 상점 엔티티에는 여러 계산 코드가 있을 수 있습니다. 상점 엔티티가 삭제되면, 해당 상점 엔티티와 연관된 계산 코드도 삭제됩니다.



계산 코드에 관한 자세한 정보는 *IBM WebSphere Commerce* 계산 프레임워크 안내서를 참조하십시오.

계산 규칙

각 계산 코드는 최소한 하나의 계산 규칙을 가지며, 여기에서 각 세금 카테고리에 대한 계산을 정의하며, 계산이 수행될 조건을 지정합니다. 각 세금 계산 규칙은 세금 카테고리, 관할 구역 그룹 및 서비스 센터와 연관되며, 계산 규칙이 사용되는 조건을 함께 정의합니다. 예를 들어, 주문에서 지정된 운송 주소와 서비스 센터에 따라 다른 규칙이 선택되어 특정 세금 카테고리에 대한 총계를 계산할 수 있습니다.

각 계산 규칙은 정확히 하나의 계산 코드에 속합니다.

하나의 특정 세금 계산 코드에는 여러 계산 규칙, 세금 카테고리의 각 조합 중 하나, 세금 관할 구역 그룹 및 상점과 연관된 서비스 센터가 있습니다. 각 판매 세금과 운송

세금 계산 규칙은 복수 TaxJurisdictionGroupCalculationRules(TaxRules)와 연관될 수 있습니다. 예를 들어, 아래 표에서 계산 규칙 10001은 관할 구역 그룹 1234 및 1235 둘 다에 적용될 수 있습니다.

TAXJCRULE_ID	CALRULE_ID	FFMCENTER_ID	JURSTGROUP_ID	PRECEDENCE
10001	10001	NULL	1234	0
10002	10001	NULL	1235	0

각 TaxRule은 계산 규칙이 적용되어야 하는 조건을 정의합니다. 예를 들어, 상점이 운영하는 각 관할 구역 그룹에 대한 계산 규칙을 정의할 수 있습니다. 아래 예에서 계산 규칙 10001은 관할 구역 그룹 1234 및 1235 둘 다에 적용될 수 있습니다.

다음 예에서 세금 계산 코드는 지방 판매세 카테고리에 대해서는 계산 규칙 A를, 세금 관할 구역이 Alberta일 때는 B를, 세금 관할 구역이 British Columbia일 때는 규칙 C를 사용합니다.

세금 관할 구역	연방 판매세	지방 판매세
Alberta, Canada	Y%를 부여하는 계산 규칙 B	X%를 부여하는 계산 규칙 A
British Columbia, Canada	Y%를 부여하는 계산 규칙 B	Z%를 부여하는 계산 규칙 C

운송 주소가 둘 이상의 세금 관할 구역 그룹과 일치할 때는 가장 높이 연관된 TAXJCRULE.PRECEDENCE 열값이 있는 계산 규칙을 사용합니다.

TaxJurisdictionGroupCalculationRules(TaxRule)를 계산 규칙과 연관시키면 계산 규칙이 적용 가능한 시기를 결정합니다. 판매세나 운송세 계산 규칙은 TaxRules에서 제공하는 조건 중 하나가 충족되면 적용할 수 있습니다. 아래 예에서 계산 규칙 10001은 관할 구역 그룹 1001로 운송 중일 때나, 서비스 센터 1001에서 운송 중일 때, 또는 관할 구역 그룹 1001로 운송 중일 때 적용 가능합니다.

CALRULE_ID	FFMCENTER_ID	JURSTGROUP_ID
10001	NULL	1001
10001	1001	1001

각 TaxJurisdictionGroupCalculationRule은 기껏해야 한 개의 관할 구역 그룹과 연관됩니다. 계산 규칙 자체는 관할 구역 그룹과 직접 연관되지 않습니다.



계산 규칙에 관한 자세한 정보는 *IBM WebSphere Commerce 계산 프레임워크* 안내서를 참조하십시오.

관할 구역 및 관할 구역 그룹

관할 구역은 상품을 판매하는 지리적인 지역 또는 국가나 지역, 지방이나 영토를 나타내는 지역 또는 우편번호 범위입니다. 관할 구역들은 함께 그룹으로 묶여 관할 구역 그룹을 형성합니다.

WebSphere Commerce는 두 가지 유형의 관할 구역 즉, 운송 관할 구역 및 세금 관할 구역을 지원합니다. 이런 개별 관할 구역은 해당 그룹의 일부분입니다. 예를 들어, 운송 관할 구역은 운송 관할 구역 그룹에 속하고 세금 관할 구역은 세금 관할 구역 그룹에 속합니다.

관할 구역 및 관할 구역 그룹이 세금 비용을 계산하는 데 사용되는 계산 규칙을 결정합니다.

관할 구역 및 관할 구역 그룹은 상점 엔티티의 일부분입니다. 개별 관할 구역 및 관할 구역 그룹이 한 상점 엔티티의 일부분이지만, 상점 엔티티는 여러 관할 구역이나 관할 구역 그룹을 가질 수 있습니다. 상점 엔티티를 삭제하면, 해당 상점과 연관된 관할 구역 및 관할 구역 그룹도 삭제됩니다.



WebSphere Commerce 서버의 세금 자원 구조에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말의 세금 오브젝트 및 데이터 모델을 참조하십시오.

WebSphere Commerce에서 세금 자원 작성

WebSphere Commerce의 상점 서비스 도구를 사용하면, 상점 아카이브의 특정 세금 자원(예: 세금 카테고리 및 관할 구역)을 작성하고 편집할 수 있지만, 모든 세금 자원을 작성하고 편집할 수는 없습니다. 상점 서비스 도구를 사용하여 편집할 수 있는 자원에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말 주제 "상점 데이터베이스 자원 변경"을 참조하십시오.

주: 상점 서비스 도구는 상점 아카이브 양식의 사전 대량 자료 반입된 XML 파일을 사용하여 작업합니다.

로더 패키지를 사용하여 데이터베이스에 로드할 수 있는 XML 파일 양식으로 세금 자원을 작성할 수도 있습니다. 결과적으로 운송 자원 작성에 대해 다음과 같이 두 가지 옵션이 있습니다.

- WebSphere Commerce와 함께 제공된 견본 상점 중 하나 또는 기존 상점 아카이브에서 기존 세금 자원 편집
- 상점 아카이브의 일부분으로서 공개되거나, 로더 패키지를 사용하여 로드되는 XML 파일 양식으로 새 세금 자원 작성

기존 상점 아카이브의 세금 자원 편집에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오. XML 파일 양식으로 새 세금 자원 작성에 대한 자세한 정보는 『XML 파일을 사용하여 세금 자원 작성』을 참조하십시오.

XML 파일을 사용하여 세금 자원 작성

로더 패키지를 사용하여 데이터베이스에 로드할 수 있는 XML 파일 양식으로 세금 자원을 작성하십시오. 로더 패키지에 대한 자세한 정보는 247 페이지의 제 7 부 『상점



공개』를 참조하십시오. 다국어 지원 상점을 작성하는 경우에는 상점이 지원하는 로케일 각각에 대한 별도 XML 파일을 작성하십시오. 로케일 고유의 파일은 쉽게 번역할 수 있도록 모든 설명 정보를 지정해야 합니다.

이들 태스크에 사용되는 견본 상점은 번역할 필요가 없는 모든 정보에 대해서는 하나의 tax.xml 파일을 사용하고 번역할 필요가 있는 정보에 대해서는 상점이 지원하는 로케일 각각에 대해 다른 tax.xml 파일을 사용합니다. 로케일 고유의 파일은 모든 설명 정보를 포함합니다.

XML 파일을 사용하여 상점의 세금 자원을 작성하려면, 다음을 수행하십시오.

1. IBM WebSphere Commerce 계산 프레임워크 안내서를 검토하십시오. WebSphere Commerce 계산 프레임워크는 고객이 구매하려고 선택한 제품이나 서비스와 연관된 재정적 양(예: 세금)을 계산합니다.
2. 견본 상점의 상점 세금 자원을 작성하는 데 사용된 XML 파일을 검토하십시오. 견본 상점의 모든 파일은 해당 상점 아카이브 파일에 있습니다.




상점 아카이브 파일은 다음 디렉토리에 있습니다.




-  NT drive:\WebSphere\CommerceServer\samplstores
-  2000 drive:\Program Files\WebSphere\CommerceServer\samplstores
-  AIX /usr/WebSphere/CommerceServer/samplstores
-  Solaris /opt/WebSphere/CommerceServer/samplstores
-  Linux /opt/WebSphere/CommerceServer/samplstores
-  400 /qibm/proddata/WebCommerce/samplstores

주: WebSphere Commerce 온라인 도움말에는 견본 상점에 포함된 데이터 자원 각각에 대한 정보가 포함됩니다.

개별 견본 상점에는 세금 정보를 포함하는 두 개의 tax.xml 파일이 들어 있습니다. 상점 아카이브의 tax.xml 파일을 보려면 ZIP 프로그램을 사용하여 이 파일의 압축을 푸십시오. tax.xml 파일은 데이터 디렉토리에 있습니다. 언어별 tax.xml 은 데이터 디렉토리의 로케일 고유의 서브디렉토리에 있습니다.

3. 361 페이지의 부록 B 『데이터 작성』 정보를 검토하십시오.
4. 견본 상점 아카이브의 tax.xml 파일 중 하나를 복사하거나 새 파일을 작성해서 tax.xml 파일을 작성하십시오. 자세한 정보는 tax.xml의 해당 DTD 파일을 참조하십시오. DTD 파일은 다음 디렉토리에 있습니다.

-  NT drive:\WebSphere\CommerceServer\xml\sar
-  2000 drive:\Program Files\WebSphere\CommerceServer\xml\sar
-  AIX /usr/WebSphere/CommerceServer/xml/sar

-  /opt/WebSphere/CommerceServer/xml/sar
-  /opt/WebSphere/CommerceServer/xml/sar
-  /qibm/proddata/WebCommerce/xml/sar

5. 상품 및 서비스를 운송할 관할 구역 및 관할 구역 그룹을 정의하십시오. 적용 가능한 세금 카테고리 계산 규칙에 따라 세금 관할 구역을 세금 관할 구역 그룹에 지정하십시오.

a. 다음 예를 참조하여 XML 파일의 JURSTGROUP 테이블에 관할 구역 그룹을 정의하십시오.

```
< jurstgroup
jurstgroup_id="@jurstgroup_id_2"
description="Tax Jurstiction Group 1"
subclass="2"
storeent_id="@storeent_id_1"
code="World"/>
```

여기서,

- jurstgroup_id는 생성된 고유 키입니다.
- description은 관할 구역 그룹을 관리하는 사용자 인터페이스에 적절히 표시할 관할 구역 그룹에 대한 간략한 설명입니다.
- subclass는 다음과 같은 관할 구역 그룹 서브클래스입니다.
 - 1 = ShippingJurisdictionGroup
 - 2 = TaxJurisdictionGroup
- storeent_id는 이 관할 구역 그룹과 연관된 상점 엔티티입니다.
- code는 상점 엔티티 및 서브클래스와 함께 이 관할 구역 그룹을 고유하게 식별합니다.

b. 다음 예를 참조하여 XML 파일의 JURST 테이블에 관할 구역을 정의하십시오.

```
< jurst
jurst_id="@jurst_id_2"
storeent_id="@storeent_id_1"
code="World"
subclass="2"/>
```

여기서,

- jurst_id는 생성된 고유 키입니다.
- storeent_id는 이 관할 구역 그룹과 연관된 상점 엔티티입니다.

- code는 상점 엔티티 및 서브클래스와 함께 이 관할 구역 그룹을 고유하게 식별합니다.
 - subclass는 다음과 같은 관할 구역 서브클래스입니다.
 - 1 = ShippingJurisdiction
 - 2 = TaxJurisdiction
- c. 다음 예를 참조하여 JURSTGRPREL 테이블에 정보를 추가해서, b단계에서 작성한 관할 구역을 a단계에서 정의한 관할 구역 그룹과 연관시키십시오.

```
< jurstgprel
  jurst_id="@jurst_id_2"
  jurstgroup_id="@jurstgroup_id_1"
  subclass="2"/>
```

여기서,

- jurst_id는 관할 구역입니다.
- jurstgroup_id는 관할 구역 그룹입니다.
- subclass는 관할 구역 및 관할 구역 그룹의 서브클래스로 이 둘은 서로 일치해야 합니다.
 - 1 = ShippingJurisdiction[Group]
 - 2 = TaxJurisdiction[Group]

d. 상점이 지원하는 모든 관할 구역 및 관할 구역 그룹에 대해 a단계부터 c까지를 반복하십시오.

6. 상점이 사용할 세금 카테고리를 정의하십시오.

a. 다음 예를 참조하여 XML 파일의 TAXCGRY 테이블에 세금 카테고리를 정의하십시오.

```
<taxcgry
  taxcgry_id="@taxcgry_id_1"
  taxtype_id="-3"
  storeent_id="@storeent_id_1"
  name="Sales Tax"
  displayseq="0"
  displayusage="0"/>
```

여기서,

- taxcgry_id는 생성된 고유 키입니다.
- taxtype_id="-3"는 이 세금 카테고리의 세금 유형입니다. WebSphere Commerce는 다음과 같은 두 개의 세금 유형을 제공합니다.
 - 판매세 또는 사용세(-3)
 - 운송세(-4)
- storeent_id는 이 세금 카테고리와 연관된 상점 엔티티입니다.

- name은 세금 카테고리의 이름입니다. 상점 엔티티와 함께, 이름은 이 세금 카테고리를 고유하게 식별합니다.
- displayseq는 예를 들어, 주문에서 표시될 때 가장 낮은 세금에서 가장 높은 세금까지의 순서를 지정합니다.
- displayusage는 PriceDataBean과의 관계에서 이 세금 카테고리를 다음과 같이 지정합니다.
 - 0 = 계산되지 않음
 - 1 = 계산됨

PriceDataBean을 사용하여 상품 가격과 함께 표시되는 세금 총계를 얻을 수 있습니다.

- 상점에서 사용된 개별 세금 카테고리에 대해 a단계를 반복하십시오.
- 다음 예를 참조하여 XML 파일의 CALCODEDSC 테이블에 세금 카테고리 설명 정보를 추가하십시오. 다국어 지원 상점을 작성하는 경우에는 이 정보를 로케일 고유의 XML 파일에 포함시켜야 합니다.

```
<taxcgryds
  taxcgry_id="@taxcgry_id_1"
  description="Sales Tax"
  language_id("&en_US"/>
```

여기서,

- taxcgry_id는 세금 카테고리입니다.
- description은 고객에게 표시하기 위한 세금 카테고리에 대한 간략한 설명입니다.
- language_id는 이 정보가 표시될 언어입니다.


- 상점에서 사용된 개별 세금 카테고리에 대해 c단계를 반복하십시오.

7. 상점에서 사용할 계산 코드를 정의하십시오.

- 다음 예를 참조하여, XML 파일의 CALCODE 테이블에 계산 코드를 정의하십시오.

```
<calcode
calcode_id="@calcode_id_3"
code="Tax Code 1"
calusage_id="-3"
storeent_id="@storeent_id_1"
groupby="0"
published="1"
sequence="0"
calmethod_id="-43"
calmethod_id_app="-44"
calmethod_id_qfy="-42"
displaylevel="0"
flags="0"
precedence="0"
/>
```

여기서,

- calcode_id는 생성된 고유 키입니다.
- code는 계산 코드, 주어진 특정 계산 사용법 및 상점 엔티티를 고유하게 식별하는 문자열입니다.
- calusage_id는 이 계산 코드가 사용되는 계산 종류를 나타냅니다. 예를 들어, 계산 코드를 사용하여 다음과 같은 재정 총계 중 하나를 계산할 수 있습니다.
 - 할인(-1)
 - 운송 경비(-2)
 - 판매세(-3)
 - 운송세(-4)
 - 쿠폰(-5)
- storeent_id는 계산 코드와 연관된 상점 엔티티입니다.
- groupby는 계산을 수행할 때 주문 항목을 그룹으로 묶는 방법을 계산 코드 조합 방법에 나타내는 비트 플래그입니다. 0은 그룹이 없음을 지정합니다(적용 가능한 모든 주문 항목은 단일 그룹입니다). 자세한 정보는 WebSphere Commerce 온라인 도움말의 *CALCODE* 테이블: 정보를 참조하십시오.
- published는 계산 코드가 공개되었는지 여부를 지정합니다.
 - 0 = 공개되지 않음(일시적으로 사용 안함)
 - 1 = 공개
 - 2 = 삭제 표시됨(및 공개되지 않음)
- sequence는 계산 코드가 계산되는 순서입니다. 계산 코드는 가장 낮은 것에서 가장 높은 것 순서로 계산되거나 적용됩니다. 두 개의 계산 코드가 동일한 순서 번호를 갖는 경우, 하위 calcode_id를 갖는 계산 코드가 우선 계산됩니다.
- calmethod_id가 계산 코드에 대해 총 세금을 계산하는 방법을 정의하는 계산 코드 계산 방법입니다. 사용할 계산 코드 계산 방법을 결정하려면 다음을 수행하십시오.
 - WebSphere Commerce 온라인 도움말의 CALMETHOD 테이블을 참조하십시오. SUBCLASS 열의 설명을 참조하십시오. CALMETHOD 테이블: 정보에 대한 링크를 누르십시오. 이 테이블은 사용 가능한 CALMETHOD 유형을 나열합니다. 계산 코드 계산 메소드 유형은 3입니다.
 - wcs.bootstrap_xx_XX.xml 부트스트랩 파일을 여십시오. 여기서, xx_XX는 로케일 코드입니다. 부트스트랩 파일은 다음 디렉토리에 있습니다.
 -  drive:\WebSphere\CommerceServer\schema\xml

- ▶ 2000 drive:\Program Files\WebSphere\CommerceServer
 \schema\xml
- ▶ AIX /usr/WebSphere/CommerceServer/schema/xml
- ▶ Solaris /opt/WebSphere/CommerceServer/schema/xml
- ▶ Linux /opt/WebSphere/CommerceServer/schema/xml
- ▶ 400 /qibm/proddata/WebCommerce/schema/xml
- 사용 가능한 계산 방법(CALMETHOD)을 나열하는 절을 위치를 지정하
 십시오.
- 세금(판매세는 -3이고 운송세는 -4)의 calusage_ID 값과 함께 계산 방법
 을 위치 지정하십시오.
- 서브클래스 3을 가진 계산 방법을 위치 지정하십시오. 이 계산 방법은 -43
 입니다.
- calmethod_id_app는 연관된 OrderItems의 계산 총계를 저장하는
 CalculationCodeApplyMethod입니다. calmethod_id에서 설명한 방법을 사
 용하여 사용할 계산 코드 적용 방법을 결정하십시오.
 - calmethod_id_app="-44"는 판매세에 대한 CalculationCodeApplyMethod
 입니다.
- calmethod_id_qfy는 이 계산 코드와 연관되는 주문 항목을 정의하는
 CalculationCodeQualifyMethod입니다. calmethod_id에서 설명한 방법을 사
 용하여 사용할 방법을 한정하는 계산 코드를 결정하십시오.
 - calmethod_id_qfy="-42"는 판매세에 대한 CalculationCodeQualifyMethod
 입니다.
- display level은 이 계산 코드로 계산한 총계가 각각 다음과 같이 표시되
 어야 하는지를 결정합니다.
 - 0 = 주문 항목
 - 1 = 주문
 - 2 = 상품
 - 3 = 항목
 - 4 = 장기 구매 계약
- flags는 이 계산 코드의 CalculationCodeQualifyMethod가 호출되었는지 여
 부를 지정합니다.
 - 0 = 제한되지 않음. 방법이 호출되지 않습니다.
 - 1 = 제한됨. 방법이 호출됩니다.

- b. 다음 예를 참조하여, XML 파일의 CALCODEDSC 테이블에 계산 코드 설명 정보를 추가하십시오. 다국어 지원 상점을 작성하는 경우에는 이 정보를 로케일 고유의 XML 파일에 포함시켜야 합니다.

```
<calcodedsc
calcode_id="@calcode_id_3"
description="Vitamins
language_id="&en_US"
longdescription="In Ontario vitamins are taxed federally, but
not provincially."
/>
```

여기서,

- calcode_id는 이 정보를 적용할 계산 코드입니다.
- description은 계산 코드에 대한 간단한 설명입니다.
- language_id는 이 정보를 적용할 언어입니다.
- longdescription는 계산 코드에 대한 자세한 설명입니다.

- c. 상점에서 사용된 개별 계산 코드에 대해 a부터 b단계까지 반복하십시오.







8. 상점의 계산 규칙을 정의하십시오.

- a. 다음 예를 참조하여, XML 파일의 CALRULE 테이블에 계산 규칙을 설정하십시오.

```
<calrule
calrule_id="@calrule_id_10"
calcode_id="@calcode_id_3"
startdate="1900-01-01 00:00:00.000000"
taxcgr_id="@taxcgr_id_1"
enddate="2100-01-01 00:00:00.000000"
flags="1"
identifier="1"
combination="2"
calmethod_id="-47"
calmethod_id_qfy="-46"
/>
```

여기서,

- calrule_id는 생성된 고유 식별자입니다.
- calcode_id는 이 계산 규칙이 일부가 되는 계산 코드입니다.
- startdate는 이 계산 규칙이 유효하게 되는 시점입니다.
- taxcgr_id는 이 계산 규칙이 유효한 세금 카테고리입니다.
- enddate는 이 계산 규칙이 유효하지 않게 되는 시점입니다.
- combination은 CalculationRuleCombineMethod에 의해 사용되어 이 계산 규칙이 다른 계산 규칙과 결합되는 방법을 결정합니다. 자세한 정보는 CALRULE 테이블을 참조하십시오.
- identifier는 계산 코드와 조합하여 이 계산 규칙을 식별합니다.

- flags는 기본 CalculationRuleCombineMethod 구현이 수행할 특별 처리를 나타내는 비트 플래그를 지정합니다. 자세한 정보는 WebSphere Commerce 온라인 도움말의 CALRULE 테이블을 참조하십시오.
 - calmethod_id는 주문 항목 세트에 대해 재정 결과를 계산하는 CalculationRuleCalculateMethod입니다. 사용할 계산 규칙 계산 방법을 결정하려면 다음을 수행하십시오.
 - WebSphere Commerce 온라인 도움말의 CALMETHOD 테이블을 참조하십시오. SUBCLASS 열의 설명을 참조하십시오. CALMETHOD 테이블: 정보에 대한 링크를 누르십시오. 이 테이블은 사용 가능한 CALMETHOD 유형을 나열합니다. 계산 규칙 계산 방법은 7입니다.
 - wcs.bootstrap_xx_XX.xml 부트스트랩 파일을 여십시오. 여기서, xx_XX는 로케일 코드입니다. 부트스트랩 파일은 다음 디렉토리에 있습니다.
 -  NT drive:\WebSphere\CommerceServer\schema\xml
 -  2000 drive:\Program Files\WebSphere\CommerceServer\schema\xml
 -  AIX /usr/WebSphere/CommerceServer/schema/xml
 -  Solaris /opt/WebSphere/CommerceServer/schema/xml
 -  Linux /opt/WebSphere/CommerceServer/schema/xml
 -  400 /qibm/proddata/WebCommerce/schema/xml
 - 사용 가능한 계산 방법(CALMETHOD)을 나열하는 절을 위치를 지정하십시오.
 - 세금(판매세는 -3이고 운송세는 -4)의 calusage_ID 값과 함께 계산 방법을 위치 지정하십시오.
 - 서브클래스 7을 가진 계산 방법을 위치 지정하십시오. 이 계산 방법은 -47입니다.
 - calmethod_id_qfy는 CalculationRuleCalculateMethod에 보내야 하는 OrderItems 세트를 결정하는 CalculationRuleQualifyMethod입니다. calmethod_id에서 설명한 방법을 사용하여 사용할 방법을 한정하는 계산 규칙을 결정하십시오.
- b. 상점에서 사용된 개별 계산 규칙에 대해 a단계를 반복하십시오. 개별 계산 코드는 여러 계산 규칙을 가질 수 있으며, 적용 가능한 각 세금 카테고리에 대해 하나의 규칙을 갖습니다. 예를 들어, calcode_id="@calcode_id_1"는 여러 calrule_ids와 연관됩니다.
9. 상점의 계산 스케일을 정의하십시오.

계산 스케일은 계산에 적용될 범위의 세트입니다. 이런 범위가 스케일을 작성합니다.

- a. 다음 예를 참조하여 XML 파일의 CALSCALE 테이블에 계산 스케일을 설정하십시오.

```
<calscale
calscale_id="@calscale_id_19"
code="Sales Tax 1"
storeent_id="@storeent_id_1"
calusage_id="-3"
setccurr="USD"
calmethod_id="-53"
/>
```

여기서,

- calscale_id는 생성된 고유 식별자입니다.
- code는 이 계산 스케일, 주어진 특정 계산 사용법 및 상점 엔티티를 고유하게 식별하는 문자열입니다.
- storeent_id는 이 계산 스케일이 일부가 되는 상점 엔티티입니다.
- calusage_id는 이 CalculationScale이 사용되는 계산 종류를 나타냅니다. 예를 들어, CalculationScale을 사용하여 다음과 같은 재정 총계 중 하나를 계산할 수 있습니다.
 - 할인(-1)
 - 운송 경비(-2)
 - 판매세(-3)
 - 운송세(-4)
 - 쿠폰(-5)
- setccurr를 지정하면, 이 계산 스케일에 대해 계산 범위 오브젝트의 범위 시작 값에 대해 선택한 통화를 나타냅니다. CalculationScaleLookupMethod는 이 통화의 "찾아보기 번호"를 리턴해야 합니다. 이 경우, setccurr이 지정되지 않으며, CalculationScaleLookupMethod가 주문에서의 통화로 찾아보기 번호를 리턴합니다. 스케일 범위 시작 값이 0이 아닌 경우를 제외하고는 통화를 지정할 필요가 없습니다.
- calmethod_id는 주어진 주문 항목 세트가 찾아보기 번호, 기본 재정 값, 결과 곱셈 및 재정 총계를 계산하는 데 계산 스케일이 사용할 수학적 중량 세트를 결정하는 CalculationScaleLookupMethod입니다. 사용할 CalculationScaleLookupMethod를 결정하려면 다음을 수행하십시오.
 - WebSphere Commerce 온라인 도움말의 CALMETHOD 테이블을 참조하십시오. SUBCLASS 열의 설명을 참조하십시오. CALMETHOD 테이블: 정보에 대한 링크를 누르십시오. 이 테이블은 사용 가능한

CALMETHOD 유형을 나열합니다.

MonetaryCalculationScaleLookupMethod 방법은 9입니다.

- wcs.bootstrap_xx_XX.xml 부트스트랩 파일을 여십시오. 여기서, xx_XX는 로케일 코드입니다. 부트스트랩 파일은 다음 디렉토리에 있습니다.
 -  NT drive:\WebSphere\CommerceServer\schema\xml
 -  2000 drive:\Program Files\WebSphere\CommerceServer\schema\xml
 -  AIX /usr/WebSphere/CommerceServer/schema/xml
 -  Solaris /opt/WebSphere/CommerceServer/schema/xml
 -  Linux /opt/WebSphere/CommerceServer/schema/xml
 -  400 /qibm/proddata/WebCommerce/schema/xml
- 사용 가능한 계산 방법(CALMETHOD)을 나열하는 절을 위치를 지정하십시오.
- 세금(판매세는 -3이고 운송세는 -4)의 calusage_ID 값과 함께 계산 방법을 위치 지정하십시오.
- 서브클래스 9를 가진 계산 방법을 위치 지정하십시오. 서브클래스 9에는 여러 계산 방법이 있습니다. 필요에 부합하는 것을 고르십시오.

자세한 정보는 WebSphere Commerce 온라인 도움말의 CALSCALE 테이블을 참조하십시오.

- b. 상점에서 사용된 개별 계산 스케일에 대해 a단계를 반복하십시오.
- c. 다음 예를 참조하여 XML 파일의 CALSCALDS 테이블에 계산 스케일 설명 정보를 추가하십시오. 다국어 지원 상점을 작성하는 경우에는 이 정보를 로케일 고유의 XML 파일에 포함시켜야 합니다.

```
<calscaleds
calscale_id="@calscale_id_19"
description="Sales Tax 5% "
language_id="&en_US"
/>
```

여기서,

- calscale_id는 이 설명이 적용될 계산 스케일입니다.
- description은 계산이 수행되는 방식을 설명하기 위해 고객에게 표시하기 위한 계산 스케일에 대한 간략한 설명입니다. 예를 들어, "킬로그램당 \$.10, \$5.00의 최소 비용" 또는 "5개 이상의 10% 할인 수량"이 있습니다.
- language_id는 이 정보가 표시될 언어입니다.







- d. 상점에서 사용된 개별 계산 스케일에 대해 c단계를 반복하십시오.

10. 계산 스케일의 계산 범위를 정의하십시오.

- a. 다음 예를 참조하여, XML 파일의 CALRANGE 테이블에 계산 범위를 설정하십시오.

```
<calrange
calrange_id="@calrange_id_37"
calscale_id="@calscale_id_19"
calmethod_id="-59"
rangestart="0.00000"
cumulative="0"
/>
```

여기서,

- calrange_id는 생성된 고유 식별자입니다.
- calscale_id는 이 계산 범위가 일부분이 되는 계산 스케일입니다.
- calmethod_id는 CalculationRangeLookupResult로부터 재정 총계를 결정하는 CalculationRangeMethod입니다. 예를 들어, FixedAmountCalculationRangeCmd, PerUnitAmountCalculationRangeCmd 또는 PercentageCalculationRangeCmd입니다. CalculationRangeMethod를 결정하려면, 다음을 수행하십시오.
 - WebSphere Commerce 온라인 도움말의 CALMETHOD 테이블을 참조하십시오. SUBCLASS 열의 설명을 참조하십시오. CALMETHOD 테이블: 정보에 대한 링크를 누르십시오. 이 테이블은 사용 가능한 CALMETHOD 유형을 나열합니다. CalculationRangeMethod는 10입니다.
 - wcs.bootstrap_xx_XX.xml 부트스트랩 파일을 여십시오. 여기서, xx_XX는 로케일 코드입니다. 부트스트랩 파일은 다음 디렉토리에 있습니다.
 -  drive:\WebSphere\CommerceServer\schema\xml
 -  drive:\Program Files\WebSphere\CommerceServer\schema\xml
 -  /usr/WebSphere/CommerceServer/schema/xml
 -  /opt/WebSphere/CommerceServer/schema/xml
 -  /opt/WebSphere/CommerceServer/schema/xml
 -  /qibm/proddata/WebCommerce/schema/xml
 - 사용 가능한 계산 방법(CALMETHOD)을 나열하는 절을 위치를 지정하십시오.
 - 세금(판매세는 -3이고 운송세는 -4)의 calusage_ID 값과 함께 계산 방법을 위치 지정하십시오.

- 서버클래스 10을 가진 계산 방법을 위치 지정하십시오. 서버클래스가 10인 여러 계산 방법이 있습니다. 사용자 필요에 맞게 하나를 오더피킹하십시오.
- 찾아보기 번호가 RANGESTART 이상이거나, RANGESTART가 NULL인 경우, rangestart 행은 찾아보기 숫자와 일치합니다.
- cumulative는 다음입니다.
 - 0 = 가장 높은 RANGESTART 값과 일치하는 CalculationRange만 사용됩니다.
 - 1 = 일치하는 모든 CalculationRanges가 사용됩니다. 계산된 재정 총계는 최종 결과에 합계됩니다.

자세한 정보는 WebSphere Commerce 온라인 도움말의 CALRANGE 테이블을 참조하십시오.

- b. 상점에서 사용된 계산 스케일과 연관된 개별 계산 범위에 대해 a단계를 반복하십시오. 위의 예에서 모든 총계는 동일한 비율로 세금이 계산되므로 오직 하나의 범위만이 있습니다.

11. 계산 스케일에 대해 계산 찾아보기 값을 정의하십시오. 계산 찾아보기 값은 계산 스케일과 연관된 값입니다. 예를 들어, 레스토랑에서 제공하는 식사에 대한 온타리오 지방 판매세에 대해 다음과 같은 범위와 연관된 세율이 계산 스케일에 포함됩니다.

- 0.00% 세율의 \$0.00 - \$3.99
- 8.00% 세율의 \$4.00 이상

찾아보기 값은 0.00 및 8.00입니다.

- a. 다음 예를 참조하여 XML 파일의 CALRLOOKUP 테이블에 계산 찾아보기를 설정하십시오.

```
<calrlookup
  calrlookup_id="@calrlookup_id_37"
  calrange_id="@calrange_id_37"
  value="5.00"
/>
```

여기서,

- calrlookup_id는 생성된 고유 식별자입니다.
- calrange_id는 이 계산 범위 찾아보기 결과가 일부분이 되는 계산 범위입니다.
- value는 재정 결과를 결정하는 데 계산 범위의 계산 범위 방법이 사용한 계산 범위 찾아보기 결과의 값입니다. 이 예에서, 세율은 5.00%입니다.

자세한 정보는 WebSphere Commerce 온라인 도움말의 CALRLOOKUP 테이블을 참조하십시오.

- b. 상점에서 사용된 계산 스케일과 연관된 개별 찾아보기 값에 대해 a 와 b단계를 반복하십시오. 이 예에서 CALRLOOKUP.SETCCURR은 NULL이므로 오직 하나의 CALRANGE가 있으며, 모든 총계에 대한 세율이 동일하므로 오직 하나의 CALRANGE가 있습니다.

12. 계산 규칙과 계산 스케일을 연관시키십시오.

- a. 다음 예를 참조하여 XML 파일의 CRULESCALE 테이블에 있는 계산 규칙을 계산 스케일과 연관시키십시오.

```
<crulescale
  calrule_id="@calrule_id_10"
  calscale_id="@calscale_id_19"
/>
```

여기서,

- calrule_id는 계산 규칙입니다.
 - calscale_id는 계산 스케일입니다.
- b. 개별 계산 스케일 및 규칙 연관에 대해 a단계를 반복하십시오. 위에서 사용된 예에서, 각 계산 규칙에는 오직 하나의 계산 스케일이 있습니다.

주: 세율이 구입 총계에 따라 다양한 경우, 0이 아닌 범위 시작 값으로 스케일을 작성할 필요가 있습니다. 그런 다음 변환 비율(CURCONVERT 테이블을 나타냄)을 설정하지 않은 각 지원 통화(CALSCALE.SETCCURR을 해당 통화로 설정)에 대한 계산 스케일을 작성하여 해당 특정 세금 카테고리에 대한 계산 규칙과 연관시키는 것이 필요합니다. 예를 들어, \$4.00 미만의 식사에 대한 온타리오 지방 판매세는 없습니다. 상점에서 미국 달러로 식사 판매를 지원하는 경우, 미국 달러에서 캐나다 달러로 변환을 설정하거나 \$6.00 USD의 해당 범위 시작 값으로 각 세금 계산 스케일을 작성하고 동일한 세금 계산 규칙으로 연관시키는 것이 필요할 수 있습니다. 주문 통화에 따라 해당 계산 스케일만이 사용됩니다.



@ 및 & 사용에 대한 자세한 정보는 361 페이지의 부록 B 『데이터 작성』을 참조하십시오.

세금 이행 자원 작성




세금 자원이 상점에서 올바르게 작업하게 하려면, 상점의 세금 관할 구역 그룹을 상점이 사용하는 서비스 센터와 연관시킨 다음, 계산 규칙을 이 둘과 연관시켜야 합니다.

세금 자원을 서비스 센터에 연관시키기 전에 이행 자원을 작성해야 합니다. 이행 자원 작성에 대한 자세한 정보는 124 페이지의 『WebSphere Commerce에서 이행 자원 작성』을 참조하십시오.

이행 자원을 작성한 후, TAXJCRULE 테이블에 정보를 추가해서 세금 자원을 이행 자원과 연관시키십시오. 다음을 수행하십시오.

1. IBM WebSphere Commerce 계산 프레임워크 안내서를 검토하십시오. WebSphere Commerce 계산 프레임워크는 고객이 구매하려고 선택한 제품이나 서비스와 연관된 재정적 양(예: 세금)을 계산합니다.
2. 견본 상점의 세금 자원을 작성하는 데 사용된 XML 파일을 검토하십시오. 견본 상점의 모든 파일은 해당 상점 아카이브 파일에 있습니다.

상점 아카이브 파일은 다음 디렉토리에 있습니다.

-  NT drive:\WebSphere\CommerceServer\samplstores
-  2000 drive:\Program Files\WebSphere\CommerceServer\samplstores
-  AIX /usr/WebSphere/CommerceServer/samplstores
-  Solaris /opt/WebSphere/CommerceServer/samplstores
-  Linux /opt/WebSphere/CommerceServer/samplstores
-  400 /qibm/proddata/WebCommerce/samplstores

주: WebSphere Commerce 온라인 도움말에는 견본 상점에 포함된 데이터 자원 각각에 대한 정보가 포함됩니다.

개별 견본 상점에는 세금 정보를 포함하는 taxfulfill.xml 파일이 들어 있습니다. 상점 아카이브의 taxfulfill.xml 파일을 보려면, ZIP 프로그램을 사용하여 이 파일의 압축을 푸십시오. taxfulfill.xml 파일은 데이터 디렉토리에 있습니다.

3. 361 페이지의 부록 B 『데이터 작성』 정보를 검토하십시오.
4. 견본 상점 아카이브의 taxfulfill.xml 파일 중 하나를 복사하거나 새 파일을 작성해서 taxfulfill.xml 파일을 작성하십시오. 자세한 정보는 taxfulfill.xml의 해당 DTD 파일을 참조하십시오. DTD 파일은 다음 디렉토리에 있습니다.

-  NT drive:\WebSphere\CommerceServer\xml\sar
-  2000 drive:\Program Files\WebSphere\CommerceServer\xml\sar
-  AIX /usr/WebSphere/CommerceServer/xml/sar
-  Solaris /opt/WebSphere/CommerceServer/xml/sar
-  Linux /opt/WebSphere/CommerceServer/xml/sar
-  400 /qibm/proddata/WebCommerce/xml/sar

5. 다음 예를 참조하여 XML 파일에 TAXJCRULE 테이블에 정보를 추가하십시오.

```
<taxjcrule
taxjcrule_id="@taxjcrule_id_1"
calrule_id="@calrule_id_10"
```

```
ffmcenter_id="@ffmcenter_id_1"
jurstgroup_id="@jurstgroup_id_2"
precedence="0"
/>
```

여기서,

- taxjcrule_id는 생성된 고유 식별자입니다.
 - calrule_id는 사용된 계산 규칙입니다.
 - ffmcenter_id는 서비스 센터입니다. 이것이 널(NULL)값이면 이 연관은 모든 서비스 센터에 적용됩니다.
 - jurstgroup_id는 세금 관할 구역 그룹입니다. 이것이 널(NULL)값이면 이 연관은 모든 세금 관할 구역 그룹에 적용됩니다.
 - precedence는 운송 주소가 둘 이상의 지정된 세금 관할 구역 그룹 내에 있지 않을 때, 같은 서비스 센터에 대해 가장 높은 TAXJCRULE.PRECEDENCE 값 조건이 있는 계산 규칙만입니다.
6. 상점의 개별 관할 구역 그룹, 서비스 센터 및 규칙 연관에 대해 3단계를 반복하십시오.



@ 및 & 사용에 대한 자세한 정보는 361 페이지의 부록 B 『데이터 작성』을 참조하십시오.

상점-카탈로그-세금 자원 작성



세금을 상점의 상품 및 서비스와 연관시키려면, 상점이 포함하는 개별 장기 구매 계약에 대해 계산 코드를 상점의 카탈로그 항목과 연관시켜야 합니다.

상점-카탈로그-세금 자원을 작성하기 전에 상점 및 카탈로그 자원을 작성해야 합니다. 상점 자원 작성에 대한 자세한 정보는 47 페이지의 『XML 파일에 상점 데이터 자원 작성』을 참조하십시오. 카탈로그 자원 작성에 대한 자세한 정보는 84 페이지의 『상점 카탈로그 자원 표시』를 참조하십시오.

상점-카탈로그-세금 자원을 작성하려면, 다음을 수행하십시오.

1. *IBM WebSphere Commerce 계산 프레임워크 안내서*를 검토하십시오. WebSphere Commerce 계산 프레임워크는 고객이 구매하려고 선택한 제품이나 서비스와 연관된 재정적 양(예: 운송 비용)을 계산합니다.
2. 견본 상점의 상점-카탈로그-세금 자원을 작성하는 데 사용된 XML 파일을 검토하십시오. 견본 상점의 모든 파일은 해당 상점 아카이브 파일에 있습니다.

상점 아카이브 파일은 다음 디렉토리에 있습니다.

-  drive:\WebSphere\CommerceServer\samplstores
-  drive:\Program Files\WebSphere\CommerceServer\samplstores

- ▶ **AIX** /usr/WebSphere/CommerceServer/samplestores
- ▶ **Solaris** /opt/WebSphere/CommerceServer/samplestores
- ▶ **Linux** /opt/WebSphere/CommerceServer/samplestores
- ▶ **400** /qibm/proddata/WebCommerce/samplestores

개별 견본 상점에는 운송 이행 정보를 포함하는 store-catalog-tax.xml 파일이 들어 있습니다. 상점 아카이브의 store-catalog-tax.xml 파일을 보려면, ZIP 프로그램을 사용하여 이 파일의 압축을 푸십시오. store-catalog-tax.xml 파일은 데이터 디렉토리에 있습니다.

주: WebSphere Commerce 온라인 도움말에는 견본 상점에 포함된 데이터 자원 각각에 대한 정보가 포함됩니다.

- 361 페이지의 부록 B 『데이터 작성』 정보를 검토하십시오.
- 견본 상점 아카이브의 store-catalog-tax.xml 파일 중 하나를 복사하거나 새 파일을 작성해서 store-catalog-tax.xml 파일을 작성하십시오. 자세한 정보는 store-catalog-tax.xml에 해당 DTD 파일을 참조하십시오. DTD 파일은 다음 디렉토리에 있습니다.

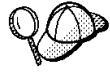
- ▶ **NT** drive:\WebSphere\CommerceServer\xml\sar
- ▶ **2000** drive:\Program Files\WebSphere\CommerceServer\xml\sar
- ▶ **AIX** /usr/WebSphere/CommerceServer/xml/sar
- ▶ **Solaris** /opt/WebSphere/CommerceServer/xml/sar
- ▶ **Linux** /opt/WebSphere/CommerceServer/xml/sar
- ▶ **400** /qibm/proddata/WebCommerce/xml/sar

- CATENCALCD 테이블에 정보를 추가해서 상점-카탈로그-세금 관계를 작성하십시오. 다음 예를 안내서로 사용하십시오.

```
<catencalcd
  calcode_id="@calcode_id_3"
  catencalcd_id="@catencalcd_id_3"
  store_id="@storeent_id_1"
/>
```

여기서,

- calcode_id는 계산 코드입니다.
- catencalcd_id는 생성된 고유 식별자입니다.
- store_id는 상점입니다.



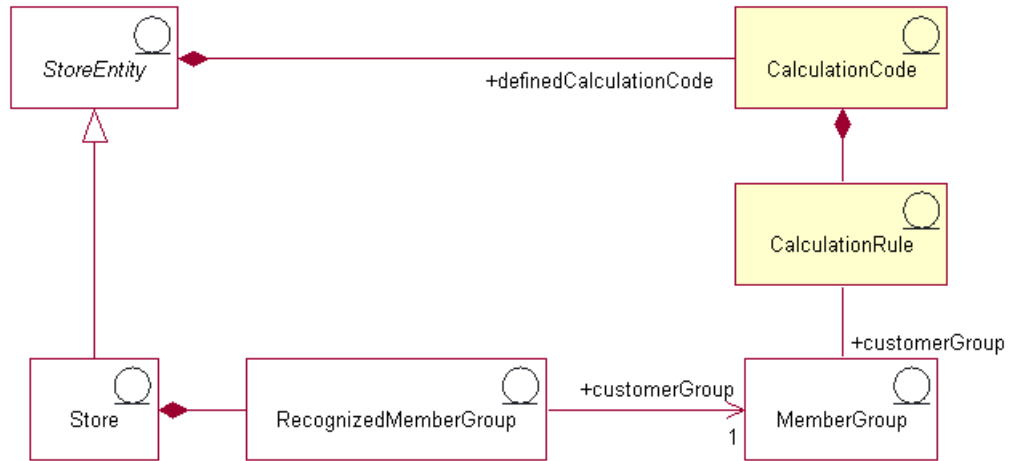
@ 및 & 사용에 대한 자세한 정보는 361 페이지의 부록 B 『데이터 작성』을 참조하십시오.

제 20 장 할인 지원

할인을 사용하면 고객에게 가격 인센티브를 제공하여 구매를 증가시킬 수 있습니다. 백분을 할인(예: 10% 할인) 또는 고정 금액 할인(예: \$15 할인)을 제공할 수 있습니다. 특정 상품 또는 구매 총계에 할인을 적용할 수 있습니다. 예를 들어, 노인 고객을 대상으로 20% 할인을 제공할 수도 있고 빨간색 야구 모자 재고가 많은 경우 제한 시간 동안 이 모자를 25% 할인하여 판매할 수 있습니다.

WebSphere Commerce의 할인 이해

다음 도표는 WebSphere Commerce 서버의 할인 구조를 보여줍니다.



이 도표와 상점 데이터 절의 모든 도표들은 WebSphere Commerce 서버 정보 모델의 부분입니다. 정보 모델에 대한 자세한 정보는 29 페이지의 『상점 데이터 정보 모델』을 참조하십시오. 이 도표에서 사용된 규칙에 대한 자세한 정보는 359 페이지의 부록 A 『UML 범례』를 참조하십시오.

계산 코드

할인은 할인 계산 코드를 사용하여 계산되고 표시됩니다. 할인 계산 코드는 주문 항목에 대해 할인을 계산하는 법을 표시합니다.

계산 코드는 상점 엔티티에 속합니다. 복수 계산 코드는 상점 엔티티 내에서 정의될 수 있습니다. 상점 엔티티가 삭제되면, 해당 상점 엔티티 내에 정의되어 계산 코드도 삭제됩니다.

각 할인 계산 코드는 할인이 유효한 기간을 정의한 시작 날짜와 종료 날짜를 가질 수 있습니다. 할인 계산 코드는 적합한 구성원 그룹을 정의하는 하나 이상의 구성원 그룹과 연관될 수 있습니다.

할인 계산 코드는 하나 이상의 카탈로그 항목과 카탈로그 그룹에 첨부될 수 있습니다. 하나의 계산 코드를 하나의 카탈로그 그룹에 첨부하면 하나의 계산 코드를 카탈로그 그룹의 카탈로그 항목에 직접 첨부하는 것과 같은 효과가 있습니다. 그러나 카탈로그 그룹 A에 첨부된 할인 계산 코드는 카탈로그 그룹 A가 카탈로그 그룹 B에 속하는 경우, 카탈로그 그룹 B의 항목과 상품에 첨부되지 않습니다.

카탈로그 항목 또는 카탈로그 그룹은 이들과 연관된 하나 이상의 할인을 가질 수 있습니다. 하나 이상의 할인 계산 코드를 하나의 주문에 적용 가능할 때, 할인 계산은 할인 코드 순서 속성의 오름차순 순서로 수행됩니다.

주: 할인을 구현하기 위한 할인 순서 주문을 할인에서 정의하십시오.

주문 항목은 다음 중 하나의 계산 방법을 따라 그룹으로 묶입니다.

- 거래 계약별
- 상품별
- 판매 설정별
- 운송 주소별

자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오.



계산 코드에 관한 자세한 정보는 *IBM WebSphere Commerce 계산 프레임워크 안* 내서를 참조하십시오.

계산 규칙

각 계산 코드는 계산이 이루어지는 조건을 정의하는 계산 규칙 세트를 갖고 있습니다. 각 할인 계산 규칙은 할인이 유효한 하나 이상의 구성원 그룹과 연관됩니다. 구성원 그룹은 한 번에 하나 이상 할인해주는 것이 좋습니다.

주: 적합한 구성원 그룹이 계산 코드 레벨에서 정의되는 경우, 계산 규칙 레벨에서 다시 정의될 필요가 없습니다.



계산 규칙에 관한 자세한 정보는 *IBM WebSphere Commerce 계산 프레임워크 안* 내서를 참조하십시오.

WebSphere Commerce에서 할인 자원 작성

WebSphere Commerce로 작성된 상점에서 할인을 작성하는 기본적인 방법은 WebSphere Commerce 액셀러레이터의 할인 마법사를 사용하는 것입니다. WebSphere Commerce 액셀러레이터를 사용하는 할인 작성에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오.

할인은 XML 파일을 사용하여 작성할 수도 있고, 로더 패키지를 로드하거나 상점 서비스를 공개하여 작성할 수도 있습니다. 그러나 이런 식으로 작성된 할인은 이전 버전에서 이주하는 동안 반입된 할인과 마찬가지로 올바르게 기능하지만, WebSphere Commerce 액셀러레이터에서는 적절하게 표시되지 않을 수도 있습니다.

제 21 장 재고 지원

재고에는 서비스 센터에서 물리적으로 계산할 수 있는 모든 것이 포함됩니다. 항목, 상품, SKU, 번들 및 패키지와 같이 서비스 대상이 되는 재고 유형에 대한 특정 정의가 있습니다. 그러나 이들은 모두 재고로 간주됩니다. 상품 노트북과 상품 마법사의 이행에 대한 상품이 구성됩니다. 여기에는 재고 추적, 백오더 허용, 백오더 강제 실행, 별도 출고 및 상품 반환 금지 지정과 같은 옵션이 포함됩니다. WebSphere Commerce 액셀러레이터는 받을 수 있는 두 개의 주요한 재고 유형을 다음과 같이 구별합니다.

- 연관 예상 재고 레코드를 갖고 있는 예상 재고
- 임시 재고 또는 예상 재고로 기록되지 않은 재고

예상 재고는 공급업체로부터 수령하며, 일반적으로 구매 주문에 따라 대금을 지불합니다. WebSphere Commerce 액셀러레이터는 예상 재고 레코드에 따라 예상 재고를 추적하며, 사용자가 외부 식별자와 일반적으로 외부 시스템을 사용하여 구매 주문 번호를 기록하게 합니다. 이러한 방법을 사용하여, 도착했거나 도착하지 않은 것은 물론 주문한 재고 추적을 용이하게 유지할 수 있습니다. 예상 재고 정보는 상품, 예상 수령 날짜, 예상 수량 및 설명을 예상하는 서비스 센터와 마찬가지로 예상 재고 레코드의 상품에 대한 고유한 정보입니다.

예상 재고 레코드는 그에 대해 재고를 수령하면 삭제할 수 없으며 재고를 수령한 후에는 예상 재고 정보를 변경 또는 삭제할 수 없습니다.

서비스 센터에서 사용 가능한 재고에 대해 주문을 제출하면, 주문 시스템은 재고를 이러한 주문으로 할당합니다. 재고를 주문으로 할당하면 주문 시스템에서 사용할 수 없습니다. 주문이 취소되면 재고는 다시 사용 가능해집니다. 사용할 수 없는 재고에 주문이 제출되면 백오더가 작성됩니다. 백오더를 서비스하는 데 사용할 수 있는 예상 재고가 있으면, 예상 재고는 백오더에 할당되고, 고객은 예상 운송 날짜를 제공받을 수 있습니다.

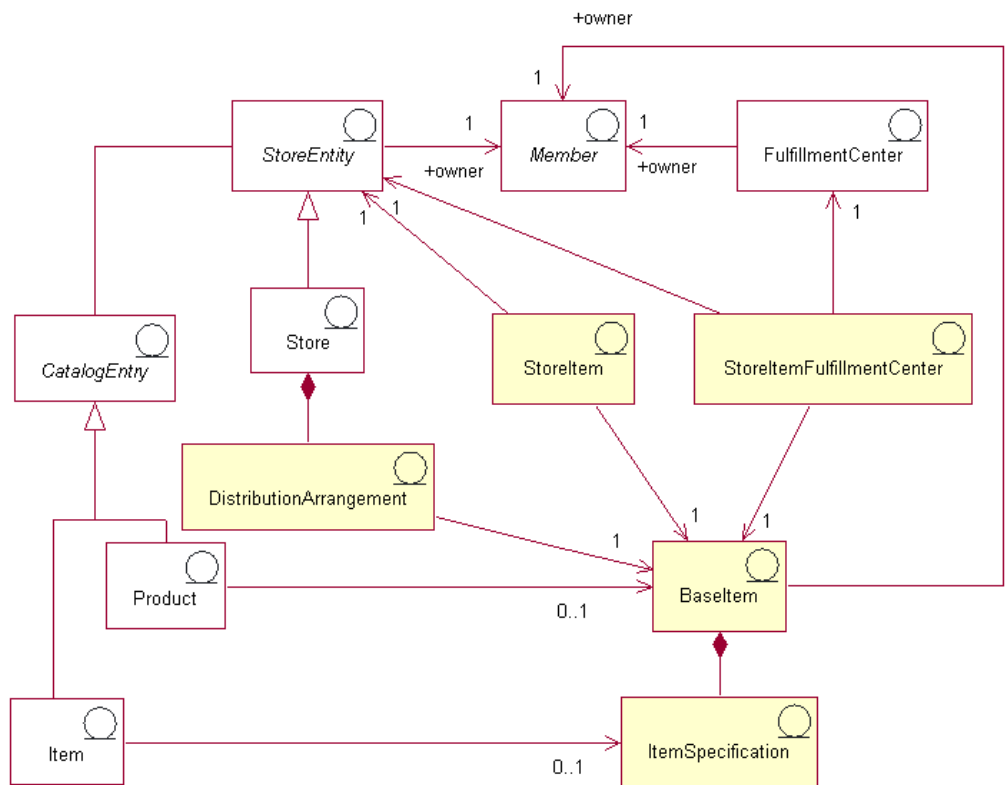
임시 재고 수령증은 해당 예상 재고 레코드 없이 재고가 서비스 센터에 도착할 때 작성됩니다. 이것은 예상치 못한 재고가 발생하거나, 재고 수령을 기록할 예상 재고 레코드를 사용하지 않기로 판매자가 선택할 수 있기 때문입니다.

주: 재고 수령이 예상 또는 임시 어느쪽이든, 수령하기 위해서는 상품이 WebSphere Commerce 시스템에 있어야 합니다.

WebSphere Commerce의 재고 자원 이해

재고 자원을 이해하려면, 재고와 상점간의 관계를 이해해야 합니다. 이것은 정보 모델을 사용하여 설명할 수 있습니다. 다음 절은 상점과 다른 자원들에 대해 관련 재고가 갖고 있는 관계에 대해 설명합니다. 아래 도표는 ATP 재고 및 ATP가 아닌 재고에 대한 관계와 연관을 설명합니다. ATP가 아닌 재고는 상품의 이전 버전을 재고로 처리한 방법이며, 상점이 ATP 기능을 사용하지 않기로 결정할 경우 사용할 수 있습니다. 각 도표와 그 연관은 아래에서 설명됩니다. ATP에 대한 자세한 정보는 온라인 도움말을 참조하십시오.

ATP 재고



이 도표와 상점 데이터 절의 모든 도표들은 WebSphere Commerce 서버 정보 모델의 부분입니다. 정보 모델에 대한 자세한 정보는 29 페이지의 『상점 데이터 정보 모델』을 참조하십시오. 이 도표에서 사용된 규칙에 대한 자세한 정보는 359 페이지의 부록 A 『UML 범례』를 참조하십시오.

기본 항목

기본 항목은 재고 도표의 중심이며 공통 이름 및 설명과 함께 상품의 일반 계열을 나타냅니다. 기본 항목들은 이행에만 독점적으로 사용되며 모든 상점에 특정한 것은 아닙니다. 카탈로그의 상품을 나타내는 각 카탈로그 항목은 이행용 해당 기본 항목을 가지고 있습니다. 기본 항목들은 BASEITEM 테이블에서 정의됩니다.

항목 스펙

항목 스펙은 자체의 모든 속성에 대해 정의된 값을 가진 기본 항목입니다. 카탈로그의 항목을 나타내는 각 카탈로그 항목은 이행용 해당 기본 스펙을 가지고 있습니다.

카탈로그 항목

상품 및 항목은 카탈로그 항목입니다. 카탈로그 항목들은 상품 및 항목과 같이 카탈로그 항목을 의미하는 상점 엔티티와 연관되어 있으며, 상점에 있습니다.

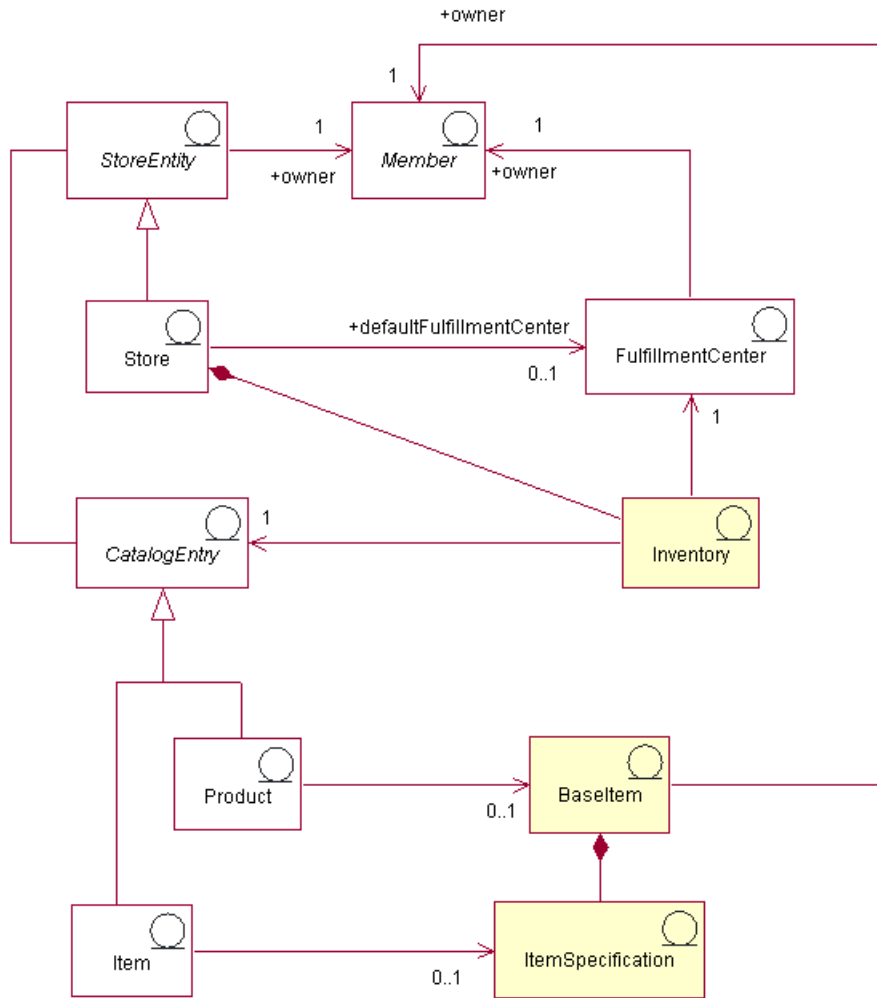
분배 정렬

분배 정렬은 기본 항목과 연관된 것으로, 상점이 자신의 재고를 판매할 수 있게 합니다. 분배 정렬은 DISTARRANG 테이블에 저장됩니다.

상점 항목

상점 항목은 백오더 허용 및 재고 추적 여부를 포함하여, 특정 상점이나 상점 그룹이 특정 기본 항목의 지정된 항목에 대해 재고를 할당하는 방법에 영향을 주는 속성을 나타냅니다. STORITMFFC 테이블은 주문 항목이 이행으로 출고될 때부터 고객에게 운송될 때까지의 시간을 정의합니다. 이 테이블은 상점이 상점 항목에 대해 FFMCENTER 기본 운송 오프셋으로 대체 속성을 정의하고자 할 경우에만 입력할 수 있습니다.

비ATP 재고



이 도표와 상점 데이터 질의 모든 도표들은 WebSphere Commerce 서버 정보 모델의 부분입니다. 정보 모델에 대한 자세한 정보는 29 페이지의 『상점 데이터 정보 모델』을 참조하십시오. 이 도표에서 사용된 규칙에 대한 자세한 정보는 359 페이지의 부록 A 『UML 범례』를 참조하십시오.

기본 항목 또한 ATP가 아닌 재고 도표의 중심입니다. 상품, 항목 및 카탈로그 항목에 대한 기본 항목의 관계는 일반 재고 도표에 대한 관계와 동일합니다. 기본 항목은 여전히 구성원이 소유하고 있으며, 일단 해당 구성원이 정의하면 상점에서 판매될 수 있습니다. 그러나 이런 경우에는 분배 정렬, 상점 항목 연관 또는 상점 항목 서비스 센터가 없습니다.

서비스 센터

재고는 하나의 서비스 센터 및 하나의 상점과 연관되어 있습니다. 하나의 상점은 하나의 기본 서비스 센터를 지정합니다. 기본 항목과 마찬가지로, 서비스 센터는 구성원이 소유하고 있으며, 상점과 소유권을 공유합니다. 이행 자원에 대한 자세한 정보는 121 페이지의 제 11 장 『이행 자원』을 참조하십시오.



WebSphere Commerce 서버의 재고 자원의 구조에 대한 자세한 정보는
WebSphere Commerce 온라인 도움말의 재고 오브젝트 및 데이터 모델을 참조하
십시오.

WebSphere Commerce에서 재고 자원 작성

재고는 운영 데이터이기 때문에, 고객이 상점에서 상품을 구매하거나 항목을 상점에 반
품할 때 매일 변경됩니다. 그 결과 재고 수준은 상품을 팔 때마다 그리고 서비스 센터
가 공급자로부터 새 재고를 수령할 때마다 올라가거나 내려갑니다. WebSphere
Commerce 액셀러레이터를 사용하면 다음 재고 관련 태스크를 완료할 수 있습니다.

- 예상 재고 기록
- 공급업체로부터 예상 재고 및 임시 재고 수령
- 재고 조정
- 반품 레코드 유지보수
- 반품 이유 유지보수
- 고객으로부터 반품된 재고 수령
- 반품된 재고 처리 관리

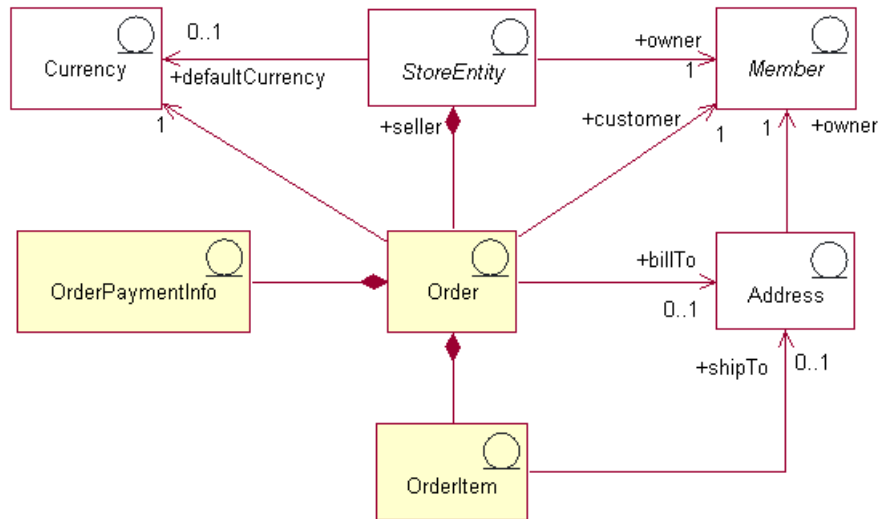
WebSphere Commerce 액셀러레이터를 사용하여 재고 관리에 대한 자세한 정보는
WebSphere Commerce 온라인 도움말을 참조하십시오.

제 22 장 주문 자원

WebSphere Commerce 시스템의 주문 자원은 장비구니, 주문 관리 및 주문 처리 기능을 제공합니다. 주문 처리 기능으로는 빠른 주문 또는 구매, 계획된 주문, 보류 중인 복수 주문, 재주문 그리고 주문 나누기 또는 백오더가 있습니다. 가격 책정, 세금 계산, 지불, 재고 및 이행과 같은 관련 서비스도 주문 자원의 부분입니다.

WebSphere Commerce의 주문 자원 이해

다음 도표는 WebSphere Commerce 서버의 주문 자원을 보여줍니다. 각 주문에 대한 설명은 도표 뒤에 나옵니다.



이 도표와 상점 데이터 절의 모든 도표들은 WebSphere Commerce 서버 정보 모델의 부분입니다. 정보 모델에 대한 자세한 정보는 29 페이지의 『상점 데이터 정보 모델』을 참조하십시오.

주문 및 주문 항목

WebSphere Commerce 시스템에서, 고객 또는 구매자일 경우, 주문은 선택한 상품 목록이고(예를 들어, 하나의 주문에 두 권의 책과 하나의 CD가 포함될 수 있음), 해당 목록에 있는 각 상품은 주문 항목입니다(예를 들어, 책과 CD가 각각 동일한 주문의 주문 항목임). 고객이 상점에서 주문을 제출하면, 고객은 상점이 인보이스를 보내는 지불 청구 주소를 제공해야 합니다. 단일 통화 식별자는 각 주문과 연관됩니다. 상점 측에서 볼 때 주문은 주문 항목의 목록입니다. 이것은 상점의 데이터의 부분입니다.

통화

상점은 하나의 통화로 가격을 표시하거나 여러 통화를 사용할 수 있습니다. 각 상점은 기본 통화를 정의해야 합니다. 또 고객이 구매 통화를 선택하도록 할 수 있습니다. 구매 통화가 상점의 기본 통화와 동일하면, 이 통화는 STOREENT 테이블에 이미 지원되어 있습니다. 구매 통화가 상점의 기본 통화가 아닌 경우에는 CURLIST 테이블에 통화를 추가해야 합니다. 고객은 구매 통화를 사용하여 상점에서 주문을 합니다.

지불 정보

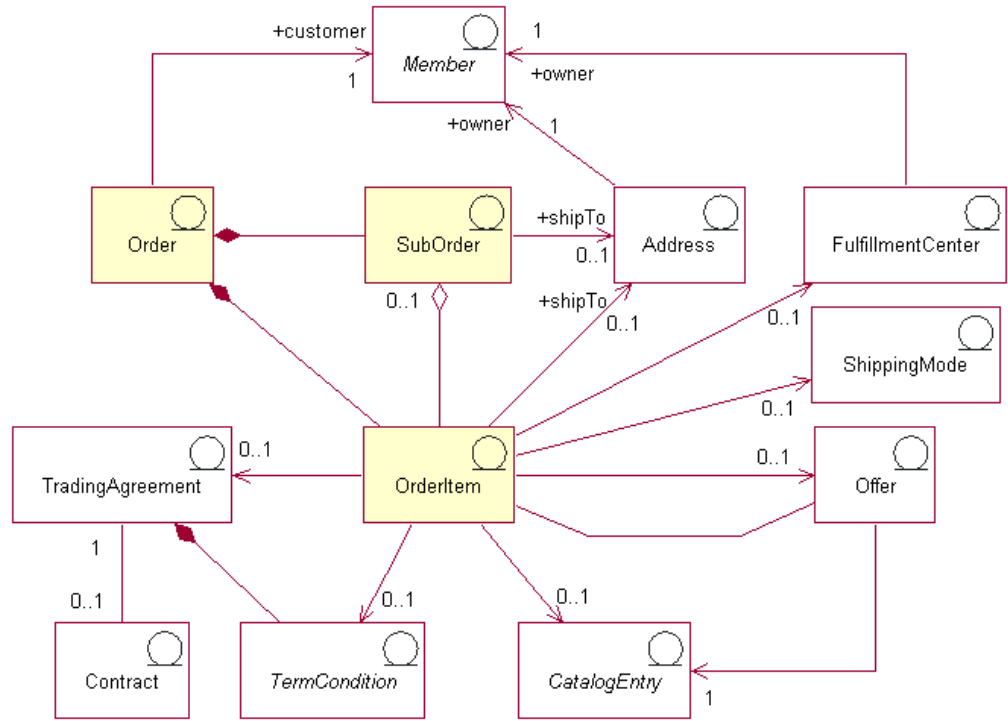
고객이 선호하는 구매 통화를 선택하면 모든 지불이 그 통화로 처리됩니다. 상점의 지불 지원 및 정책에 따라, 고객은 온라인 지불(고객이 상점의 사이트에서 인터넷을 통해 지불 정보를 제공) 또는 오프라인 지불(고객이 인터넷 채널 없이 전화 또는 팩스를 통해 지불 정보 제공)을 사용하여 구매 상품에 대해 지불할 수 있습니다. 온라인 또는 오프라인 지불 방법과 무관하게, 고객은 다음 중 한 방법을 사용하여 주문 제출 시 지불 정보를 제공해야 합니다.

- 지불 방법: 주문에 대한 고객의 지불 방법. 상점의 Payment Manager에서 구성된 지불 카세트에 따라 상점이 오프라인 지불, 온라인 지불용 CyberCash, 고객에게 온라인 Wallet을 사용하도록 요청하지 않는 온라인 지불용 SET(Secure Electronic Transaction) 및 MIA(Merchant Initiated Authorization) 또는 사용자 지정 지불 방법을 승인하도록 설정할 수 있습니다.
- 신용 카드 지불의 경우 카드 정보: 고객의 신용 카드 종류, 번호 그리고 주문에 대한 지불로 사용되는 만기 날짜. 신용 카드 정보는 일반적으로 상점이 온라인 지불을 지원하는 경우 요청됩니다.
- 구매 주문 번호: 고객이 상점에서 주문할 때 제공한 구매 주문 번호. 구매 주문 번호는 상점과 고객 간 장기 구매 계약 내의 조건에 규정된 대로 고객이 상점에서 주문할 권한을 부여받은 사람임을 승인합니다.

주문 항목

주문 항목은 주문을 구성하는 개별적인 상품 또는 항목입니다. 주문은 최소한 하나의 주문 항목을 가져야 합니다. 각 주문 항목은 고객이 구매를 목적으로 선택한 것입니다. 또한, 각 주문 항목은 거래 계약(보통 장기 구매 계약), 운송 방법, 서비스 센터 및 가격 판매 설정에 대한 참조를 갖습니다. 할인, 운송 경비 및 총 세금은 각 주문 항목과 함께 저장됩니다.

다음 도표는 WebSphere Commerce 주문 항목 자원을 보여줍니다. 각 자원의 설명은 도표 뒤에 나옵니다.



이 도표와 상점 데이터 절의 모든 도표들은 WebSphere Commerce 서버 정보 모델의 부분입니다. 정보 모델에 대한 자세한 정보는 29 페이지의 『상점 데이터 정보 모델』을 참조하십시오. 이 도표에서 사용된 규칙에 대한 자세한 정보는 359 페이지의 부록 A 『UML 범례』를 참조하십시오.

서브오더

주문 항목은 그룹으로 묶여 서브오더를 형성합니다. 서브오더는 특정 주소로 운송될 주문의 부분입니다. 예를 들어, 고객은 장바구니에 있는 상품마다 서로 다른 운송 주소를 지정할 수 있습니다. 각 운송 주소 및 해당 주소와 연관된 상품은 운송 주소별 주문을 구성합니다. 서브오더의 주문 항목은 동일한 운송 주소를 가지며, 주문 항목 총계의 하위 총계를 표시하는 데 사용될 수 있습니다.

OrderItem 오브젝트의 수량 속성은 CatalogEntryShippingInformation 오브젝트의 기본 수량 속성에 의해 증가하여 OrderItem이 나타내는 실제 수량에 도달할 수 있는 매우 큰 숫자입니다. CatalogEntryShippingInformation 오브젝트는 수량이 표시된 측정 단위를 지정합니다.

주문이 대체로 단일 상점과 연관되지만, 상점이나 상점 그룹과 연관될 수 있는 특정 유형의 주문은 주문 프로파일입니다. 주문 프로파일은 오브젝트 모델에서 'Q' 상태의 주문으로 나타납니다. 주문 프로파일은 지불 정보, 운송 주소, 운송 방법 및 지불 청구 주소와 같이 고객에 대한 기본 정보를 보유합니다.

기타 주문 항목 자원

주문 항목은 다음 각 오브젝트 중 0개 또는 한 개와 연관될 수 있습니다.

- 주문 항목을 포함하여 주문을 제출한 고객의 운송 주소. 고객은 상점의 서비스 센터가 이 주소를 사용하여 주문 항목을 적절히 운송할 수 있도록 주문 처리 기간 동안 운송 주소를 지정해야 합니다.
- 고객 주문에 의해 요청된 주문 항목을 운송하고 수령하는, 주문 항목을 재고로 저장하는 서비스 센터.
- 운송 회사(서비스 센터에서 고객에게로 운송 서비스를 제공하는 회사) 담당인 주문 항목에 대한 운송 방법과 해당 운송 회사가 제공하는 운송 서비스. 예를 들어, ABC 운송 회사, 운송 회사, 야간 서비스 및 ABC 운송 회사, 특급 배송은 운송 방법입니다.
- 주문 항목과 연관된 가격 판매 설정. 다른 가격 목록(또는 "거래 포지션 컨테이너")에 다른 판매 설정을 포함시켜 상점은 동일한 상품이나 SKU에 대한 다른 가격을 다른 고객들에게 제시할 수 있습니다. 예를 들어, 여행사는 비행기 표를 성인 가격, 노인 가격, 어린이 가격 및 학생 가격 등 서로 다른 네 가지 가격 목록으로 제안할 수 있습니다.
- 주문 항목에 대한 카탈로그 항목. 각 주문 항목은 카탈로그에서 하나의 항목을 주문합니다.
- 항목을 주문한 규정을 정의하는 거래 계약. 이것은 일반적으로 장기 구매 계약이지만, 주문이 처리되기 위해 제출되기까지는 협상을 나타내는 RFQ(Request for Quotation)일 수 있습니다.



WebSphere Commerce 서버의 주문 자원의 구조에 대한 자세한 정보는
WebSphere Commerce 온라인 도움말의 주문 오브젝트 및 데이터 모델을 참조하십시오.

WebSphere Commerce에서 주문 자원 작성

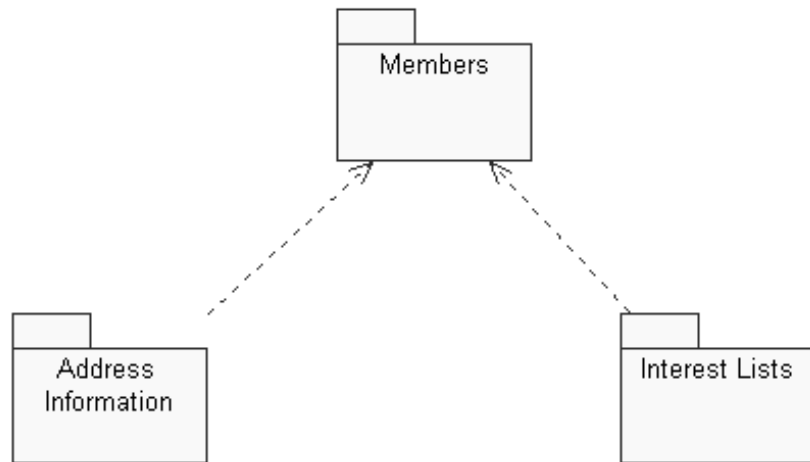
고객은 상점에서 주문을 하거나 WebSphere Commerce 액셀러레이터를 사용하여 상점의 고객 서비스 영업대표가 이러한 일을 완료하도록 도움을 요청할 수 있습니다. B2C 고객을 대표하여 주문을 작성하려면, WebSphere Commerce 온라인 도움말의 주제 "등록된 고객을 위한 주문 작성" 및 "미등록 고객을 위한 주문 작성"을 참조하십시오. B2B 고객을 대표하여 주문을 작성하려면, 도움말의 주제 "회사 사용자를 위한 주문 작성"을 참조하십시오.

제 23 장 고객 및 판매자 자원

상점은 상점 활동에 참여하는 여러 역할자를 포함하지만, 최소한 고객과 판매자를 가집니다.

WebSphere Commerce의 고객 자원 이해

고객은 상점에서 구매하고, 원하는 경우 관심 목록을 작성, 주문 및 상점이나 판매자로부터 구매하는 사람입니다. 다음 도표는 고객이 상점에서 주문하는 데 필요한 자원을 보여줍니다.



이 도표와 상점 데이터 절의 모든 도표들은 WebSphere Commerce 서버 정보 모델의 부분입니다. 정보 모델에 대한 자세한 정보는 29 페이지의 『상점 데이터 정보 모델』을 참조하십시오. 이 도표에서 사용된 규칙에 대한 자세한 정보는 359 페이지의 부록 A 『UML 범례』를 참조하십시오.

앞의 도표에서 처럼 WebSphere Commerce 시스템은 구성원을 포함합니다. 각 사용자 구성원 및 조직 엔티티 구성원은 역할을 지정받을 수 있습니다.

주: WebSphere Commerce에서, 구성원은 조직 엔티티, 사용자 또는 구성원 그룹이 될 수 있습니다. 자세한 정보는 213 페이지의 『구성원』을 참조하십시오.

이 경우, 사용자 구성원은 고객입니다. 고객은 주소 정보를 제공해야 하며, 관심 항목 리스트를 작성할 수 있습니다. 도표는 구성원(고객)과 고객 자원 간의 관계가 서로 연

관련 두 가지 방법을 보여줍니다. 고객은 주소를 소유하고 제공해야 하며, 상점에서 구매할 관심 목록을 작성할 수 있습니다. 주소 및 관심 목록은 고객이 있느냐에 따라 다르며, 고객이 없으면 효용성이 없습니다.

주소 정보

고객은 상점에서 구매할 때 연락 주소, 지불 청구 주소 및 운송 주소 등 세 가지 유형의 주소 정보를 제공해야 합니다. 다음은 이러한 주소 유형을 설명합니다. 각 주소는 고유한 거나 동일할 수 있습니다.

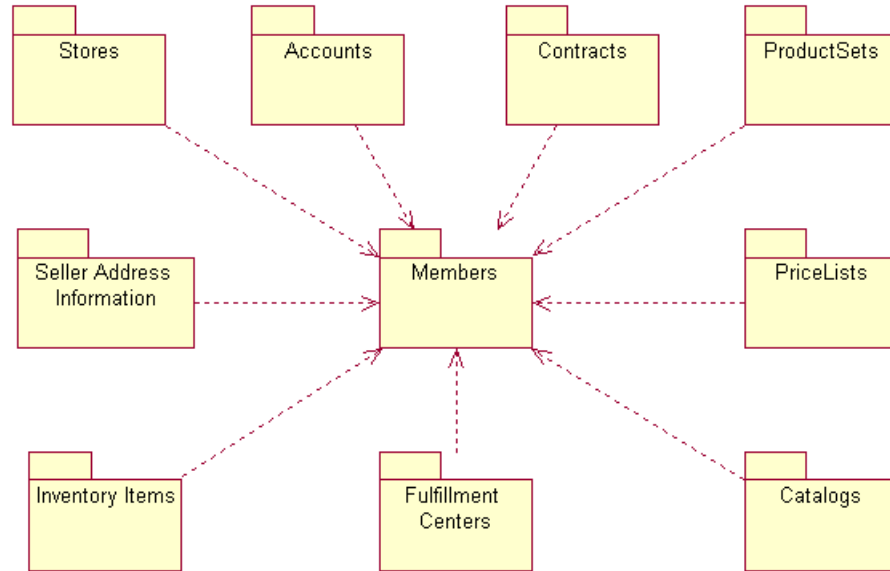
- 연락 주소는 주문의 상태 또는 변경, 판촉 또는 상점 유지보수와 같은 향후의 상점 이벤트 알림 등 다양한 목적으로 고객에게 알리기 위해 사용됩니다. 고객의 연락 주소에는 거리 이름 및 번호, 구/군/시, 시/도, 우편번호, 국가/지역, 전자 우편 주소, 전화번호 및 팩스 번호가 포함됩니다. 일반적으로, 연락 주소는 회사 주소와 같이 고객과 가장 쉽게 통화할 수 있는 것이어야 합니다.
- 지불 청구 주소는 구매시 계산서 또는 인보이스를 보낼 때 사용됩니다. 지불 청구 주소에는 거리 이름 및 번호, 구/군/시, 시/도, 우편번호, 국가/지역, 전화번호, 전자 우편 주소가 포함됩니다. 지불 청구 주소는 연락 주소 또는 운송 주소와 같을 수도 있고 다를 수도 있습니다.
- 운송 주소는 구매한 상품의 운송에 사용됩니다. 운송 주소에는 거리 이름 및 번호, 구/군/시, 시/도, 우편번호, 국가/지역, 전화번호, 전자 우편 주소가 포함됩니다. 운송 주소는 연락 주소 또는 지불 청구 주소와 같을 수도 있고 다를 수도 있습니다.

관심 목록

상점은 관심 목록을 지원할 수 있습니다. 즉, 고객은 미래에 주문하고자 하는 상품을 관심 목록에 추가합니다. 관심 목록은 장바구니가 아니며 관심 목록은 여러 상점의 항목을 포함할 수 있으나 할인, 운송 경비, 세금과 같이 계산된 총계, 가격, 운송 주소, 운송 방법, 재고 사용 가능성 등은 포함하지 않습니다.

WebSphere Commerce의 판매자 자원 이해

판매자는 상점 판매를 추적할 뿐 아니라 전반적인 상점 목표와 관리를 감독합니다. 판매자는 고객에게 상품과 서비스를 판매합니다. 판매자의 역할은 판매자와 동등하며, 모든 WebSphere Commerce 액셀러레이터 기능에 액세스하는 것입니다. 다음 도표는 판매자가 상점을 유지보수하고 고객에게 판매하는 데 필요한 자원을 보여줍니다.



이 도표와 상점 데이터 절의 모든 도표들은 WebSphere Commerce 서버 정보 모델의 부분입니다. 정보 모델에 대한 자세한 정보는 29 페이지의 『상점 데이터 정보 모델』을 참조하십시오. 이 도표에서 사용된 규칙에 대한 자세한 정보는 359 페이지의 부록 A 『UML 범례』를 참조하십시오.

앞의 도표에서 처럼 WebSphere Commerce 시스템은 구성원을 포함합니다. 개별 구성원에게는 상점에 대한 고객 서비스 영업대표나 창고에서의 재고 수령인과 같은 역할이 지정됩니다. 판매자 역할은 고객에게 판매하기 위해 다음 자원들을 유지보수할 수 있습니다.

- 상점
- 계정(선택)
- 장기 구매 계약(또는 최소한 WebSphere Commerce 기본 장기 구매 계약)
- 상품 세트
- 가격 목록
- 카탈로그
- 서비스 센터
- 재고 항목

위의 도표는 구성원(판매자)과 판매자 자원 간의 관계를 보여줍니다. 즉 판매자는 상점을 유지보수하기 위해 위에서 나열된 자원을 가질 수 있으며, 자원은 판매자가 자원을 배치하도록 해야 합니다.

상점

WebSphere Commerce 온라인 상점은 상점 아카이브에 포함되어 있는 세금, 운송, 지불, 카탈로그 및 기타 데이터베이스 자원 뿐 아니라 HTML 및 JavaServer Pages 파일들로 구성되어 있습니다. 상점에는 상점이 기능하도록 하기 위해 WebSphere Commerce 데이터베이스에 대량 자료 반입된 정보인 상점 데이터가 들어 있습니다.

WebSphere Commerce 상점에 대한 자세한 정보는 45 페이지의 제 6 장 『상점 자원』 및 39 페이지의 제 4 부 『상점 데이터 개발』을 참조하십시오.

계정

상점은 비즈니스 계정을 설정하여 고객이 상점에서 구매할 수 있게 합니다. 계정에는 다음과 같은 정보가 포함됩니다.

- 종종 고객이 연관되어 있는 조직의 이름인 계정 이름 이 조직은 상점과의 장기 구매 계약을 정의하며, 고객이 상점에서 구매하는 조건을 규정합니다. 예를 들어, IBM과 같은 조직은 ABC 사무 용품 회사와 장기 구매 계약을 맺습니다.
- 계정에 대한 책임을 지는 판매자 조직 내의 영업대표 조직의 이름인 영업대표 이름.
- 계정에 속한 장기 구매 계약 수.

WebSphere Commerce 계정에 대한 자세한 정보는 108 페이지의 『계정(비즈니스 계정)』 및 WebSphere Commerce 온라인 도움말을 참조하십시오.

장기 구매 계약

일반적으로 WebSphere Commerce에서, 모든 고객은 장기 구매 계약에 따라 구매해야 합니다. 고객과 판매자 간의 각 계정은 하나 이상의 장기 구매 계약으로 연관되어야 합니다(또는 최소한 미등록 고객 또는 고객이 상점에서 구매하거나, 고객이 다른 장기 구매 계약에 얽매이지 않고 상품을 구매하게 할 수 있는 기본 장기 구매 계약). 장기 구매 계약을 체결하면 고객은 장기 구매 계약에 명시된 규정과 비즈니스 정책에 따라 지정된 기간 동안 지정된 가격으로 상점에서 상품을 구매할 수 있습니다. 판매자는 고객이 상점에서 상품을 구매하도록 장기 구매 계약을 배치합니다.

WebSphere Commerce 장기 구매 계약 및 판매자가 사용할 수 있는 기본 장기 구매 계약에 대한 자세한 정보는 109 페이지의 『장기 구매 계약』을 참조하십시오.

상품 세트

상품 세트는 판매자가 다양한 고객에게 서로 다른 카탈로그 보기를 활용하게 하기 위해 판매자가 온라인 카탈로그를 논리적 서브세트로 범주화하는 메커니즘을 제공합니다.

또한 판매자는 고객을 위해 장기 구매 계약을 작성할 수 있으며, 고객이 미리 정의된 상품 세트에 따라 상품을 구매할 수 있음을 명시할 수 있습니다.

WebSphere Commerce 상품 세트에 대한 자세한 정보는 62 페이지의 『상품 세트』를 참조하십시오.

가격 목록

가격 목록은 판매자가 고객에게 제안하거나 제시하는 가격과 연관됩니다. 판매자는 동일한 상품에 대한 다른 가격을 다른 고객에게 제시할 수 있습니다. WebSphere Commerce에서 가격 판매 설정은 *거래 포지션*으로 알려져 있으며, 고객이 해당 가격 지불에 만족하도록 하는 카탈로그 항목 및 기준 가격을 제시합니다.

WebSphere Commerce에서, 판매 설정 오브젝트는 구성원이 소유하고 있는 TradingPositionContainer의 파트입니다. TradingPositionContainer에는 TradingPosition이 포함되어 있으며, 모든 고객이나 일정한 그룹의 고객만이 거래 계약 또는 장기 구매 계약을 통해 사용할 수 있습니다. 때로는 TradingPositionContainer가 가격 목록으로 참조되기도 합니다. 다음과 같이 두 가지 종류의 가격 목록이 있습니다. 하나는 상점 카탈로그의 상품에 대한 기본 가격을 포함한 표준 가격 목록이고, 다른 하나는 상품 목록과 사용자 정의 가격을 지정하는 사용자 정의 가격 목록입니다.

WebSphere Commerce 가격 목록에 대한 자세한 정보는 99 페이지의 제 9 장 『가격 책정 자원』을 참조하십시오.

카탈로그

WebSphere Commerce 상점은 판매자가 판매용으로 제공하는 상품 및 서비스를 보여 주기 위해 최소 하나의 온라인 카탈로그를 사용합니다. 일반적으로 온라인 카탈로그에는 가격, 이미지 및 판매용 항목에 대한 설명이 들어 있습니다. 또한 온라인 카탈로그는 탐색을 가능하게 하기 위해 명확한 카테고리 상품 제공을 제공합니다.

WebSphere Commerce 시스템의 각 상점은 카탈로그 관리를 위해 사용되는 *마스터 카탈로그*를 갖고 있어야 합니다. 마스터 카탈로그는 판매자의 판매 계획을 관리하는 중심 위치입니다. 이것은 모든 상품, 항목, 관계 및 상점에서 판매하는 모든 상품에 대한 표준 가격을 포함하고 있는 단일 카탈로그입니다. 판매자가 하나 이상의 상점을 가지고 있는 경우, 이러한 상점 간에 마스터 카탈로그를 공유할 수 있습니다.

WebSphere Commerce 상품 세트에 대한 자세한 정보는 59 페이지의 제 8 장 『카탈로그 자원』을 참조하십시오.

서비스 센터

서비스 센터는 상점에서 재고 창고 및 운송과 수령 센터의 두 가지로 사용됩니다. 판매자는 하나 이상의 서비스 센터를 가질 수 있습니다.

WebSphere Commerce 서버 관점에서 FulfillmentCenter 오브젝트는 상점 오브젝트와 분리되어 있습니다. 이 오브젝트는 상품 재고 및 운송을 관리합니다. 주문을 운송하기 위해 서비스 센터는 고객이 지정한 ShippingMode 오브젝트에 의존합니다. ShippingMode 오브젝트는 운송 회사와 서비스 주문을 위한 운송 방법을 나타냅니다. 서비스 센터에서 ShippingArrangement 오브젝트는 상점 오브젝트가 일정한 ShippingMode를 사용하여 상품을 운송하기 위해 FulfillmentCenter와 함께 배열되어 있음을 나타냅니다.

WebSphere Commerce 서비스 센터에 대한 자세한 정보는 121 페이지의 제 11 장 『이행 자원』 및 151 페이지의 제 18 장 『운송 자원』을 참조하십시오.

재고 항목

재고 항목에는 판매자의 서비스 센터에서 물리적으로 계산할 수 있는 모든 것이 포함됩니다. WebSphere Commerce 시스템은 항목, 상품, SKU, 번들 및 패키지와 같이 서비스 대상이 되는 특정 유형의 재고를 정의합니다. 그러나 이들은 모두 재고로 간주됩니다. WebSphere Commerce 액셀러레이터의 상품 관리 도구를 사용하는 서비스에 대한 상품이 구성됩니다.

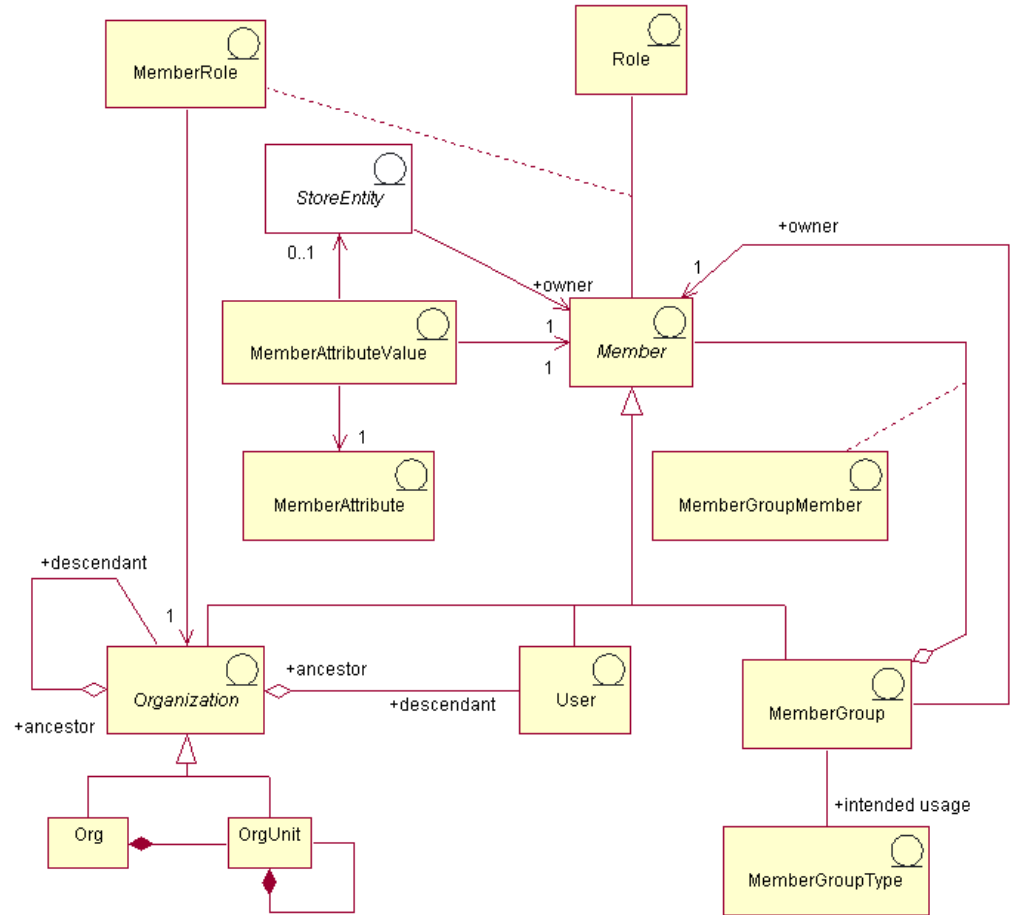
WebSphere Commerce 재고 항목에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말 및 197 페이지의 제 21 장 『재고 자원』을 참조하십시오.

WebSphere Commerce의 구성원 자원 이해

WebSphere Commerce 구성원 자원에는 WebSphere Commerce 시스템 참여자에 대한 데이터가 포함됩니다. 구성원은 사용자, 사용자 그룹 또는 조직 엔티티가 될 수 있습니다. 사이트 운영자와 같은 운영자가 사용자 및 조직 엔티티 구성원에게 역할을 지정합니다. 일단 구성원에게 역할이 지정되면, 액세스 제어 구성요소는 구성원이 활동에 참여할 수 있는 권한을 부여합니다. 예를 들어, 조직은 구매자나 판매자 또는 둘다가 될 수 있습니다. 한 명의 사용자가 여러 역할을 가질 수도 있습니다. 운영자는 또 여러 비즈니스 이유로 카테고리화된 사용자 그룹인 구성원 그룹을 작성할 수 있습니다. WebSphere Commerce 관리 콘솔을 사용하여 조직, 사용자, 역할 및 구성원 그룹을 작성해서 작업해야 합니다.

구성원 자원의 비즈니스 로직은 구성원 등록 및 프로파일 관리 서비스를 제공합니다. 구성원 자원과 밀접하게 연관되어 있는 다른 서비스로는 액세스 제어, 인증 및 세션 관리가 있습니다. 이러한 주제에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오.

다음 도표는 WebSphere Commerce 구성원 자원을 보여줍니다. 각 주문에 대한 설명은 도표 뒤에 나옵니다.



이 도표와 상점 데이터 절의 모든 도표들은 WebSphere Commerce 서버 정보 모델의 부분입니다. 정보 모델에 대한 자세한 정보는 29 페이지의 『상점 데이터 정보 모델』을 참조하십시오. 이 도표에서 사용된 규칙에 대한 자세한 정보는 359 페이지의 부록 A 『UML 범례』를 참조하십시오.

구성원

WebSphere Commerce의 구성원은 다음 중 어느 것도 될 수 있습니다.

- 조직 엔티티. "IBM" 또는 거대한 조직 내의 조직 단위(예: IBM 내의 "전자 상거래 부서")와 같은 조직일 수 있습니다.
- 사용자(등록 또는 미등록). 등록된 사용자는 고유한 식별자 및 암호를 가지며, 등록을 목적으로 한 프로필 데이터를 제공할 필요가 있습니다. 등록된 사용자는 프로필 유형에 따라 분류될 수 있습니다. 'B' 유형은 비즈니스 사용자(또는 B2B 고객)를, 'C' 유형은 소매 사용자(또는 B2C 고객)를 나타냅니다. 등록 및 미등록 사용자에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말의 "구성원"을 참조하십시오.

- **구성원 그룹.** 여러 비즈니스 이유로 인해 카테고리화된 사용자 그룹입니다. 이러한 그룹은 할인, 가격 계산 및 상품 표시와 같은 마케팅 목적 뿐 아니라 액세스 제어, 승인 목적으로도 사용할 수 있습니다.

각 **상점 엔티티**(즉, 상점 또는 상점 그룹)는 구성원이 소유하고 있습니다.

구성원 속성

WebSphere Commerce 구성원은 속성 세트를 가지며, 각 속성은 그와 연관된 값을 갖습니다. 구성원의 기본 사용자 프로파일은 특성, 인적사항, 주소 정보, 구매 히스토리 및 기타 속성을 포함합니다.

비즈니스 사용자 프로파일에는 사원 번호, 직위 또는 직무 설명과 같은 고용 정보 뿐 아니라 기본 사용자 프로파일과 동일한 정보가 포함됩니다. 등록하는 동안, 비즈니스 사용자는 자신이 소속된 비즈니스 조직을 식별해야 합니다. 조직 엔티티의 프로파일에는 조직 이름 및 비즈니스 카테고리과 같은 추가 정보가 포함됩니다.

액세스 제어 규칙은 프로파일 관리 수행을 위한 사용자 권한을 강제합니다. 구성원 프로파일에는 역할, 지불 정보, 주소, 선호 언어 및 통화, 혼한 컴퓨터 장치와 같은 다양한 개인 및 비즈니스 관련 속성이 포함될 수 있습니다. 속성은 상점에 따라 다를 수 있습니다. 이 속성은 구성원 그룹의 지원을 받지 않습니다.

역할

각 사용자 구성원은 조직에서 하나 이상의 역할을 수행할 수 있습니다. 사이트 관리자는 사용자 구성원에게 역할을 지정합니다. 예를 들어, IBM 조직의 구성원 John Smith의 역할은 고객 서비스 영업 대표입니다. John이 IBM 고객을 대신하여 태스크를 수행하고 등록 정보, 주문 또는 반품에 관련된 고객의 조회 및 관심에 대한 지원을 의미합니다. John은 고객 서비스 대표의 역할도 하는 데, 이 역할은 다른 고객 서비스 영업대표에 대한 승인 및 감독 권한 뿐 아니라 위에서 설명한 태스크의 모든 책임을 지는 것입니다.

WebSphere Commerce 시스템은 다음 기본 역할 유형 세트를 제공합니다.

- 사이트 및 콘텐츠 개발 역할
- 기술적 운영 역할
- 마케팅 관리 역할
- 상품 관리 역할
- 비즈니스 관계 관리 역할
- 물류 및 운영 관리 역할
- 조직 관리 역할

이러한 역할에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말 역할을 참조하십시오. 사이트 운영자는 조직 엔티티나 사이트 운영자가 작성한 모든 새로운 역할 뿐 아니라 이러한 역할을 지정할 수 있습니다. 즉, 조직 엔티티에 속한 사용자는 해당 조직 엔티티에 지정된 역할을 맡을 수 있습니다.

사용자 역할을 지정하면 역할 범위가 조직 엔티티가 됩니다. 사용자에게 역할을 지정했다고 해서 사용자가 해당 조직 엔티티 역할을 반드시 수행해야 되는 것은 아닙니다. 즉, 운영자가 지정 역할을 수행하면 운영자는 사용자가 수행하는 역할의 조직 엔티티를 선택할 수 있습니다. 운영자가 루트 조직을 선택하면 사용자는 모든 조직 엔티티 역할을 수행합니다.



WebSphere Commerce의 구성원 자원 구조에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말의 구성원 오브젝트 및 데이터 모델을 참조하십시오.

WebSphere Commerce에서 구성원 자원 작성

상점 소유자로서 행동하는 조직인 판매자를 작성하고 판매자에 대한 정보를 유지보수하려면, 관리 콘솔을 사용해야 합니다. 자세한 정보는 WebSphere Commerce 온라인 도움말 주제 "조직 작성"을 참조하십시오.

고객은 상점 개발자에 의해 작성되지 않습니다. 고객이 상점에서 등록하면 특성이 WebSphere Commerce에 의해 수집되고 유지보수됩니다.

제 5 부 상점에 액세스 제어 추가

제 24 장 상점의 액세스 제어

WebSphere를 사용하면, 사용자는 액세스 제어를 통해 고객이나 운영자의 입장이 되어 어떤 태스크를 수행할 수 있는지 결정할 수 있습니다. 이 장에서는 상점에 액세스 제어를 추가하는 방법과 그에 따라 고객이 볼 수 있는 페이지와 수행할 수 있는 태스크를 제한하는 방법을 중점적으로 다룹니다.

WebSphere의 액세스 제어 모델 및 사이트 레벨에서의 액세스 제어 적용에 관한 자세한 정보는 *IBM WebSphere Commerce 액세스 제어 안내서*를 참조하십시오. 사용자 정의된 코드에서 액세스 제어 구현에 관한 자세한 정보는 *IBM WebSphere Commerce 프로그래머 안내서*를 참조하십시오. 이 안내서는 다음 URL에서 볼 수 있습니다.

Business http://www.ibm.com/software/webservers/commerce/wc_be/lit-tech-general.html

Professional http://www.ibm.com/software/webservers/commerce/wc_pe/lit-tech-general.html

WebSphere Commerce의 액세스 제어에 대한 이해

WebSphere Commerce의 액세스 제어 모델은 *IBM WebSphere Commerce 액세스 제어 안내서*에 자세히 설명되어 있습니다. 여기서는 액세스 제어가 상점 개발에 끼치는 영향을 이해하기 위한 간단한 정보 요약이 제공됩니다.

WebSphere Commerce에서의 액세스 제어는 사용자, 조치, 자원 및 관계로 구성되어 있습니다.

- 사용자는 시스템을 사용하는 사람입니다. 액세스 제어를 위해 사용자는 관련있는 액세스 그룹으로 그룹화되어야 합니다. 예를 들면, 등록된 고객, 게스트 고객 또는 고객 서비스 영업대표와 같은 관리 그룹 등과 같은 액세스 그룹으로 사용자들을 그룹화할 수 있습니다.
- 조치는 사용자가 자원에서 수행할 수 있는 활동입니다. 액세스 제어를 위해 조치도 관련있는 조치 그룹으로 그룹화되어야 합니다. 예를 들면, 상점에서 사용되는 공통 조치는 보기입니다. 보기는 고객에서 상점 페이지를 표시해주기 위해 호출됩니다. 상점에서 사용되는 보기는 조치 그룹에 지정되어야 합니다.
- 자원은 보호된 엔티티입니다. 예를 들면, 조치가 보기일 때 조치에 대한 자원은 클래스 `com.ibm.commerce.command.ViewCommand`입니다. 액세스 제어를 위해 자원은 자원 그룹으로 그룹화됩니다.
- 관계는 사용자와 자원 사이의 관계입니다. 액세스 제어 정책은 사용자와 자원 사이의 관계가 충족되는 것을 필요로 할 수 있습니다.

액세스 제어 정책

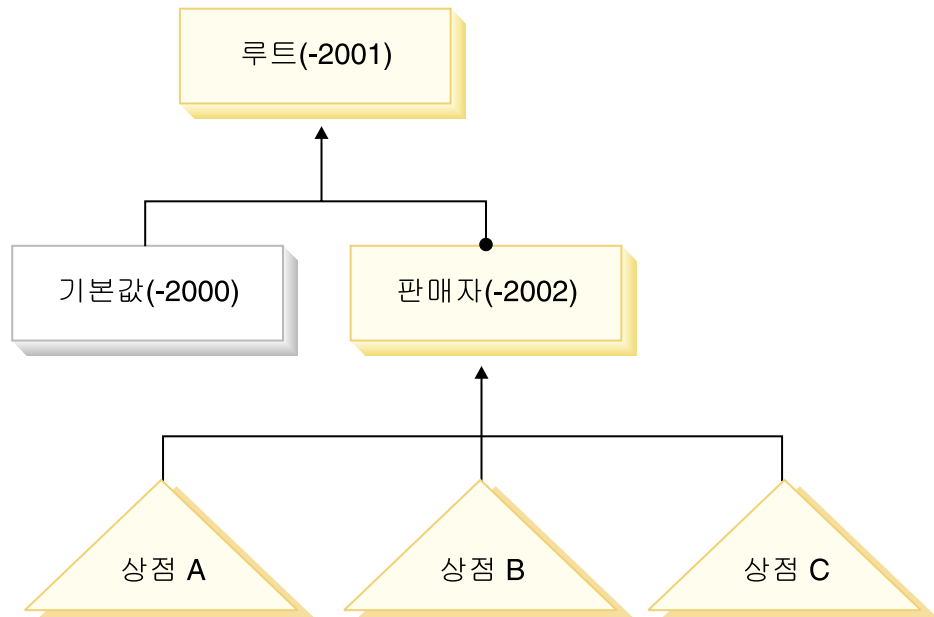
액세스 제어 정책은 액세스 그룹에 있는 사용자가 자원에 대한 특정 관계를 만족시키는 한, 액세스 그룹이 WebSphere의 자원에서 특정 조치를 수행할 수 있는 권한을 부여합니다.

WebSphere Commerce는 인스턴스 작성 중에 로드되는 200개가 넘는 기본 액세스 제어 정책을 제공합니다. 이 정책들은 주문 작성 및 처리, **Business** 견적 요청 및 **Business** 장기 구매 계약과 같은 거래를 포함하는 폭넓은 공통 비즈니스 활동에 적용될 수 있습니다. 기본 정책들은 *IBM WebSphere Commerce 액세스 제어 안내서*에 문서화되어 있습니다.

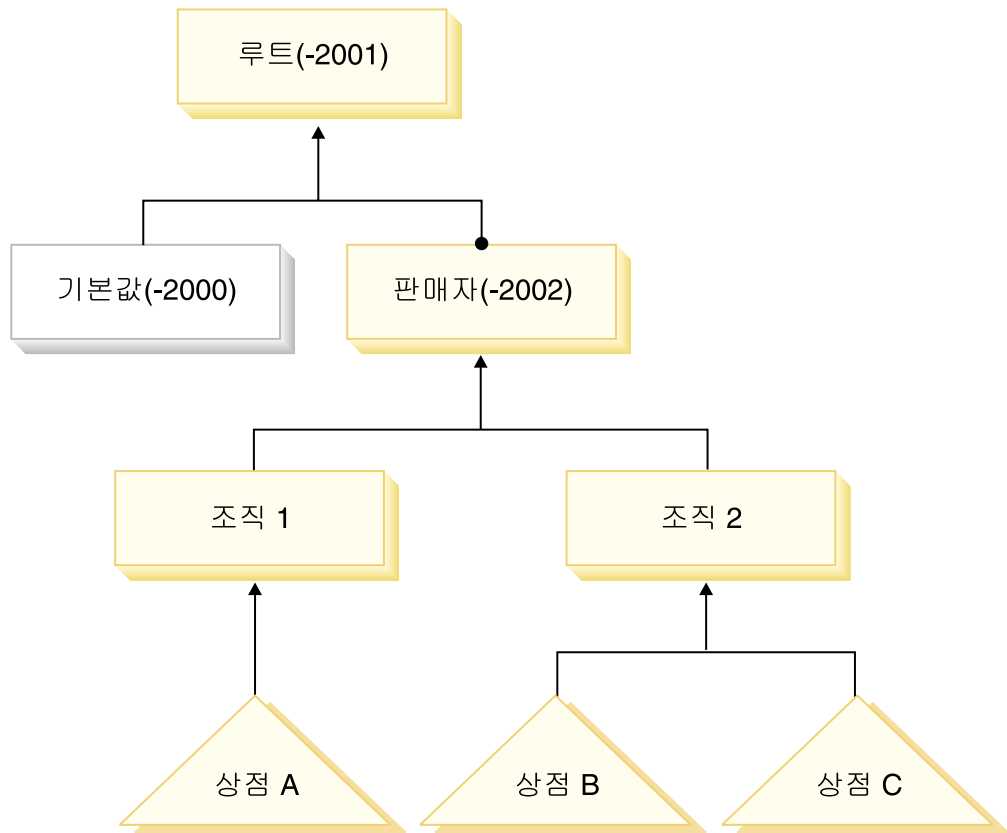
각 액세스 제어 정책은 조직 엔티티가 소유합니다. 액세스 제어 정책은 액세스 제어 정책 소유자가 소유한 자원에만 적용됩니다.

주: 액세스 제어에서 자원의 소유권은 특별한 의미를 가집니다. 모든 자원은 `com.ibm.commerce.security.Protectable` 인터페이스를 구현해야 합니다. 이 인터페이스의 메소드 중 하나는 `getOwner()`이며, 이는 자원 소유자의 구성원 ID를 리턴합니다. 예를 들면, 주문 엔티티 bean은 원격 인터페이스가 보호 가능한 인터페이스를 확장하므로써 보호되는 자원입니다. `getOwner()`의 주문 구현은 특정 주문 자원이 주문이 제출된 상점의 소유자를 리턴하는 것입니다. 예를 들어, `com.ibm.commerce.command.ViewCommand`와 같이 자원이 명령인 정책에 대하여 `getOwner()`의 기본 구현은 현재 명령 컨텍스트에 있는 상점의 소유자를 리턴하는 것입니다. 명령 컨텍스트에 상점이 없다면 루트 조직이 소유자로 사용됩니다. 자세한 정보는 *IBM WebSphere Commerce 프로그래머 안내서*를 참조하십시오.

다음 도표를 고려하십시오.



WebSphere Commerce Professional Edition에서 루트 조직은 최상위 레벨입니다. 이는 모든 다른 조직 엔티티를 포함합니다. 결과적으로 루트 조직에 의해 소유된 액세스 제어 정책은 사이트의 모든 자원에 적용됩니다. 액세스 제어 정책이 판매자 조직에 의해 소유된다면, 액세스 제어 정책은 판매자에 의해 소유된 자원에만 적용됩니다.



WebSphere Commerce Business Edition에서 루트 조직은 최상위 레벨입니다. 이는 모든 다른 조직 엔티티를 포함합니다. 결과적으로, 루트 조직에 의해 소유된 액세스 제어 정책은 사이트의 모든 자원에 적용됩니다. 액세스 제어 정책이 판매자 조직에 의해 소유된다면, 액세스 제어 정책은 판매자에 의해 소유된 자원에만 적용됩니다. 다시 말하면, 조직 1과 2에 의해 소유된 자원(상점 A, B, C 포함)에만 적용됩니다. 액세스 제어 정책이 조직 1에 의해 소유되어 있다면, 이는 상점 A를 포함하는 자원에만 적용됩니다. 액세스 제어 정책이 조직 2에 의해 소유된다면, 상점 B와 C에 있는 자원에 적용됩니다.

Business 액세스 제어 정책이 조직 엔티티에 의해 소유되어 있으므로, 사이트에 복수의 상점을 작성하고 각 상점에 서로 다른 액세스 제어 정책을 적용하려 한다면 각 상점을 소유하는 분리된 조직을 작성해야 합니다.

주: WebSphere Commerce에서 제공되는 거의 모든 기본 액세스 제어 정책은 루트 조직에 의해 소유되어 있고 사이트의 모든 자원에 적용됩니다. 루트 조직에 의해 소유된 액세스 제어 정책은 사이트 레벨 정책이라 불립니다. 다른 조직 엔티티에 의해 소유된 정책은 조직 레벨 정책이라 합니다.

상점의 액세스 제어


WebSphere Commerce에서 작성된 모든 상점은 루트 조직에 의해 소유된 기본 액세스 제어 정책에 종속됩니다. 상점의 자원은 상점을 소유하고 있는 조직에 의해 소유된 모든 액세스 제어 정책이나 조직의 상위 조직에 의해 소유된 모든 액세스 제어 정책에 종속됩니다.


기본적으로, 루트 조직은 거의 모든 액세스 제어 정책을 소유합니다. 하지만 WebSphere Commerce에서 제공된 견본 상점 중 하나를 바탕으로 상점을 작성한다면 상점을 소유하고 있는 조직에 의해 소유된 새로운 액세스 제어 정책을 만들어야 합니다. 견본 상점에서의 액세스 제어 정책에 관한 자세한 정보는 『견본 상점의 액세스 제어』를 참조하십시오.

견본을 바탕으로 만드는 것과 관계없이 사용자가 상점을 작성하고 있을 때, 사용자는 그 조직에 의해 소유된 상점에만 적용되는 액세스 제어 정책을 새로 작성하거나 수정하고자 할 수 있습니다. 예를 들면, 상점 페이지를 표시하기 위한 새로운 보기를 만든다면 이 보기들에 액세스 제어 정책을 지정해야 합니다.

조직 레벨의 액세스 제어 데이터는 상위 레벨 액세스 제어 정책 파일에 정의되어 있습니다. 이 파일은 모든 정책에 의해 사용될 수 있는 가능한 조치, 조치 그룹, 자원, 자원 그룹 및 관계를 정의합니다. 또한 특정 조직에만 해당하는 정책도 정의합니다. WebSphere Commerce에 제공된 견본 상점은 이 상위 레벨 액세스 제어 정책 파일을 포함하고 있습니다. 다음 절에서는 견본 상점이 이 액세스 제어 정책 파일을 사용하여 조직 정보를 정의하는 방법을 설명합니다.

견본 상점의 액세스 제어

모든 견본 상점은 상점에 대하여 특수하게 작성된 액세스 제어 정책을 정의하는 상위 레벨 액세스 제어 정책 파일을 포함합니다. 이 정책은 상점을 소유한 조직에 의해 소유됩니다. NewFashion 및  ToolTech 견본 상점에 대하여 정의된 액세스 제어 정책은 아래에서 논의됩니다.

 ToolTech에 대하여 작성된 액세스 제어 정책은 다음과 같습니다.

- AllUsersForToolTechExecuteToolTechAllUsersViews
- RegisteredApprovedUsersForToolTechExecuteToolTechRegisteredApprovedUsersViews

NewFashion에 대하여 작성된 액세스 제어 정책은 다음과 같습니다.


- AllUsersExecuteNewFashionAllUsersViews


이 액세스 제어 정책은 견본 상점 또는 견본에 바탕을 둔 상점에서 어떤 사용자가 어떤 보기를 보게되는지를 결정합니다.

이 정책에 대한 액세스 제어 정보는 *samplestorenameAccessPolicies.xml* 및 *samplestorenameAccessPolicies_locale.xml* 등과 같은 견본 상점에서 사용된 가능한 조치, 조치 그룹, 자원 및 정책 정의 등을 정의하는 두 개의 상위 레벨 액세스 제어 정책 파일에 정의되어 있습니다. 이들 파일은 다음 디렉토리에 위치합니다.

-  NT drive:\WebSphere\CommerceServer\samples\stores\
samplestorename
-  2000 drive:\ProgramFiles\WebSphere\CommerceServer\samples\
\stores*samplestorename*
-  AIX /usr/WebSphere/CommerceServer/samples/stores/
samplestorename
-  Solaris /opt/WebSphere/CommerceServer/samples/stores/
samplestorename
-  Linux /opt/WebSphere/CommerceServer/samples/stores/
samplestorename
-  400 /QIBM/ProdData/WebCommerce/samples/stores/*samplestorename*
여기서 *samplestorename*은 NewFashion과 같이 사용자가 작성한 상점이 바탕을 둔 견본 상점 아카이브의 이름입니다.

견본 상점을 상점 아카이브로 패키징하기 전에, 이 두 개의 상위 레벨 액세스 제어 정책 파일은 로더 패키지와 함께 사용되기 용이한 두 개의 XML 파일로 변환됩니다. 이 변환된 XML 파일들은 상점 아카이브에 패키징되고 나머지 상점 아카이브와 함께 공개됩니다.

견본 상점 액세스 제어 정책 파일의 이해: 상점 레벨에서 액세스 제어가 추가되는 방법을 이해하려면, 상위 레벨 견본 상점 액세스 제어 정책 파일을 잘 알아야 합니다. 다음 예는  *ToolTechAccessPolicies.xml* 파일입니다.

조치 정의:  *ToolTechAccessPolicies.xml* 파일의 첫 번째 절은 기존 액세스 제어 정책이 미치지 못하는 새로운 조치를 상점에서 정의합니다. 이런 경우, 조치는 상점에서 사용되는 모든 보기입니다. URL에서 직접 호출될 수 있거나 다른 명령으로부터의 방향 재지정(보기로의 전달로 실행되는 것과 정반대의)으로 실행될 수 있는 보기를 사용하여 상점에서 페이지를 표시하기 위해서는, 이 보기를 조치로 정의해야 합니다. 다음 예를 살펴보십시오.

```
<!-- [Start of Action definitions] -->
<!-- [this is the dictionary of possible actions -->
<Action Name="GenericApplicationError"
CommandName="GenericApplicationError">
</Action>

<Action Name="GenericSystemError"
```

```
CommandName="GenericSystemError">
</Action>
```

```
<Action Name="OrderOptionsView"
CommandName="OrderOptionsView">
</Action>
```

```
<!--[End of Action definitions] -->
```

여기서

- Action Name은 XML 파일에서 이 조치를 참조하는 데 사용되는 레이블입니다. 이 예에서 레이블은 보기 이름과 같습니다.
- CommandName은 VIEWREG 테이블의 VIEWNAME 열에 저장되어 있는 보기의 이름입니다. CommandName은 ACACTION 테이블의 조치 열에 저장됩니다.

주: 조치가 WebSphere Commerce의 기본 정책이나 다른 정책에 정의되어 있다면, 조치를 사용하는 각 정책에 대하여 조치를 재지정할 필요가 없습니다.

자원 카테고리 정의: 파일의 두 번째 절에서는 자원 카테고리를 정의합니다. 자원 카테고리는 자원의 클래스를 참조합니다. ToolTech에 대하여 식별된 자원 카테고리는 클래스 com.ibm.commerce.command.ViewCommand 및 com.ibm.commerce.tools.command.ToolsForwardViewCommand입니다.

```
<!-- [Start of Resource Category definitions] -->
<!-- the dictionary of Protectable resources -->
<ResourceCategory Name="com.ibm.commerce.command.ViewCommandResourceCategory"
ResourceBeanClass="com.ibm.commerce.command.ViewCommand">
</ResourceCategory>
<ResourceCategory Name="com.ibm.commerce.tools.command.
ToolsForwardViewCommandResourceCategory"
ResourceBeanClass="com.ibm.commerce.tools.command.ToolsForwardViewCommand">
</ResourceCategory>
<!-- [End of Resource Category definitions] -->
```

여기서,

- ResourceCategory Name 레벨은 XML 파일에서 이 자원 카테고리를 참조하기 위해 사용됩니다.
- ResourceBeanClass는 클래스의 이름입니다.

주: 자원 카테고리가 WebSphere Commerce의 기본 정책이나 다른 정책에 정의되어 있다면, 자원 카테고리를 사용하는 각 정책에 대하여 조치를 재지정할 필요가 없습니다. 위의 예에 정의되어 있는 자원 카테고리는 이미 기본 정책에 정의되어 있지만 설명을 목적으로 여기에서 다시 한번 정의됩니다.

조치 그룹 정의: 세 번째 절에서는 조치 그룹을 정의합니다. 조치 그룹은 파일의 첫 번째 절에 정의된 조치의 그룹화입니다. ToolTech 예에서 모든 사용자 보기는 모든 사용자가 보기에 액세스할 수 있게 해주는 정책에서 사용될 수 있는

ToolTechAllUserViews 그룹 또는 등록된 사용자만 보기에 액세스할 수 있게 해주는 정책에서 사용될 수 있는 ToolTechRegisteredApprovedUserViews 그룹으로 그룹화됩니다.

주: WebSphere Commerce의 다른 부분에 정의되어 있는 조치를 사용자 조치 그룹에 추가할 수도 있습니다. WebSphere Commerce의 다른 곳에서 정의되었다면, 그 조치들은 224 페이지의 『조치 정의』에서 논의되는 조치 목록에 정의될 필요가 없습니다.

```
<!-- [Start of Action Group definitions] -->
<!-- Dictionary of grouped actions usable in policies -->
<!-- cross-component view-related action groups -->
<ActionGroup Name="ToolTechAllUsersViews"
OwnerID="RootOrganization">

<ActionGroupAction Name="UserRegistrationForm"/>
<ActionGroupAction Name="UserRegistrationErrorView"/>
<ActionGroupAction Name="GenericApplicationError"/>
<ActionGroupAction Name="GenericSystemError"/>
<ActionGroupAction Name="LogonForm"/>
</ActionGroup>

<!-- [End of Action Group definitions] -->
```

여기서

- ActionGroup Name은 조치 그룹의 이름입니다.
- OwnerID는 조치 그룹의 소유자입니다. 루트 조직은 조치 그룹의 소유자이어야 합니다.
- ActionGroupAction Name은 이 그룹에 포함된 조치의 이름입니다. ActionGroupAction Name은 224 페이지의 『조치 정의』에서 조치 이름 요소에 정의된 이름과 일치해야 합니다.

자원 그룹 정의: 다음 절에서는 자원 그룹을 정의합니다. 자원 그룹은 연관된 자원 세트를 식별합니다.

```
<!-- [Start of Resource Group definitions] -->
<!-- Dictionary of grouped resources usable in policies -->

<!-- Grouped resources permitting view execution,
by any command extending com.ibm.commerce.command.ViewCommand
(with or without the Tools Framework) -->
<ResourceGroup Name="ViewCommandResourceGroup" OwnerID="RootOrganization">

<ResourceGroupResource Name="com.ibm.commerce.command.ViewCommandResourceCategory"/>
</ResourceGroup>

</ResourceGroup> !-- [End of Resource Group definitions] -->
```

여기서

- ResourceGroup Name은 자원 그룹의 이름입니다.
- OwnerID는 자원 그룹의 소유자입니다. 루트 조직은 자원 그룹을 소유해야 합니다.

- ResourceGroupName은 그룹에 포함된 자원 카테고리의 이름입니다.

정책 정의: 마지막 절에서는 상점에서 사용되는 새 정책을 정의합니다.

```
!-- [Start of Policy definitions] -->
!-- AllUsers for ToolTech can execute ToolTechAllUsersViews -->

<Policy Name="AllUsersForToolTechExecuteToolTechAllUsersViews"
OwnerID="MEMBER_ID"
UserGroup="AllUsers"
UserGroupOwner="RootOrganization"
ActionGroupName="ToolTechAllUsersViews"
ResourceGroupName="ViewCommandResourceGroup">
  </Policy>
!-- RegisteredApprovedUsers for ToolTech can execute
ToolTechRegisteredApprovedUsersViews -->

<Policy Name="RegisteredApprovedUsersFor
ToolTechExecuteToolTechRegisteredApprovedUsersViews"
OwnerID="MEMBER_ID"
UserGroup="RegisteredApprovedUsers"
UserGroupOwner="RootOrganization"
ActionGroupName="ToolTechRegisteredApprovedUsersViews"
ResourceGroupName="ViewCommandResourceGroup">
  </Policy>

!-- [End of of Policy definitions] -->
```

여기서,

- Policy Name은 정의되는 정책의 이름입니다.
- OwnerId는 정책 소유자입니다. 이 경우, 정책의 소유자는 상점을 소유하고 있는 조직입니다.
- UserGroup은 정책이 적용되는 사용자 그룹(액세스 그룹)입니다.
- UserGroupOwner는 액세스 그룹의 소유자입니다. 이 예에서 액세스 그룹의 소유자는 정책 소유자와 다릅니다. 정책 소유자와 UserGroupOwner가 같다면 이 요소는 생략될 수 있습니다.
- ActionGroupName은 정책이 적용되는 조치의 그룹입니다.
- ResourceGroupName은 정책이 적용되는 자원의 그룹입니다.

상점에 액세스 제어 추가

상점 개발 관점에서 필요한 가장 일반적인 액세스 제어 유형은 상점에 대하여 작성하는 새로운 보기 및 명령에 대한 것입니다. 하지만 상점에 다른 액세스 제어 유형을 추가하기를 원할 수 있습니다. 보기, 명령 및 다른 특징에 대한 액세스 제어에 관한 자세한 정보는 *IBM WebSphere Commerce 액세스 제어 안내서*를 참조하십시오. 이 안내서의 다음 단계로 계속 진행하기 전에 *IBM WebSphere Commerce 액세스 제어 안내서*를 검토하였는지 확인하십시오.

견본 상점을 바탕으로 한 상점에 새로운 액세스 제어 특징을 추가하려 한다면, 기존 상위 레벨 액세스 제어 정책 XML 파일을 편집하십시오. 자세한 지시사항은 232 페이지의 『상점 아카이브의 액세스 제어 파일 편집』을 참조하십시오. 견본 상점을 바탕으로 하지 않는 상점에 액세스 제어를 추가하려 한다면, 새로운 상위 레벨 액세스 제어 정책 XML 파일을 작성하고 그것을 변환해야 합니다. 자세한 지시사항은 『상점에 액세스 제어 작성』을 참조하십시오.






상점에 액세스 제어 작성

이 액세스 제어 자원은 액세스 제어를 위해 두 개의 상위 레벨 액세스 제어 XML 파일을 작성하고 그것을 변환하는 상점에 있는 다른 자원과 다릅니다. 변환된 XML 파일은 로더 패키지를 사용하는 데이터베이스에서 로드되거나 상점 아카이브에서 사용됩니다.

액세스 제어 자원 작성을 위해 다음을 수행하십시오.

1. 견본 상점 *samplestorenameAccessPolicies.xml* 및

*samplestorenameAccessPolicies_locale.xml*에 대하여 상점 자원을 작성하는 데 사용되는 상위 레벨 XML 파일을 검토하십시오. 이들 파일은 다음 디렉토리에 위치합니다.

-  NT drive:\WebSphere\CommerceServer\samples\stores*samplestorename*
-  2000 drive:\ProgramFiles\WebSphere\CommerceServer\samples\stores*samplestorename*
-  AIX /usr/WebSphere/CommerceServer/samples/stores/*samplestorename*
-  Solaris /opt/WebSphere/CommerceServer/samples/stores/*samplestorename*
-  Linux /opt/WebSphere/CommerceServer/samples/stores/*samplestorename*
-  400 /QIBM/ProdData/WebCommerce/samples/stores/*samplestorename*

여기서 *samplestorename*은 NewFashion과 같이 사용자가 작성한 상점이 바탕을 둔 견본 상점 아카이브의 이름입니다.

2. 361 페이지의 부록 B 『데이터 작성』 정보를 검토하십시오.

3. *samplestorenameAccessPolicies.xml* 파일 중 하나를 복사하거나 새 파일을 작성해서 *storenameAccessPolicies.xml* 파일을 작성하십시오. 자세한 정보는 *samplestorenameAccessPolicies.xml*에 해당하는 DTD 파일을 참조하십시오. DTD 파일은 다음 디렉토리에 있습니다.

- ▶ NT drive:\WebSphere\CommerceServer\xml\policies\dtd
 - ▶ 2000 drive:\Program Files\WebSphere\CommerceServer\xml\policies\dtd
 - ▶ AIX /usr/WebSphere/CommerceServer/xml/policies/dtd
 - ▶ Solaris /opt/WebSphere/CommerceServer/xml/policies/dtd
 - ▶ Linux /opt/WebSphere/CommerceServer/xml/policies/dtd
 - ▶ 400 /QIBM/ProdData/WebCommerce/xml/policies/dtd
4. 상점에서 지원하는 각 로케일에 대하여 별도의 상위 레벨 XML 파일을 작성하십시오. 로케일 고유의 파일은 모든 설명 정보를 지정하고 이름 정보를 표시해야 하므로 쉽게 번역할 수 있습니다. *samplestorenameAccessPolicies_locale.xml* 파일 중 하나를 복사하거나 새로운 파일을 작성하여 상점에 있는 각 언어에 대하여 *storenameAccessPolicies_locale.xml* 파일을 작성하십시오.
- 주: 상점이 오직 하나의 언어만을 지원해도 로케일 고유의 파일을 작성할 필요가 있습니다.
5. 올바른 액세스 제어 정보를 파일에 추가하십시오. 자세한 정보는 223 페이지의 『견본 상점의 액세스 제어』 및 *IBM WebSphere Commerce 액세스 제어 안내서*를 참조하십시오.
6. *storenameAccessPolicies.xml* 및 *storenameAccessPolicies_locale.xml*을 다음 디렉토리에 복사하십시오.
- ▶ NT drive:\WebSphere\CommerceServer\xml\policies\xml
 - ▶ 2000 drive:\Program Files\WebSphere\CommerceServer\xml\policies\xml
 - ▶ AIX /usr/WebSphere/CommerceServer/xml/policies/xml
 - ▶ Solaris /opt/WebSphere/CommerceServer/xml/policies/xml
 - ▶ Linux /opt/WebSphere/CommerceServer/xml/policies/xml
 - ▶ 400 사용자 데이터 디렉토리로 복사하십시오. 기존 XML 파일에 DTD로의 전체 경로를 지정하십시오. 액세스 제어 DTD 파일은 /QIBM/ProdData/WebCommerce/xml/policies/dtd 디렉토리에 위치합니다.
7. `xmltransform` 명령을 수행하여 *storenameAccessPolicies.xml*을 변환하십시오.
- a. 명령 프롬프트에서 디렉토리를 다음과 같이 변경하십시오.
- ▶ NT drive:\WebSphere\CommerceServer\bin
 - ▶ 2000 drive:\ProgramFiles\WebSphere\CommerceServer\bin

- ▶ AIX /usr/WebSphere/CommerceServer/bin
- ▶ Solaris /opt/WebSphere/CommerceServer/bin
- ▶ Linux /opt/WebSphere/CommerceServer/bin

b. 다음을 입력하십시오.

```
xmltransform -infile
..\xml\policies\xml\samplestorenameAccessPolicies.xml -transform
..\xml\policies\xsl\accesscontrol.xsl -outfile
..\xml\policies\xml\samplestorenameAccessPoliciesOut.xml
▶ 400 TRNWCXML INFILE (input file) TRANSFORM('/QIBM/
ProdData/WebCommerce/xml/policies/xsl
/accesscontrol.xsl') INSTRoot(instance_root) OUTFILE(output_file)
```

c. 다음 로그 파일을 확인하여 변환이 완료되었는지 확인하십시오.

- ▶ NT drive:\WebSphere\CommerceServer\bin\xmltransform.db2.log
- ▶ 2000 drive:\Program Files\WebSphere\CommerceServer\bin\xmltransform.db2.log
- ▶ AIX /usr/WebSphere/CommerceServer/bin/xmltransform.db2.log
- ▶ Solaris /opt/WebSphere/CommerceServer/bin/xmltransform.db2.log
- ▶ Linux /opt/WebSphere/CommerceServer/bin/xmltransform.db2.log
- ▶ 400 /QIBM/UserData/WebCommerce/instances/instancename /logs/TRNWCXML.tx

변환이 성공적이었다면, 다음 메시지가 표시됩니다.

```
"<DATE> <TIME> java.lang.Class main XMLTransformer Transform
Successful"
```

8. xmltransform 명령을 실행하여 storenameAccessPolicies_locale.xml을 변환하십시오.

a. 명령 프롬프트에서 디렉토리를 다음과 같이 변경하십시오.

- ▶ NT drive:\WebSphere\CommerceServer\bin
- ▶ 2000 drive:\ProgramFiles\WebSphere\CommerceServer\bin
- ▶ AIX /usr/WebSphere/CommerceServer\bin
- ▶ Solaris /opt/WebSphere/CommerceServer/bin

- ▶ Linux /opt/WebSphere/CommerceServer/bin

b. 다음을 입력하십시오.

```
xmltransform -infile
..\xml\policies\xml\storenameAccessPolicies_locale.xml -transform
..\xml\policies\xsl\accesscontrolnls.xsl -outfile
..\xml\policies\xml\storenameAccessPoliciesOut_locale.xml
```

c. ▶ 400 TRNWCSXML INFILE(input file) TRANSFORM('/QIBM/ProdData/
WebCommerce/xml/policies/ xsl/accesscontrolnls.xsl')
INSTROOT(instance_root) OUTFILE(output_file)

9. 변환된 XML 파일에 다음 변경을 수행하십시오.

a. *storenameAccessPoliciesOut.xml*에서 열기 및 닫기 태그를 다음과 같이 바꾸십시오.

```
<?xml version="1.0"?>
<!DOCTYPE accesscontrol-asset SYSTEM "accesscontrol.dtd">
<accesscontrol-asset>
</accesscontrol-asset>
```

b. *storenameAccessPoliciesOut_locale.xml*에서 열기 및 닫기 태그를 다음과 같이 바꾸십시오.

```
<?xml version="1.0" encoding="correct language code for the file"?>
<!DOCTYPE accesscontrol-asset SYSTEM "../accesscontrol.dtd">
<accesscontrol-asset>
</accesscontrol-asset>
```

c. *storenameAccessPoliciesOut_locale.xml*에서 @locale을 &locale로 바꾸십시오. 예를 들면, LANGUAGE_ID="@en_US"를 LANGUAGE_ID("&en_US;"로 변경하십시오.

d. *storenameAccessPoliciesOut_locale.xml*에서 "acpoldesc" 테이블 참조를 찾으십시오. ACPOLICY_ID 값 끝에 있는 @를 제거하십시오. 예를 들면, "@AllUsersExecuteInFashionAllUsersViews@"를 "@AllUsersExecuteInFashionAllUsersViews"로 변경하십시오.

e. *storenameAccessPoliciesOut.xml*에서 MEMBER_ID="MEMBER_ID"를 MEMBER_ID("&MEMBER_ID;"로 바꾸십시오.

f. *storenameAccessPoliciesOut.xml*에서 "acpolicy" 테이블 참조를 찾으십시오. ACPOLICY_ID 값 끝에 있는 "@MEMBER_ID"을 제거하십시오. 예를 들면, "@AllUsersExecuteInFashionAllUsersViews@MEMBER_ID"를 "@AllUsersExecuteInFashionAllUsersViews"로 변경하십시오.

10. 여기서 사용자는 두 가지 옵션을 갖습니다.

- 로더 패키지를 사용하여 액세스 제어 데이터를 로드하십시오. 자세한 정보는 247 페이지의 제 7 부 『상점 공개』를 참조하십시오.

- 상점 아카이브에 액세스 제어 데이터를 추가하십시오. 상점 아카이브 작성에 대한 자세한 정보는 237 페이지의 제 6 부 『상점 패키지』를 참조하십시오.



@ 및 & 사용에 대한 자세한 정보는 361 페이지의 부록 B 『데이터 작성』을 참조하십시오.

상점 아카이브의 액세스 제어 파일 편집

WebSphere Commerce 는 두 개의 사전 변환된 액세스 제어 XML 파일을 제공합니다. 하나는 모든 언어에 해당하는 것(*samplestorenameAccessPolicies.xml*)이며, 다른 하나는 각 견본 상점에 대한 로케일 특정 정보를 포함하는 것(*samplestorenameAccessPolicies_locale.xml*)입니다. 사용자는 결과적으로 만들어진 이 두 개의 XML 파일을 하나는 모든 언어에 적용되는 것(*samplestorenameAccessPoliciesOut.xml*)으로, 다른 하나는 로케일 특정 정보를 포함하는 것(*samplestorenameAccessPoliciesOut_locale.xml*)으로 변환해야 합니다.

상점 아카이브의 액세스 제어 데이터베이스 자원을 편집하려면 다음을 수행하십시오.

1. 사용자의 상점이 기반을 둔 견본 상점에 대하여 사전 변환된 액세스 제어 XML 파일들을 찾으십시오. 이 파일들은 *samplestorenameAccessPolicies.xml* 및 *samplestorenameAccessPolicies_locale.xml*입니다. 이 파일들은 기본적으로 다음 디렉토리에 위치합니다.

-  NT drive:\WebSphere\CommerceServer\samples\stores*samplestorename*
-  2000 drive:\ProgramFiles\WebSphere\CommerceServer\samples\stores*samplestorename*
-  AIX /usr/WebSphere/CommerceServer/samples/stores/*samplestorename*
-  Solaris /opt/WebSphere/CommerceServer/samples/stores/*samplestorename*
-  Linux /opt/WebSphere/CommerceServer/samples/stores/*samplestorename*
-  400 /QIBM/ProdData/WebCommerce/samples/stores/*samplestorename*

여기서 *samplestorename*은 NewFashion과 같이 사용자가 작성한 상점이 바탕을 둔 견본 상점 아카이브의 이름입니다.

주: 해당하는 DTD 파일을 변경하면 정책을 사용할 수 없게 될 수 있습니다.

2. 파일에 필요한 변경을 하십시오. 기존 파일에 관한 자세한 정보는 224 페이지의 『견본 상점 액세스 제어 정책 파일의 이해』를 참조하십시오. WebSphere Commerce 에 있는 액세스 제어 방법에 관한 자세한 정보는 *IBM WebSphere Commerce 액세스 제어 안내서*를 참조하십시오.

3. *samplestorenameAccessPolicies.xml* 및 *samplestorenameAccessPolicies_locale.xml*을 다음 디렉토리에 복사하십시오.

- ▶ NT drive:\WebSphere\CommerceServer\xml\policies\xml
- ▶ 2000 drive:\Program Files\WebSphere\CommerceServer\xml\policies\xml
- ▶ AIX /usr/WebSphere/CommerceServer/xml/policies/xml
- ▶ Solaris /opt/WebSphere/CommerceServer/xml/policies/xml
- ▶ Linux /opt/WebSphere/CommerceServer/xml/policies/xml
- ▶ 400 사용자 데이터 디렉토리로 복사하십시오. 기존 XML 파일에 DTD로 의 전체 경로를 지정하십시오. 액세스 제어 DTD 파일은 /QIBM/ProdData/WebCommerce/xml/policies/dtd 디렉토리에 위치합니다.

4. `xmltransform` 명령을 실행하여 *samplestorenameAccessPolicies.xml*을 변환 하십시오.

a. 명령 프롬프트에서 디렉토리를 다음과 같이 변경하십시오.

- ▶ NT drive:\WebSphere\CommerceServer\bin
- ▶ 2000 drive:\ProgramFiles\WebSphere\CommerceServer\bin
- ▶ AIX /usr/WebSphere/CommerceServer\bin
- ▶ Solaris /opt/WebSphere/CommerceServer/bin
- ▶ Linux /opt/WebSphere/CommerceServer/bin

b. 다음을 입력하십시오.

```
xmltransform -infile
..\xml\policies\xml\samplestorenameAccessPolicies -transform
..\xml\policies\xsl\accesscontrol.xsl -outfile
..\xml\policies\xml\samplestorenameAccessPoliciesOut.xml
▶ 400 TRANSFORM('/QIBM/ProdData/WebCommerce/xml/policies
/xsl/accesscontrol.xsl')
INSTROOT(instance_root) OUTFILE(output_file)
```

c. 다음 로그 파일을 확인하여 변환이 완료되었는지 확인하십시오.

- **NT** drive:\WebSphere\CommerceServer\bin\xmltransform.db2.log
 - **2000** drive:\Program Files\WebSphere\CommerceServer\bin\xmltransform.db2.log
 - **AIX** /usr/WebSphere/CommerceServer/bin/xmltransform.db2.log
 - **Solaris** /opt/WebSphere/CommerceServer/bin/xmltransform.db2.log
 - **Linux** /opt/WebSphere/CommerceServer/bin/xmltransform.db2.log
 - **400** /QIBM/UserData/WebCommerce/instances/
instancename/logs/TRNWCXML.tx
- 변환이 성공적이었다면, 다음 메시지가 표시됩니다.
- ```
"<DATE> <TIME> java.lang.Class main XMLTransformer Transform Successful"
```

5. xmltransform 명령을 실행하여 *samplestorenameAccessPolicies\_locale.xml* 을 변환하십시오.

a. 명령 프롬프트에서 디렉토리를 다음과 같이 변경하십시오.

- **NT** drive:\WebSphere\CommerceServer\bin
- **2000** drive:\ProgramFiles\WebSphere\CommerceServer\bin
- **AIX** /usr/WebSphere/CommerceServer\bin
- **Solaris** /opt/WebSphere/CommerceServer/bin
- **Linux** /opt/WebSphere/CommerceServer/bin

b. 다음을 입력하십시오.

```
xmltransform -infile
..\xml\policies\xml\samplestorenameAccessPolicies_locale.xml
-transform ..\xml\policies\xsl\accesscontrolns.xsl -outfile
..\xml\policies\xml\samplestorenameAccessPoliciesOut_locale.xml
```

c. **400** TRNWCXML INFILE(input file)

```
TRANSFORM('/QIBM/ProdData/WebCommerce/xml/policies/
xsl/accesscontrolns.xsl')
INSTROOT(instance_root) OUTFILE(output_file)
```

6. 변환된 XML 파일에 다음 변경을 수행하십시오.

- a. `samplestorenameAccessPoliciesOut.xml`에서 열기 및 닫기 태그를 다음과 같이 바꾸십시오.

```
<?xml version="1.0"?>
<!DOCTYPE accesscontrol-asset SYSTEM "accesscontrol.dtd">
<accesscontrol-asset>
</accesscontrol-asset>
```

- b. `samplestorenameAccessPoliciesOut_locale.xml`에서 열기 및 닫기 태그를 다음과 같이 변경하십시오.

```
<?xml version="1.0" encoding="correct language code for the file"?>
<!DOCTYPE accesscontrol-asset SYSTEM "../accesscontrol.dtd">
<accesscontrol-asset>
</accesscontrol-asset>
```






- c. `samplestorenameAccessPoliciesOut_locale.xml`에서 @locale을 &locale로 바꾸십시오. 예를 들면, LANGUAGE\_ID="@en\_US"를 LANGUAGE\_ID("&en\_US;"로 변경하십시오.

- d. `samplestorenameAccessPoliciesOut_locale.xml`에서 "acpoldesc" 테이블 참조를 찾으십시오. ACPOLICY\_ID 값 끝에 있는 @를 제거하십시오. 예를 들면, "@AllUsersExecuteInFashionAllUsersViews@"를 "@AllUsersExecuteInFashionAllUsersViews"로 변경하십시오.

- e. `samplestorenameAccessPoliciesOut.xml`에서 MEMBER\_ID="MEMBER\_ID"를 MEMBER\_ID("&MEMBER\_ID;"로 바꾸십시오.

- f. `samplestorenameAccessPoliciesOut.xml`에서 "acpolicy" 테이블 참조를 찾으십시오. ACPOLICY\_ID 값 끝에 있는 "@MEMBER\_ID"을 제거하십시오. 예를 들면, "@AllUsersExecuteInFashionAllUsersViews@MEMBER\_ID"를 "@AllUsersExecuteInFashionAllUsersViews"로 변경하십시오.

7. 상점에 대한 상점 아카이브 파일을 찾으십시오(예: mystore.sar). 상점 아카이브 파일은 기본값으로 다음 디렉토리에 위치합니다.

-  NT drive:\WebSphere\CommerceServer\instances\instancename\sar
-  2000 drive:\Program Files\WebSphere\CommerceServer\instances\instancename\sar
-  AIX /usr/WebSphere/CommerceServer/instances/instancename/sar
-  Solaris /opt/WebSphere/CommerceServer/instances/instancename/sar
-  Linux /opt/WebSphere/CommerceServer/instances/instancename/sar

- ▶ 400 /QIBM/UserData/WebCommerce/instances/instancename/sar

8. *samplestorenameAccessPoliciesOut.xml* 및 *samplestorenameAccessPoliciesOut\_locale.xml*를 *accesscontrol.xml*으로 변경하십시오.

주: 로케일 고유의 *accesscontrol.xml* 파일은 기본적으로 *data/en\_US* 처럼 *data/* 로케일 디렉토리에 위치합니다.

9. ZIP 프로그램을 사용하여 상점 아카이브 파일을 여십시오.
10. 상점 아카이브 파일에 있는 기존 *accesscontrol.xml* 및 로케일 고유의 *accesscontrol.xml*을 8단계에서 이름을 바꾼 파일들로 바꾸십시오.
11. 상점 아카이브 파일을 저장하십시오.

---

## 제 6 부 상점 패키지





---

## 제 25 장 상점 패키지

상점을 견본으로 사용하여 다른 상점에 배달하거나, 다른 서버나 플랫폼에 배치하거나 다른 상점을 작성하는 기초로서 사용하고자 한다면, 이를 상점 아카이브 형태로 패키징할 수 있습니다.

WebSphere Commerce와 함께 제공되는 견본 상점은 상점 아카이브로 패키징됩니다. 견본 상점 아카이브의 파일은 다음과 같이 그룹화됩니다.

- 웹 자원: HTML 파일, JSP 파일, 이미지 및 그래픽과 같은 상점 페이지를 작성하고 파일들을 포함하는 데 사용되는 파일. 웹 자원은 상점 아카이브에서 압축된 파일(webapp.zip)로 함께 그룹화됩니다.
- 특성 자원 번들(선택적): 상점 페이지의 텍스트 포함. 상점에서 둘 이상의 언어를 지원할 경우, 자원 번들에는 여러 번들이 포함될 수 있습니다. 즉, 언어당 하나의 번들이 있을 수 있습니다. 특성 자원 번들은 상점 아카이브에서 압축된 아카이브 파일(properties.zip)로 함께 그룹화됩니다.
- 상점 데이터 자원: 데이터베이스에 로드할 데이터. 상점 데이터 자원에는 캠페인, 카탈로그, 통화, 이행 정보, 가격 책정, 운송, 저장 및 세금 정보와 같은 데이터가 포함됩니다.
- 지불 자원: IBM Payment Manager의 구성 정보.
- 설명자: XML 파일 sarinfo.xml은 웹 자원 압축 아카이브 파일, 자원 번들 및 상점 데이터베이스 자원 XML 파일을 포함하여 상점 아카이브를 설명합니다. sarinfo.xml 파일에는 또한 공개 처리 동안 필요한 아카이브 파일에 대한 정보 뿐만 아니라 포함 파일과 일관성 확인 파일의 이름이 포함됩니다. sarinfo.xml은 상점 아카이브에서 유일한 필수 파일입니다.





---

### 상점 아카이브 작성

상점 아카이브로 상점을 패키징하려면, 다음을 수행하십시오.

1. WebSphere Commerce와 함께 제공된 견본 상점 아카이브의 콘텐츠와 구조를 검토하십시오.

상점 아카이브 파일은 다음 디렉토리에 있습니다.

-  NT drive:\WebSphere\CommerceServer\samplstores
-  2000 drive:\Program Files\WebSphere\CommerceServer\samplstores
-  AIX /usr/WeSphere/CommerceServer/samplstores
-  Solaris /opt/WebSphere/CommerceServer/samplstores

- ▶ Linux /opt/WebSphere/CommerceServer/samplestores
- ▶ 400 /qibm/proddata/WebCommerce/samplestores

상점 아카이브를 보려면, 압축 풀기 프로그램을 사용하십시오.

2. 상점의 WebSphere Commerce 서버에서 임시 디렉토리를 작성하십시오(예: *mystore*).
3. 임시 디렉토리에서 *webapp*라는 디렉토리에 웹 자원(JSP 파일, HTML, 이미지)을 함께 그룹화하십시오. *webapp* 폴더의 ZIP 프로그램을 사용하여 압축된 아카이브 파일을 작성하십시오.
4. (선택적) 임시 디렉토리에서 *properties*라는 디렉토리에 특성 자원 번들을 함께 그룹화하십시오. 상점에서 둘 이상의 언어를 지원할 경우, 자원 번들에는 여러 번들이 포함될 수 있습니다. 즉, 언어당 하나의 번들이 있을 수 있습니다. *properties* 폴더의 ZIP 프로그램을 사용하여 압축된 아카이브 파일을 작성하십시오.
5. 임시 디렉토리의 *data*라는 디렉토리에 상점 데이터 자원을 함께 그룹화하십시오. 상점이 여러 언어를 지원하는 경우, 로케일 이름을 사용하여 언어별 정보에 대한 서브디렉토리를 작성하십시오(예: *en\_US*).
6. (선택적) 기존 상점 아카이브에서 *data* 디렉토리로 *sarrule.xml* 파일을 복사하십시오. *sarrule.xml* 파일은 견본 상점 아카이브에서 *data* 디렉토리에 있습니다. 상점 서비스를 사용하여 공개할 때 *sarrule.xml*은 일관성 확인 프로그램의 역할을 합니다. *sarrule* 파일에 대한 자세한 정보는 373 페이지의 부록 D 『*sarrule.xml*』을 참조하십시오.
7. 임시 디렉토리에서 *SAR-INF*라는 디렉토리를 작성하십시오.
8. 상점 아카이브에 대한 *sarinfo.xml* 파일을 작성하십시오. XML 스펙에 대한 자세한 정보는 다음 디렉토리에서 아카이브 설명자 *sarinfo.dtd*를 참조하십시오.

- ▶ NT drive:\WebSphere\CommerceServer\xml\sar
- ▶ 2000 drive:\Program Files\WebSphere\CommerceServer\xml\sar
- ▶ AIX /usr/WebSphere/CommerceServer/xml/sar
- ▶ Solaris /opt/WebSphere/CommercServer/xml/sar
- ▶ Linux /opt/WebSphere/CommercServer/xml/sar
- ▶ 400 /qibm/proddata/WebCommerce/xml/sar

- a. 365 페이지의 부록 C 『*sarinfo.xml*』 예와 정보를 참조하여 *sarinfo.xml* 파일을 작성하십시오. 일부 데이터 자원이 다른 자원 이전에 공개되어야 하므로 데이터 자원이 공개된 순서는 중요합니다. 결과적으로 *sarinfo.xml* 파일에 지정된 대로 사용자 자원의 순서는 견본 상점에 대한 *sarinfo.xml* 파일에 지정된 자원의 순서와 일치해야 합니다.

주: 상점 아카이브(6단계 참조)에 `sarrule.xml` 파일을 추가하는 것을 선택하면, `sarinfo.xml` 파일에 있는 `sarrule.xml`에 관한 정보를 포함해야 합니다. `sarrule.xml` 파일을 포함하지 않는 것을 선택하면, `sarinfo.xml` 파일에서 이 파일을 언급하지 않도록 주의하십시오.

b. 6단계에서 작성한 SAR-INF 디렉토리에 `sarinfo.xml`를 저장하십시오.

9. 웹 자원 ZIP 파일, 특성 번들, 상점 데이터 자원 및 `sar-inf` 디렉토리로 구성된 ZIP 파일을 작성하십시오. 이 ZIP 파일을 `storearchivename.sar`라고 이름 지정하십시오.

10. 상점 서비스를 사용하여 상점 아카이브를 편집하거나 공개하려는 경우, `storearchivename.sar`을 다음 디렉토리로 저장하십시오.

- ▶ NT drive:\WebSphere\CommerceServer\instances\*instancename*\sar
- ▶ 2000 drive:\Program Files\WebSphere\CommerceServer\instances\*instancename*\sar
- ▶ AIX /usr/WebSphere/CommerceServer/instances/*instancename*/sar
- ▶ Solaris /opt/WebSphere/CommerceServer/instances/*instancename*/sar
- ▶ Linux /opt/WebSphere/CommerceServer/instances/*instancename*/sar
- ▶ 400 /QIBM/UserData/WebCommerce/instances/*instancename*/sar

11. 이제 상점 아카이브는 상점 서비스에서 상점 아카이브의 목록에 표시됩니다.

---

## 견본 상점 아카이브 작성

상점을 상점 아카이브로 패키징한 후에, 상점 서비스에서 견본 상점으로 사용하도록 선택할 수 있습니다. 견본 상점 아카이브는 새 상점을 작성하기 위한 기준으로 복사되고 사용됨을 의미하는 상점 아카이브입니다. 견본 상점 아카이브로 상점 아카이브를 사용하려면, 다음을 수행하십시오.

1. 상점 아카이브 파일을 다음 디렉토리에 저장하십시오.

- ▶ NT drive:\WebSphere\CommerceServer\samplestores\*storearchivename*
- ▶ 2000 drive:\Program Files\WebSphere\CommerceServer\samplestores\*storearchivename*
- ▶ AIX /usr/WebSphere/CommerceServer/samplestores/*storearchivename*

- `> Solaris` /opt/WebSphere/CommerceServer/samplestores/  
storeachivename
  - `> Linux` /opt/WebSphere/CommerceServer/samplestores/  
storeachivename
  - `> 400` /qibm/proddata/WebCommerce/samplestores/storeachivename
2. (선택적) 미리보기 페이지를 작성하십시오. 상점 페이지의 미리보기가 상점 서비스를 표시하게 하려면, 미리보기 페이지를 작성해야 합니다. 다음을 수행하십시오.
- a. (선택적) 상점 서비스에서 새로 만들기를 선택하십시오. 상점 아카이브 작성 페이지가 표시됩니다. 견본 목록에서 견본 상점 중 하나를 선택한 다음, 미리보기를 누르십시오. 표시하는 페이지를 미리보기 페이지라고 합니다. 이 페이지는 사전 정의된 견본 쇼핑 플로우를 나타내며 견본 상점의 미리보기로 역할하는 HTML 파일입니다.
  - b. 미리보기 페이지에서 표시하려는 쇼핑 플로우를 결정합니다.
  - c. (선택적) 공개된 상점에서 견본 데이터를 작성하십시오. 예를 들어, 항목을 장바구니에 추가하고, 몇 개의 운송 주소와 지불 청구 주소를 작성하십시오. 이 상점에서 미리보기 페이지를 작성하게 되며 데이터는 페이지가 더 현실적으로 보이게 합니다.
  - d. Internet Explorer를 사용하여 상점을 찾아보십시오. 파일, 다른 이름으로 저장을 선택하여 각 페이지에 대한 HTML을 저장하십시오. 또한 스타일 시트(.css) 및 이미지를 저장해야 합니다. 파일을 다음 디렉토리에 저장하십시오.
    - `stylesheet.css`
      - `> NT` drive:\WebSphere\AppServer\installedApps\  
WC\_Enterprise\_App\_instancename.ear\wcstools.war\  
tools\devtools\storearchivename\preview
      - `> 2000` drive:\Program Files  
\WebSphere\AppServer\installedApps\  
WC\_Enterprise\_App\_instancename.ear\wcstools.war\  
\tools\devtools\storearchivename\preview
      - `> AIX` /usr/WebSphere/AppServer/installedApps  
WC\_Enterprise\_App\_instancename.ear/wcstools.war  
/tools/devtools/storearchivename/preview
      - `> Solaris` /opt/WebSphere/AppServer/installedApps/  
WC\_Enterprise\_App\_instancename.ear/wcstools.war  
/tools/devtools/storearchivename/preview

- ▶ Linux /opt/WebSphere/AppServer/installedApps/  
WC\_Enterprise\_App\_instancename.ear/wcstools.war  
/tools/devtools/storearchivename/preview
- ▶ 400 /Qibm/UserData/WebAsAdv4/WASinstancename/  
installedApps/WC\_Enterprise\_App\_instancename.ear  
/wcstools.war/tools/devtools/storearchivename/preview
- HTML
  - ▶ NT drive:\WebSphere\AppServer\installedApps\  
WC\_Enterprise\_App\_instancename.ear\wcstools.war  
\tools\devtools\storearchivename\preview\locale
  - ▶ 2000 drive:\Program  
Files\WebSphere\AppServer\installedApps\  
WC\_Enterprise\_App\_instancename.ear\wcstools.war\  
tools\devtools\storearchivename\preview\locale
  - ▶ AIX /usr/WebSphere/AppServer/installedApps/  
WC\_Enterprise\_App\_instancename.ear/wcstools.war/tools  
/devtools/storearchivename/preview/locale
  - ▶ Solaris /opt/WebSphere/AppServer/installedApps/  
WC\_Enterprise\_App\_instancename.ear/wcstools.war/tools  
/devtools/storearchivename/preview/locale
  - ▶ Linux /opt/WebSphere/AppServer/installedApps/  
WC\_Enterprise\_App\_instancename.ear/wcstools.war/tools  
/devtools/storearchivename/preview/locale
  - ▶ 400 /Qibm/UserData/WebAsAdv4/WASinstancename/  
installedApps/WC\_Enterprise\_App\_instancename.ear  
/wcstools.war/tools/devtools/storearchivename/preview/  
locale
- 로케일 독립 이미지
  - ▶ NT drive:\WebSphere\AppServer\installedApps\  
WC\_Enterprise\_App\_instancename.ear\wcstools.war\  
tools\devtools\storearchivename\preview\images
  - ▶ 2000 drive:\Program  
Files\WebSphere\AppServer\installedApps\  
WC\_Enterprise\_App\_instancename.ear\wcstools.war  
\tools\devtools\storearchivename\preview\images

- ▶ AIX /usr/WebSphere/AppServer/installedApps/  
WC\_Enterprise\_App\_instancename.ear/wcstools.war/  
tools/devtools/storearchivename/preview/images
- ▶ Solaris /opt/WebSphere/AppServer/installedApps/  
WC\_Enterprise\_App\_instancename.ear/wcstools.war/  
tools/devtools/storearchivename/preview/images
- ▶ Linux /opt/WebSphere/AppServer/installedApps/  
WC\_Enterprise\_App\_instancename.ear/wcstools.war/  
tools/devtools/storearchivename/preview/images
- ▶ 400 /Qibm/UserData/WebAsAdv4/WASinstancename/  
installedApps/WC\_Enterprise\_App\_instancename.ear  
/wcstools.war/tools/devtools/storearchivename/preview/  
images
- 로케일 종속 이미지
  - ▶ NT drive:\WebSphere\AppServer\installedApps\  
WC\_Enterprise\_App\_instancename.ear\wcstools.war\tools  
\devtools\storearchivename\preview\locale\images
  - ▶ 2000 drive:\Program Files\WebSphere\AppServer\  
installedApps\WC\_Enterprise\_App\_instancename.ear\  
wcstools.war\tools\devtools\storearchivename  
\preview\locale\images
  - ▶ AIX /usr/WebSphere/AppServer/installedApps/  
WC\_Enterprise\_App\_instancename.ear/wcstools.war/tools/  
devtools  
/storearchivename/preview/locale/images
  - ▶ Solaris /opt/WebSphere/AppServer/installedApps/  
WC\_Enterprise\_App\_instancename.ear/wcstools.war/tools/  
devtools  
/storearchivename/preview/locale/images
  - ▶ Linux /opt/WebSphere/AppServer/installedApps/  
WC\_Enterprise\_App\_instancename.ear/wcstools.war/tools/  
devtools  
/storearchivename/preview/locale/images

- ▶ 400 /Qibm/UserData/WebAsAdv4/WASinstancename/  
 installedApps/WC\_Enterprise\_App\_instancename.ear  
 /wcstools.war/tools/devtoolsstorearchivename/preview  
 /locale/images

- e. 이미지의 위치와 css 파일이 변경되었으므로, HTML 페이지에서 이미지와 css 파일에 대한 참조를 변경해야 합니다. 참조를 변경한 후에, 브라우저에서 HTML 페이지를 열 때 이미지를 볼 수 있는지 확인하십시오.
  - f. 명령에서 HTML 페이지에 있는 링크를 HTML 파일을 참조하는 링크로 변경하십시오.
3. 상점 아카이브를 요약하는 HTML 파일을 작성하십시오. 이 정보는 상점 서비스의 상점 아카이브 작성 페이지에서 **견본 설명**에 표시됩니다.

a. 다음 예를 참조하여 새 파일을 작성하십시오.

```
<doctype html public "-//w3c//dtd html 4.0 transitional//en">
<html>
<head>
 <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>
<body>
Describe store here
</body>
</html>
```

b. 이 파일을 `Feature_locale.html`으로 저장하십시오. 여기서 로케일은 사용 중인 언어에 대한 약어입니다(예: `en_US`). 이 파일을 다음 디렉토리에 저장하십시오.

- ▶ NT drive:\WebSphere\CommerceServer\  
samplestores\storearchivename
- ▶ 2000 drive:\Program Files\WebSphere\CommerceServer\  
\samplestores\storearchivename
- ▶ AIX /usr/WebSphere/CommerceServer/samplestores/  
storearchivename
- ▶ Solaris /opt/WebSphere/CommerceServer/samplestores/  
storearchivename
- ▶ Linux /opt/WebSphere/CommerceServer/samplestores/  
storearchivename
- ▶ 400 /qibm/proddata/WebCommerce/samplestores/  
storearchivename

4. 상점 아카이브를 `sarregistry.xml` 파일에 추가하십시오. `sarregistry.xml` 파일은 상점 서비스의 상점 아카이브 작성 페이지에서 **견본 목록**에 어떤 상점 아카이

브를 표시할지 결정합니다. 또한 sarregistry.xml은 어떤 미리보기 페이지와 특징 파일이 각 상점 아카이브와 연관되는지 결정합니다. sarregistry.xml은 다음 디렉토리에 있습니다.

- ▶ NT drive:\WebSphere\CommerceServer\xml\tools\devtools
- ▶ 2000 drive:\Program Files\WebSphere\CommerceServer\xml\tools\devtools
- ▶ AIX /usr/WebSphere/CommerceServer/xml/tools/devtools
- ▶ Solaris /opt/WebSphere/CommerceServer/xml/tools/devtools
- ▶ Linux /opt/WebSphere/CommerceServer/xml/tools/devtools
- ▶ 400 /QIBM/ProdData/WebCommerce/xml/tools/devtools

a. 다음 예를 참조하여 새 상점 아카이브를 sarregistry.xml에 추가하십시오.

```
<SampleSAR fileName="infashion_en_US_es_ES.sar" relativePath="InFashion">
 <html locale="es_ES" featureFile="InFashion/Feature_es_ES.html"
 sampleSite="RetailModel/preview/es_ES/index.html"/>
 <html locale="en_US" featureFile="InFashion/Feature_en_US.html"
 sampleSite="RetailModel/preview/en_US/index.html"/>
</SampleSAR>
```

이 예는 영어 및 스페인어 InFashion 상점 아카이브에 대한 미리보기 페이지를 정의합니다. 굵은체 행은 영어 및 스페인어 미리보기 페이지에 대한 홈페이지를 정의합니다.

5. 이제 상점 서비스의 상점 아카이브 작성 페이지에서 견본 목록에 있는 상점 아카이브를 볼 수 있어야 합니다.



---

## 제 7 부 상점 공개

기능하는 상점을 작성하려면, 상점 입구 웹 자원을 WebSphere Commerce 서버에 공개해야 하고 상점 데이터를 WebSphere Commerce 데이터베이스에 공개해야 합니다.

다음 장에서는 WebSphere Commerce가 제공하는 공개 옵션에 대해 논의합니다.

- 249 페이지의 제 26 장 『완성된 상점 공개』 - 이 장에서는 상점이 상점 아카이브의 양식에 있는 경우, 상점 서비스 또는 명령행 공개를 사용한 전체 상점 공개(상점 입구 및 상점 데이터 자원)를 논의합니다.
- 265 페이지의 제 27 장 『상점 데이터 로드 개요』 - 이 장에서는 로더 패키지 및 WebSphere Catalog Manager를 사용한 데이터베이스로의 상점 데이터 자원 공개를 논의합니다.
- 307 페이지의 제 28 장 『WebSphere Commerce 데이터베이스 자원 그룹 로드』 - 이 장에서는 로더 패키지 및 WebSphere Catalog Manager의 다른 구성요소를 사용한 데이터베이스로의 모든 상점 데이터 또는 상점 데이터 자원 그룹 공개를 논의합니다.
- 323 페이지의 제 29 장 『비즈니스 계정 및 장기 구매 계약 공개』 - 이 장에서는 계정, 장기 구매 계약 및 상품 세트 자원 공개를 논의합니다.
- 327 페이지의 제 30 장 『상점 입구 자원 및 상점 구성 파일 공개』 - 이 장에서는 상점 입구 자원 및 상점 구성 파일 공개를 논의합니다.



---

## 제 26 장 완성된 상점 공개

기능하는 상점을 작성하려면, 상점 입구 웹 자원을 WebSphere Commerce 서버에 공개해야 하고, 상점 데이터를 WebSphere Commerce 데이터베이스에 공개해야 합니다. 이 장에서는 상점이 상점 아카이브의 양식에 있는 경우, 상점 서비스 또는 명령행 공개를 사용 전체 상점 공개(상점 입구 및 상점 데이터 자원)를 논의합니다.

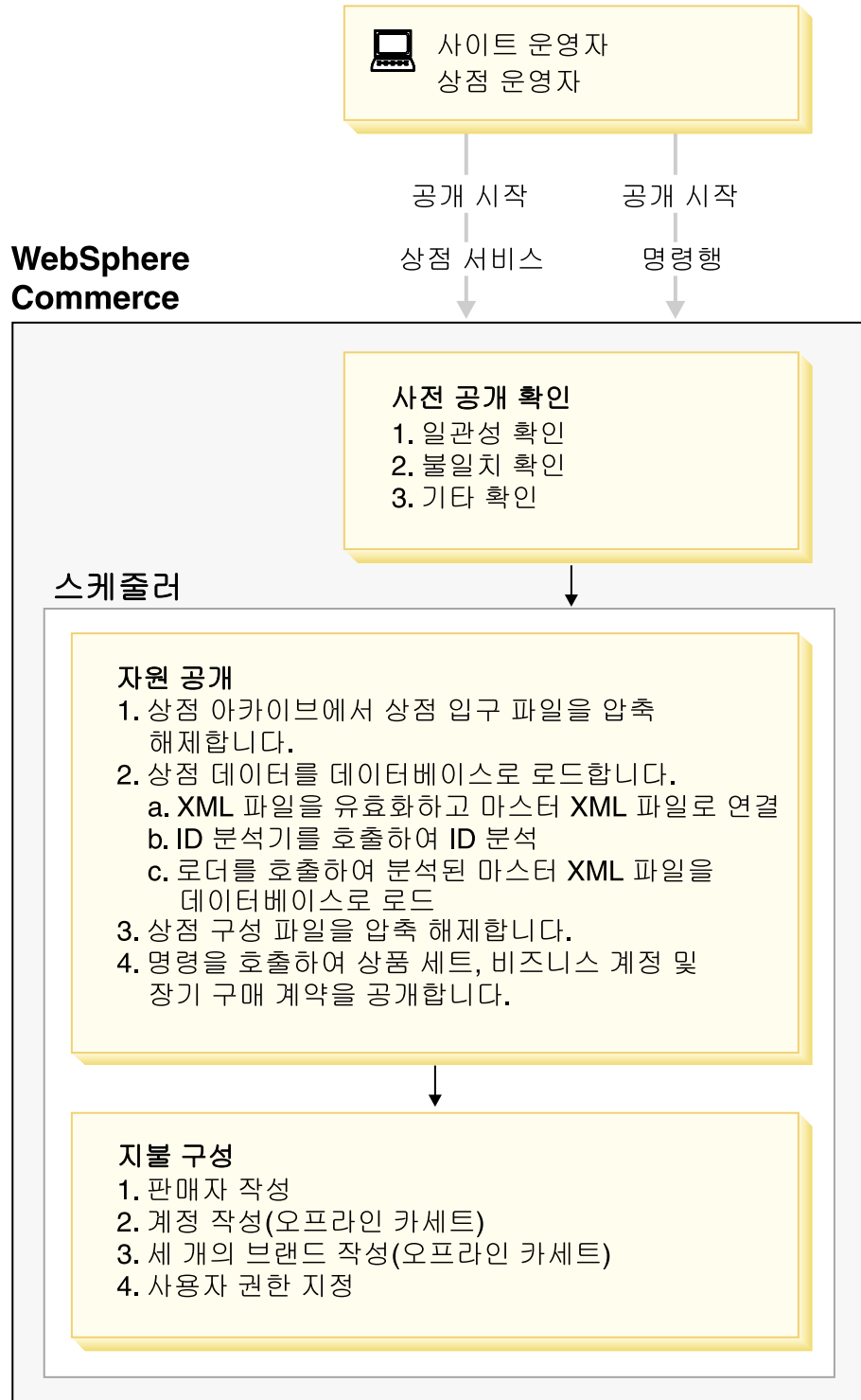
주: 상점 아카이브로 상점을 패키지하지 않으려는 경우, 개별적으로 자원을 공개할 수 있습니다. 자세한 정보는 265 페이지의 제 27 장 『상점 데이터 로드 개요』, 307 페이지의 제 28 장 『WebSphere Commerce 데이터베이스 자원 그룹 로드』, 323 페이지의 제 29 장 『비즈니스 계정 및 장기 구매 계약 공개』 및 327 페이지의 제 30 장 『상점 입구 자원 및 상점 구성 파일 공개』를 참조하십시오.

---

### WebSphere Commerce의 공개 이해

상점 서비스 또는 명령행에서 사용 가능한 공개 옵션을 사용하여 완성된 상점(상점 입구 및 상점 데이터 자원)을 한번에 공개할 수 있습니다. 이 옵션을 사용하려면, 상점 자원은 상점 아카이브 양식으로 패키지되어야 합니다. 상점 아카이브로 상점을 패키지하는 자세한 정보에 대해서는 237 페이지의 제 6 부 『상점 패키지』를 참조하십시오.

다음 도표는 공개 프로세스의 단계를 나타냅니다.



## 시작 공개

상점을 공개하려면, 사이트 운영자 또는 상점 운영자(모든 상점에 대해) 권한이 있어야 합니다. 사이트 운영자 또는 상점 운영자는 다음 방법 중 하나를 사용하여 공개 프로세스를 시작할 수 있습니다.

- 상점 서비스
- 명령행

두 가지 공개 방법을 수행하려면 공개하려는 상점 아카이브를 선택해야 합니다. 어떤 방법을 사용하든지 공개를 원하는 상점 아카이브의 양을 선택해야 합니다. 예를 들어, 다음 모든 조합 중 하나를 선택하여 공개할 수 있습니다.

- 온라인 카탈로그 데이터가 있거나 없는 상점 데이터베이스 자원
- JSP 파일, HTML 파일 및 이미지와 같은 웹 자원
- 특성 파일(상점에 대한 텍스트)

처음 공개할 경우, 기능하는 상점을 볼 수 있도록 전체 상점 아카이브(위의 모두)를 공개하는 것을 권장합니다. 하지만 연속적인 공개의 경우, 전체 상점 아카이브를 다시 공개하는 대신 데이터베이스 자원, 웹 자원 또는 특성 자원 번들만을 갱신하고자 할 수 있습니다.

상점 아카이브를 처음 공개할 때 온라인 카탈로그 데이터를 공개하지 않도록 선택한 경우, 일부 데이터 자원(장기 구매 계약 및 계정)은 카탈로그의 데이터에 의존하므로 올바르게 공개되지 않을 수도 있습니다. 또한 처음 공개할 때 카탈로그 데이터 및 웹 자원 모두를 공개하지 않는 경우, 상점 ID, 카탈로그 ID 및 언어 ID가 작성되지 않으며 상점 서비스에서 상점을 시작할 수 없습니다. 자세한 정보는 260 페이지의 『parameters.jsp 파일 작성』을 참조하십시오.

온라인 카탈로그 데이터는 상점 아카이브에서 다음 XML 파일로 구성되어 있습니다.

- catalog.xml
- offering.xml
- store-catalog.xml
- store-catalog-shipping.xml
- store-catalog-tax.xml
- storefulfill.xml

이 정보를 나중에 공개하도록 선택한 경우, 상점 서비스 또는 로더 패키지를 통해 이 정보를 공개할 수 있습니다. 데이터베이스 자원 그룹(예를 들어, 위에 나열된 카탈로그 그룹 데이터)을 사용하여 데이터의 서브세트 로드와 관련한 자세한 사항은 316 페이지의 『데이터베이스 자원 그룹 로드』를 참조하십시오.

상점 서비스 또는 명령행을 사용하여 상점 아카이브를 공개하는 방법에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오.

주: 상점 서비스 또는 명령행을 사용하여 공개 프로세스를 시작한 후에는 아무것도 할 필요가 없습니다. 이전 도표 및 이 장에서 나열된 모든 단계가 WebSphere Commerce 시스템에 의해 완료됩니다.

## 사전 공개 확인

일단, 사이트 또는 상점 운영자에 의해 공개가 시작되면, WebSphere Commerce는 실제 공개 프로세스를 시작하기 전에 몇 가지 확인을 수행합니다. 이 확인 사항은 다음과 같습니다.

- 일관성 확인
- 불일치 확인
- 기타 확인

주: 명령행 공개는 이에 전달된 매개변수를 확인하며 일관성 및 불일치 점검을 완료합니다. 그러나 명령행 공개를 사용하여 공개할 경우, 다음 단락에 설명된 메시지를 볼 수 없습니다. 대신에, 갱신 모드를 사용하여 공개할 경우, 명령행이 사용자를 프롬프트하지 않고 기존 상점에 겹쳐씹니다. 삽입 모드를 사용하여 공개하는 경우, 명령행이 새 상점을 로드합니다. 공개에 실패하면 공개 실패 오류 메시지가 표시됩니다.

### 일관성 확인

사전 공개 확인은 `sarrule.xml` 파일의 규칙을 사용하여 XML 파일의 정보가 상점 아카이브의 웹 자원과 일치하도록 합니다. 예를 들어, `command.xml` 파일이 특정 JSP 파일을 참조하는 경우, 해당 JSP 파일이 상점 아카이브의 `webapp.zip` 파일에 있어야 합니다. 일관성 확인에서 오류를 발견하는 경우, 오류는 로그에 작성되지만 공개는 정상적으로 계속됩니다. 로그 파일에 대한 자세한 정보는 262 페이지의 『로그 파일 공개』를 참조하십시오.

`sarrule.xml` 파일의 예제에 대해서는 373 페이지의 부록 D 『`sarrule.xml`』을 참조하십시오.

### 불일치 확인

사전 공개 확인을 하는 동안 WebSphere Commerce는 상점과 카탈로그 사이의 불일치를 확인합니다. 특히, 이는 데이터베이스에 대해 다음을 확인합니다.

- SOTREENT 테이블에서 동일한 식별자를 가지는 기존의 상점: 상점이 이미 있는 경우, 메시지는 상점을 겹쳐씹 것인지 또는 공개를 취소할 것인지 여부를 묻습니다.
- 동일한 인식 이름을 가지는 기존의 카탈로그: 카탈로그가 이미 있는 경우, 메시지는 카탈로그를 겹쳐씹 것인지 또는 공개를 취소할 것인지 여부를 묻습니다. 카탈로그가

다른 상점에 속하는 경우, 카탈로그가 어떤 상점에 속하는지를 알려주는 메시지가 표시되며, 공개를 계속하고 카탈로그를 겹쳐쓸 것인지 또는 공개를 취소할 것인지 여부를 묻습니다.


### 기타 확인

사전 공개를 확인하는 동안 WebSphere Commerce는 또한 스케줄러가 사용 가능한지, 캐시 트리거와 캐시가 사용 불가능한지 그리고 요약 테이블이 사용 불가능한지 확인합니다. 그렇지 않은 경우, 경고 메시지가 표시되어 어떤 기능이 부적절하게 사용 또는 사용 불가능한지 알려줍니다.

주: 명령행 공개는 이 확인을 완료하지 않습니다.

**스케줄러:** 다음 절에서 좀더 자세하게 논의되겠지만, 스케줄러는 공개 작업을 실행합니다. 스케줄러가 사용 불가능한 경우, 공개 프로세스를 실행할 수 없습니다.

**캐시 및 캐시 트리거:** 공개하는 동안 캐시를 사용하면 데이터베이스 갱신 시 캐시가 호출되는 결과를 야기합니다. 캐시 트리거는 데이터베이스 트랜잭션 로그 오버플로우를 야기하고 불필요한 데이터베이스 활동을 생성하여 공개 성능에 영향을 줍니다. 캐시 트리거를 사용 불가능하게 하려면, WebSphere Commerce 온라인 도움말을 참조하십시오.

**요약 테이블:**  공개하는 동안, 요약 테이블을 사용하면 요약 테이블이 갱신되며 이는 데이터베이스 트랜잭션 로그 오버플로우를 야기하여 공개 성능에 영향을 줍니다. 요약 테이블을 사용 불가능하게 하려면, WebSphere Commerce 온라인 도움말을 참조하십시오.

## 자원 공개

공개 프로세스의 자원 공개 단계는 스케줄러에 의해 실행되는 스케줄된 작업입니다. 스케줄러가 공개 작업을 실행할 때, WebSphere Commerce는 다음 조치를 완료합니다.

- 상점 아카이브에서 상점 입구 파일을 압축 해제합니다.
- 상점 데이터를 상점 아카이브의 XML 파일에서 데이터베이스로 로드합니다.
- parameters.jsp 파일을 작성합니다.
- 상점 구성 파일을 압축 해제합니다.
- 비즈니스 계정 및 장기 구매 계약을 공개하려면 명령을 호출합니다.
- 레지스트리 구성요소를 갱신합니다.

### 상점 아카이브에서 상점 입구 파일을 압축 해제합니다.

상점 아카이브에서 웹 자원을 WebSphere Commerce 서버로 압축 해제하는 것은 자원 공개 단계의 첫 번째 조치입니다. 상점 아카이브에서 파일을 압축 해제하는 동안, WebSphere Commerce는 다음을 수행합니다.

**웹 자원을 압축 해제하고 WebSphere Commerce 서버의 다음 위치로 복사:**

- 파일, 이미지 및 그래픽을 포함하는 HTML 및 JSP 파일이 상점 웹 응용프로그램 문서 루트 아래의 상점 디렉토리(*storedir*)로 공개됩니다.

- **▶ NT** drive:\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_instancename.ear\wcstores.war\storedir

- **▶ 2000** drive:\Program Files\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_instancename.ear\wcstores.war\storedir

- **▶ AIX** /usr/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_instancename.ear/wcstores.war/storedir

- **▶ Solaris** /opt/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_instancename.ear/wcstores.war/storedir

- **▶ Linux** /opt/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_instancename.ear/wcstores.war/storedir

- **▶ 400** /QIBM/UserData/WebASAdv4/WASinstancename/installedApps/WC\_Enterprise\_App\_instancename.ear/wcstores.war/storedir

- 자원 번들 및 특성 파일이 응용프로그램 특성 경로로 공개됩니다.

- **▶ NT** drive:\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_instancename.ear\wcstores.war\WEB-INF\classes\storedir


- **▶ 2000** drive:\Program Files\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_instancename.ear\wcstores.war\WEB-INF\classes\storedir

- **▶ AIX** /usr/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_instancename.ear/wcstores.war/WEB-INF/classes/storedir

- **▶ Solaris** /opt/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_instancename.ear/wcstores.war/WEB-INF/classes/storedir

- **▶ Linux** /opt/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_instancename.ear/wcstores.war/WEB-INF/classes/storedir



–  400 /QIBM/UserData/WebASAdv4/WASinstancename/installedApps/WC\_Enterprise\_App\_instancename.ear/wcstores.war/WEB-INF/classes/storedir

### 상점 아카이브의 XML 파일에서 상점 데이터를 데이터베이스로 로드합니다.






상점 데이터를 상점 아카이브의 XML 파일에서 데이터베이스로 로드하는 동안, WebSphere Commerce는 다음을 수행합니다.

주: DB 로드 유형의 XML 파일만이 데이터베이스로 로드됩니다. 파일 유형은 sarinfo.xml 파일에서 지정됩니다. sarinfo.xml 파일에 대한 자세한 정보는 365 페이지의 부록 C 『sarinfo.xml』을 참조하십시오.

상점 아카이브에서 XML 파일을 유효화하고 마스터 XML 파일로 연결: WebSphere Commerce는 관련 DTD 파일을 사용하여 XML 파일을 유효화합니다. DTD 파일은 다음 디렉토리에 있습니다.

-  NT drive:\WebSphere\CommerceServer\xml\sar
-  2000 drive:\Program Files\WebSphere\CommerceServer\xml\sar
-  AIX /usr/WebSphere/CommerceServer/xml/sar
-  Solaris /opt/WebSphere/CommerceServer/xml/sar
-  Linux /opt/WebSphere/CommerceServer/xml/sar
-  400 /qibm/proddata/WebCommerce/xml/sar

XML 파일이 올바르지 않은 경우, WebSphere Commerce는 오류를 오류 로그에 작성합니다. 오류 로그의 오류가 WebSphere Commerce 구성 파일 *instance\_name.xml*의 DevTools 위치에 지정된 오류의 최대 갯수(기본값으로 MaxErrorsInSarXML=1)를 초과하는 경우, 공개는 실패합니다. WebSphere Commerce 구성 파일 *instance\_name.xml*은 다음 디렉토리에 있습니다.

-  NT drive:\WebSphere\CommerceServer\instances\*instancename*\xml\*instance\_name.xml*
-  2000 drive:\Program Files\WebSphere\CommerceServer\instances\*instancename*\xml\*instance\_name.xml*
-  AIX /usr/WebSphere/CommerceServer/instances/*instancename*/xml/*instance\_name.xml*
-  Solaris /opt/WebSphere/CommerceServer/instances/*instancename*/xml/*instance\_name.xml*
-  Linux /opt/WebSphere/CommerceServer/instances/*instancename*/xml/*instance\_name.xml*

- ▶ 400 /QIBM/UserData/WebCommerce/instances/*instancename*  
/xml/*instance\_name.xml*

XML 파일을 유효화한 뒤 하나의 파일 *storenamemaster.xml*로 연결합니다. 이 파일은 *sarinfo.xml* 파일에 지정된 우선순위에 따라 연결됩니다. 자세한 정보는 365 페이지의 부록 C 『*sarinfo.xml*』을 참조하십시오. *storenamemaster.xml* 파일은 다음 디렉토리에 있습니다.

- ▶ NT drive:\WebSphere\CommerceServer\temp\*instancename*\tools\devtools
- ▶ 2000 drive:\Program Files\WebSphere\CommerceServer\temp\*instancename*\tools\devtools
- ▶ AIX /usr/WebSphere/CommerceServer/temp/*instancename*/tools/devtools
- ▶ Solaris /opt/WebSphere/CommerceServer/temp/*instancename*/tools/devtools
- ▶ Linux /opt/WebSphere/CommerceServer/temp/*instancename*/tools/devtools
- ▶ 400 /QIBM/UserData/WebCommerce/instances/*instancename*  
/temp/tools/devtools

**ID 분석기를 호출하여 ID 분석:** 로더 패키지 유틸리티인 ID 분석기는 상점 아카이브 XML 파일의 XML 요소에 대하여 고유한 식별자를 생성합니다. 예를 들어, ID 분석기는 견본 상점 XML 파일에서 사용되는 @ 별명을 고유한 값으로 대체합니다. 견본 상점에서 사용되는 내부 별명 분석의 예에 대해서는 361 페이지의 부록 B 『데이터 작성』을 참조하십시오.

**주:** ID 분석기는 또한 공개시 이미 공개된 상점에 대해서도 ID를 분석할 수 있습니다. 예를 들어, 상점 아카이브를 한번 공개했을 경우, 해당 상점 아카이브 또는 그 부분을 재공개할 필요가 있습니다. ID 분석기는 데이터베이스에서 고유 식별자를 검색하고 재공개 처리 동안 이를 사용합니다.

ID 분석기 및 로더 패키지의 기타 구성요소에 대한 자세한 정보는 265 페이지의 제 27 장 『상점 데이터 로드 개요』를 참조하십시오.

상점 서비스 또는 명령행 공개에서 ID 분석기를 호출하려면, 사용할 ID 분석기 방법을 지정해야 합니다. ID 분석기에는 몇 가지가 있습니다. 이는 ID 분석기 입력을 처리하기 위해 사용되는데, 특히, 데이터를 식별자가 원시 데이터에 있는 것처럼(갱신 방법) 또는 없는 것처럼(로드 방법) 처리할 것인지 여부를 결정하게 합니다. 혼합 방법이 일부 식별자가 존재하고 나머지 식별자는 존재하지 않을 때 사용됩니다.

주: 혼합 방법이 상점 서비스에 대해 권장되는 방법입니다. 명령행을 사용하여 공개하고 있고 특히 데이터베이스에 이미 공개된 데이터의 존재 여부를 모를 경우 혼합 모드로 지정하려고 할 수 있습니다.

WebSphere Commerce 구성 파일 *instance\_name.xml*에서 사용되는 상점 서비스 또는 명령행 공개를 지정할 수 있습니다. 기본적으로, 상점 서비스는 혼합 방법을 사용합니다. ID 분석기 방법에 대한 자세한 정보는 269 페이지의 『ID Resolve 명령』을 참조하십시오.

상점 서비스 및 명령행 공개는 또한 D 분석기에 사용할 사용자 정의 파일을 지정해야 합니다. WebSphere Commerce 구성 파일 *instance\_name.xml*의 사용자 정의 파일을 지정하지 않는 경우, 공개 코드는 기본 사용자 정의 파일 중 하나를 사용합니다 (DBConnectionCustomizer 또는 OracleConnectionCustomizer).

### ▶ Oracle

OracleConnectionCustomizer 사용자 정의 파일은 다음 디렉토리에 있습니다.

- ▶ NT drive:\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_*instancename*.ear\properties
- ▶ 2000 drive:\Program Files\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_*instancename*.ear\properties
- ▶ AIX /usr/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_*instancename*.ear/properties
- ▶ Solaris /opt/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_*instancename*.ear/properties
- ▶ Linux /opt/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_*instancename*.ear/properties
- ▶ 400 /QIBM/UserData/WebASAdv4/WASinstancename/installedApps/WC\_Enterprise\_App\_*instancename*.ear/properties

### ▶ DB2

DBConnectionCustomizer 파일은 다음 ZIP 파일에 있습니다.

- ▶ NT drive:\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_*instancename*.ear\lib\loader\idresgen.zip

- **2000** drive:\Program Files\WebSphere\AppServer\installedApps\  
WC\_Enterprise\_App\_instancename.ear\lib\  
loader\idresgen.zip
- **AIX** /usr/WebSphere/AppServer/installedApps/  
WC\_Enterprise\_App\_instancename.ear/lib/  
loader/idresgen.zip
- **Solaris** /opt/WebSphere/AppServer/installedApps/  
WC\_Enterprise\_App\_instancename.ear/lib/  
loader/idresgen.zip
- **Linux** /opt/WebSphere/AppServer/installedApps/  
WC\_Enterprise\_App\_instancename.ear/lib/  
loader/idresgen.zip
- **400** /QIBM/UserData/WebASAdv4/WASinstancename/installedApps/  
WC\_Enterprise\_App\_instancename.ear/lib/  
loader/idresgen.zip

주: 사용자 정의 파일을 지정하려는 경우, 다음을 *instance\_name.xml* 파일의 DevTools 절에 추가해야 합니다.

- IDResolverCustomizerFile="*myIDResolverCustomizerFile*"

기본값으로, WebSphere Commerce 구성 파일, *instance\_name.xml*은 이 속성에 대해 값을 지정하지 않습니다.

ID 분석기는 *storenamemaster.xml* 파일 및 관련 DTD 파일 *wcs.dtd* 파일을 사용합니다. ID를 분석한 후, ID 분석기는 고유한 식별자를 포함하는 다음 *storenametime\_stampmaster.xml* 파일을 작성합니다. ID 분석 프로세스 동안 오류가 발생하는 경우, 로더 패키지는 *error.xml* 파일 및 *storenamemaster.error.xml* 파일을 작성합니다.

주: 공개 처리는 공개 시도 실패시 자동으로 이 임시 파일을 저장합니다. 그러나, 공개가 성공적이면 이 파일이 기본적으로 삭제됩니다. 문제점 해결을 목적으로 또는 WebSphere 테스트 환경에서 상점을 공개 및 작업할 목적으로 이 파일을 보존하기를 원할 수 있습니다. 임시 파일을 저장하려면 WebSphere Commerce 온라인 도움말의 "상점 서비스"를 참조하십시오.

*storenametime\_stampmaster.xml* 및 *storenamemaster.error.xml* 파일이 다음 디렉토리에 있습니다.

- **NT** drive:\WebSphere\CommerceServer\temp\*instancename*\tools\  
\devtools

- **2000** drive:\Program Files\WebSphere\CommerceServer\temp  
\*instancename*\tools\devtools
- **AIX** /usr/WebSphere/CommerceServer/temp/*instancename*/tools/  
devtools
- **Solaris** /opt/WebSphere/CommerceServer/temp/*instancename*/tools/  
devtools
- **Linux** /opt/WebSphere/CommerceServer/temp/*instancename*/tools/  
devtools
- **400** /QIBM/UserData/WebCommerce/instances/*instancename*/  
temp/tools/devtools

**로더 패키지를 호출하여 분석된 마스터 XML 파일을 데이터베이스로 로드:** 로더 패키지는 분석된 *storenametime\_stampmaster.xml* 파일을 데이터베이스로 로드합니다. 로드 프로세스 동안 오류가 발생하는 경우, 로더 패키지는 *error.xml* 및 *storenametime\_stamp master.error.xml* 파일을 작성합니다.

로더 패키지에 대한 자세한 정보는 265 페이지의 제 27 장 『상점 데이터 로드 개요』를 참조하십시오.

상점 서비스 또는 명령행 공개가 로더 패키지를 호출할 때, 사용하려는 Loader 메소드를 지정해야 합니다. 상점 서비스는 다음 로더 방법을 사용할 수 있습니다.

- SQL Import
- Import
- Load

주: 기본값으로, 상점 서비스를 SQL Import 메소드를 사용합니다.

WebSphere Commerce 구성 파일, *instance\_name.xml*에서 호출되는 상점 서비스 또는 명령행 공개를 지정할 수 있습니다.

- SQL Import: 이 메소드는 JDBC(Java Database Connectivity)를 사용하여 데이터를 삽입 및 갱신하고 가장 유연한 조작 방법을 제공하는 대신 대량의 데이터를 소량의 테이블로 반입하는 경우 가장 느린 방법을 제공합니다. 이는 컬럼 레벨 갱신을 허용합니다. SQL Import 사용을 권장합니다.

주: SQL Import 메소드는 데이터가 올바르지 않은 경우 데이터베이스를 손상시키지 않으므로 사용하기에 가장 안전한 방법입니다. SQL Import을 사용하여 로드하기 전에 레코드가 데이터베이스 스키마 제한자를 갖추어야 합니다. 기타 로더 방법은 많은 점검 없이 데이터베이스로 대량 로드되기 때문에 좀 더 빠릅니다. 결과적으로, 기타 메소드를 사용하기 전에 데이터가 정확성을 확인해야 합니다.

- **Import:** 이 메소드는 DB2® 원시 반입 기능을 사용하고 셀 레벨을 중간 속도 및 유연성으로 갱신합니다. 이 메소드는 Oracle에서는 사용 가능하지 않습니다.
- **Load:** 이 메소드는 RDBMS(DB2 로드 또는 SQLLoad)의 원시 기능을 사용하고 대량의 데이터를 소량의 테이블로 로드하기 위한 가장 빠른 메소드입니다.

Load 명령 메소드에 대한 자세한 정보는 278 페이지의 『load 명령』을 참조하십시오.

상점 서비스 및 명령행 공개 또한 로더에 사용될 사용자 정의 파일을 지정해야 합니다. WebSphere Commerce 구성 파일 *instance\_name.xml*의 사용자 정의 파일을 지정하지 않는 경우, 공개 코드는 기본 사용자 정의 파일 중 하나를 사용합니다 (MassLoadCustomizer).

주: 사용자 정의 파일을 지정하려는 경우, 다음을 *instance\_name.xml* 파일의 DevTools 절로 추가해야 합니다.

- `LoaderCustomizerFile="myLoaderCustomizerFile"`

기본값으로, WebSphere Commerce 구성 파일, *instance\_name.xml*이 이 속성에 대해 값을 지정하지 않습니다.

### parameters.jsp 파일 작성

공개 처리는 parameters.jsp 파일을 작성합니다. 이 파일은 storeId, catalogId 및 langId를 포함합니다. 동일한 견본 상점의 index.jsp 파일은 상점을 실행하기 위해 이 매개변수들을 사용합니다.

parameters.jsp 파일을 작성하기 위해 공개 처리를 하는 경우, 카탈로그 데이터 및 웹 자원이 적어도 한번 동시에 공개되어야 합니다. 이 두 항목이 모두 공개되지 않은 경우, 상점 서비스가 상점을 실행할 수 없습니다.

parameters.jsp 파일은 다음 디렉토리에 있습니다.

- **NT** drive:\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_instancename.ear\wcstores.war\storedir\include
- **2000** drive:\Program Files\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_instancename.ear\wcstores.war\storedir\include
- **AIX** /usr/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_instancename.ear/wcstores.war/storedir/include
- **Solaris** /opt/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_instancename.ear/wcstores.war/storedir/include
- **Linux** /opt/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_instancename.ear/wcstores.war/storedir/include

- `400 /QIBM/UserData/WebASAdv4/WASinstancename/installedApps/WC_Enterprise_App_instancename.ear/wcstores.war/storedir/include`

## 상점 구성 파일 압축 해제

**Business** ToolTech 및 NewFashion 견본 상점 아카이브는 다음 파일도 포함합니다.

- tools\_properties.zip
- tools\_xml.zip
- runtime\_xml.zip

이 파일은 sarinfo.xml 파일에 등록되고 상점을 구성하기 위해 상점 서비스에 의해 사용됩니다. 공개 프로세스 동안, 이 파일은 압축 해제되고 WebSphere Commerce 구성 파일 *instance\_name.xml*에서 지정된 디렉토리로 복사됩니다. 추가 정보는 328 페이지의 『WebSphere Commerce 서버에 복사하여 상점 입구 자원 및 상점 구성 파일 공개』를 참조하십시오.

주: 이 파일들은 변경, 제거 다른 상점으로 복사 또는 다른 디렉토리에 저장될 수 없습니다.

## 명령을 호출하여 비즈니스 계정 및 장기 구매 계약

일부 상점 데이터베이스 자원(장기 구매 계약 및 비즈니스 계정)을 로더 패키지가 로드할 수 없으며, 공개는 또한 관련 명령을 호출하여 이 자원을 WebSphere Commerce 서버로 공개합니다. 이 명령은 다음과 같습니다.

- AccountImport -- 상점 아카이브의 businessaccount.xml 파일에서 계정을 작성합니다.
- ContractImportApprovedVersion -- 상점 아카이브의 contract.xml 파일에서 장기 구매 계약을 반입합니다.
- ProductSetPublish -- 비즈니스 계정 및 장기 구매 계약이 작성되기 전에 카탈로그로 데이터베이스의 상품 세트 데이터를 동기화합니다. 상점 서비스 및 명령행 공개는 AccountImport 및 ContractImportApprovedVersion 명령을 호출하는 ProductSetPublish 명령을 호출합니다.

비즈니스 계정 및 장기 구매 계약 공개에 대한 자세한 정보는 323 페이지의 제 29 장 『비즈니스 계정 및 장기 구매 계약 공개』를 참조하십시오.

## 레지스트리 구성요소 갱신

공개 처리에서 마지막 조치는 레지스트리 구성요소를 갱신입니다. 공개는 WebSphere Commerce에서 모든 레지스트리를 공개합니다. 레지스트리에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오.



## 오류 조절

오류가 공개 프로세스의 자원 공개 단계에서 발생하는 경우, 오류 메시지를 공개 로그(『로그 파일 공개』를 참조) 또는 상점 서비스의 공개 요약 페이지에서 볼 수 있습니다.

## 지불 구성

공개 프로세스에서 마지막 단계는 지불 구성입니다. WebSphere Commerce는 IBM Payment Manager를 지원합니다. Payment Manager를 프로세스 지불의 방법으로써 사용하려는 경우, 지불 XML 파일을 131 페이지의 제 13 장 『지불 자원』에서 설명대로 작성해야 합니다. 지불 XML 파일이 공개 중인 상점 아카이브에 포함되는 경우, WebSphere Commerce는 공개되는 동안 다음 지불 구성을 완료합니다.

- 판매자 작성
- 계정 작성(오프라인 카세트에 대해서만)
- paymentinfo.xml에 지정된 브랜드 작성(오프라인 카세트에 대해서만)
- 사용자 권한 지정

## 오류 조절


공개 처리의 지불 구성 단계에서 오류가 발생하는 경우, 공개 로그(『로그 파일 공개』 참조)의 오류 메시지를 볼 수 있습니다.

## 로그 파일 공개

공개 프로세스의 자원 공개 단계에서 직면한 오류는 다음 로그 및 추적 파일에 작성됩니다.

- messages.txt: 공개 프로세스의 로더 패키지에서 오류 메시지를 포함합니다. 공개가 실패할 때 먼저 이 로그를 확인하십시오. 이 오류 메시지에서 언급된 행 또는 열 번호는 임시 master.xml, storenamemaster.xml 또는 storenametime\_stampmaster.xml 파일을 참조하십시오.
- trace.txt: 로더 패키지 및 공개 프로세스의 ID 분석기 부분에 대한 추적 정보를 포함합니다. 기본적으로, trace.txt 파일이 꺼져 있습니다.
- ecmsg\_instancename\_timestamp.log: WebSphere Commerce 서버에서 모든 실행하는 오류 메시지를 로그합니다.
- wcs.log file: WebSphere Application Server에서 실행하는 모든 응용프로그램(공개 포함)에서 표준 콘솔 출력값을 포함합니다.

로그 파일이 다음 디렉토리에 있습니다.

-  drive:\WebSphere\CommerceServer\instances\instancename\logs



- **2000** drive:\Program Files\WebSphere\CommerceServer  
   \*instancename*\logs
- **AIX** /usr/WebSphere/CommerceServer/instances/*instancename*/logs
- **Solaris** /opt/WebSphere/CommerceServer/instances/*instancename*/logs
- **Linux** /opt/WebSphere/CommerceServer/instances/*instancename*/logs
- **400** /QIBM/UserData/WebCommerce/instances/*instancename*/logs

trace.txt 및 messages.txt 로그 파일을 구성하려면(즉, 로그 레벨 또는 기타 옵션을 조정), 다음 파일을 편집하십시오.

- **NT** drive:\WebSphere\CommerceServer\xml\loader  
   \WCALoggerConfig.xml
- **2000** drive:\Program Files\WebSphere\CommerceServer\xml  
   \loader\WCALoggerConfig.xml
- **AIX** /usr/WebSphere/CommerceServer/xml/loader/WCALoggerConfig.  
   xml
- **Solaris** /opt/WebSphere/CommerceServer/xml/loader/WCALoggerConfig.  
   xml
- **Linux** /opt/WebSphere/CommerceServer/xml/loader/WCALoggerConfig.  
   xml
- **400** /QIBM/UserData/WebCommerce/instances/*instancename*/  
   xml/WCALoggerConfig.xml

주: WCALoggerConfig.xml 파일 구성에 대한 자세한 정보는 *WebSphere Commerce Catalog Manager* 사용자 안내서를 참조하십시오.



---

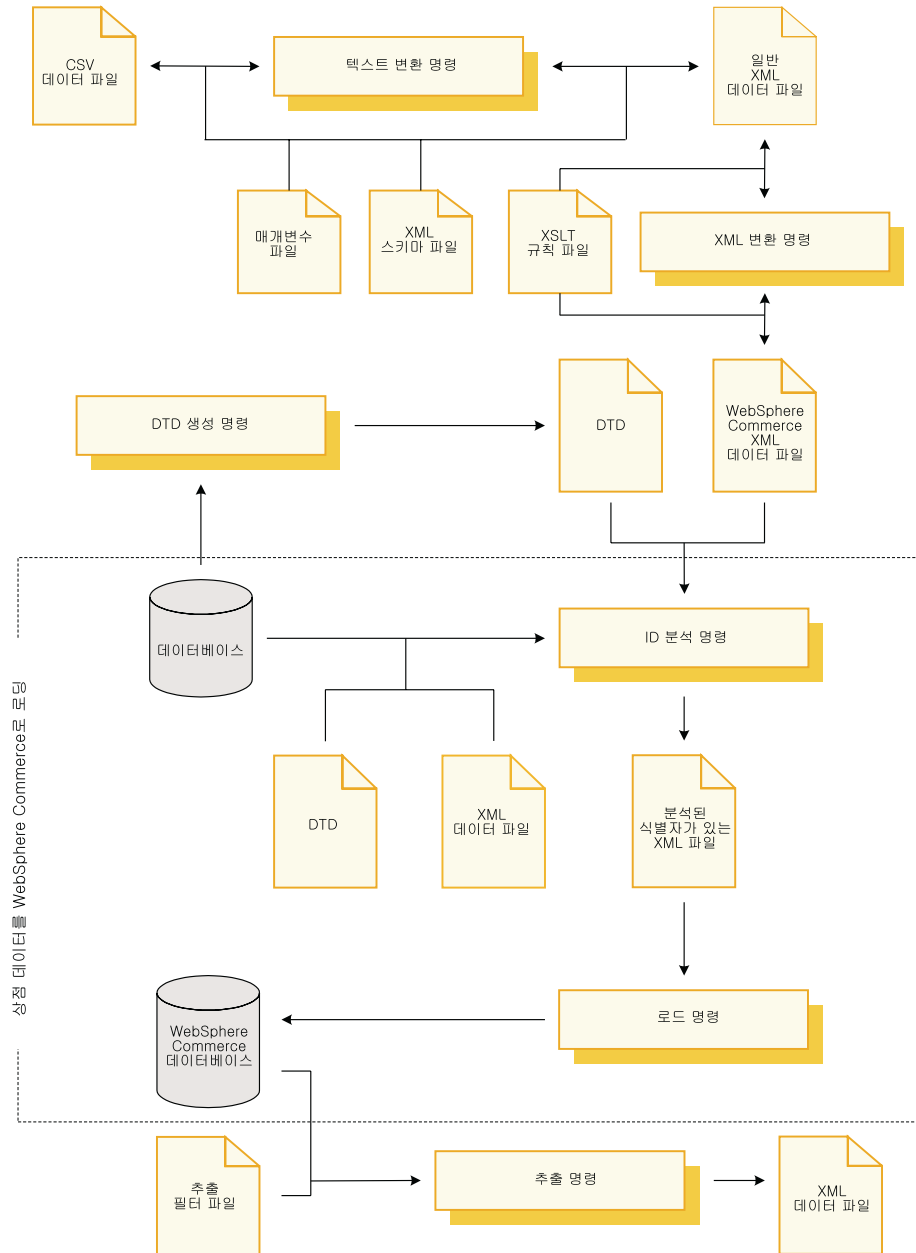
## 제 27 장 상점 데이터 로드 개요

상점 데이터를 작성한 후, 상점 아카이브로 패키지하도록 작성할 수 있으며, 상점 서비스를 사용하여 공개하거나 WebSphere Commerce Catalog Manager 로더 패키지를 사용하여 WebSphere Commerce 서버 데이터베이스로 직접 로드할 수 있습니다. WebSphere Commerce 데이터베이스 자원 그룹의 로드 프로세스에 대한 정보에 대해서는 307 페이지의 제 28 장 『WebSphere Commerce 데이터베이스 자원 그룹 로드』 및 316 페이지의 『데이터베이스 자원 그룹 로드』를 참조하십시오.

Catalog Manager에서는 여섯 개의 명령행 유틸리티("로더 패키지"를 참조) 및 로드를 위해 데이터를 준비하는 데 사용하고 데이터를 상점으로 로드할 수 있는 세 개의 관련 관리 도구를 제공합니다. 이 명령 및 도구는 XML(Extensible Markup Language) 데이터 파일을 사용하여 정보를 관리합니다.

## WebSphere Commerce의 데이터 로드 에 대한 이해

로더 패키지 명령을 사용하여 수행할 수 있는 데이터 준비, 로드 및 추출 프로세스가 다음 그림에 나타납니다.



점선은 WebSphere Commerce 서버 데이터베이스로 상점 데이터를 로드하는 데 가장 많이 사용되는 두 개의 프로세스를 표시합니다(식별자 해결 및 데이터 로드). 이 프로세스가 이 장의 핵심입니다.

WebSphere Commerce 서버 데이터베이스로 로드하기 위해 사용자 데이터를 준비하는 것에 대한 자세한 정보는 39 페이지의 제 4 부 『상점 데이터 개발』을 참조하십시오.

다음 두 로더 패키지 명령행 유틸리티는 WebSphere Commerce 서버 데이터베이스로 데이터를 로드하기 위해 일반적으로 사용됩니다.

- **ID Resolve** 명령

로더 패키지를 사용하여 WebSphere Commerce Server 데이터베이스로 XML 데이터를 로드하려면, XML 요소가 대상 WebSphere Commerce Server 데이터베이스의 스키마로 직접적으로 맵핑되어야 합니다. 데이터베이스 스키마의 고유 또는 1차 키와 일치하는 속성을 갖는 모든 XML 요소는 고유 식별자를 가지고 있어야 합니다. 그리고 데이터베이스 스키마의 널 사용이 불가능한 모든 열은 널이 아닌 값으로 정의된 대응 속성을 가져야 합니다. ID 분석기는 적절한 XML 요소의 고유 또는 1차 키 요소에 대해 고유 식별자를 생성할 수 있습니다.

주: 이 문서에서 보면 식별자는 각 행에 고유 식별자를 제공하는 데이터베이스 테이블의 단일 열에 있는 값입니다. 식별자를 생성하기 위해 ID 분석기를 사용하는 경우, 이것은 KEYS 또는 SUBKEYS 테이블의 기본 값을 확보하며 데이터베이스 테이블의 각 행에 대한 식별자를 분석하기 위해 순차적으로 값을 증가시킵니다.

이 명령의 정보에 대해서는 269 페이지의 『ID Resolve 명령』, 296 페이지의 『로더 패키지 명령 및 스크립트 사용』 및 297 페이지의 『식별자를 분석하는 예제』를 참조하십시오.

- **Load** 명령

로더는 데이터베이스로 데이터를 로드하기 위한 입력으로 올바르고 잘 형성된 XML 파일을 사용합니다. XML 문서의 구성요소는 데이터베이스에 있는 테이블 이름으로 맵핑하고, 구성 요소 속성은 열로 맵핑합니다.

주: 유효성 및 잘 구성된 제한자의 설명을 위해 W3C(World Wide Web Consortium) XML 지시사항을 참조하십시오.

이 명령의 정보에 대해서는 278 페이지의 『load 명령』, 296 페이지의 『로더 패키지 명령 및 스크립트 사용』 및 304 페이지의 『데이터 로드 예제』를 참조하십시오.

이 명령이 이 장의 주요 내용입니다.

또한, 다음 로더 패키지 명령행 유틸리티는 사용자 데이터를 사용하기 위해 사용됩니다.

- **DTD Generate** 명령

DTD 작성기는 XML 데이터가 로드되는 대상 데이터베이스의 테이블 및 열을 설명하는 DTD(Document Type Definition)를 생성합니다. 또한, DTD 작성기는 데이터베이스에 대한 XML 스키마를 생성할 수 있습니다.

DTD 작성기는 WebSphere Commerce 데이터베이스 스키마를 기반으로 DTD를 작성할 수 있습니다. 견본 상점 아카이브가 제공된 DTD를 사용하고 데이터베이스 스키마를 수정하지 않는 경우, 일반적으로 DTD 작성기를 사용하여 DTD를 생성할 필요가 없습니다.

자세한 정보는 286 페이지의 『DTD Generate 명령』을 참조하십시오.

- **Extract 명령**

추출기는 데이터베이스에 대한 조회를 사용하여 데이터베이스에서 데이터의 선택된 서브셋을 XML 문서로 추출합니다.

이 명령을 사용하여 데이터베이스에서 XML 포맷으로 추출할 수 있습니다.

자세한 정보는 289 페이지의 『Extract 명령』을 참조하십시오.

- **Text Transform 명령**

텍스트 변환기는 문자 분리 변수 포맷 및 XML 데이터 포맷 사이에 데이터를 변환합니다.

예를 들어, 데이터를 바로 데이터베이스에서 XML 포맷으로 추출할 수 없는 경우, 데이터를 문자 분리된 변수 포맷으로 저장한 후 이 명령을 사용하여 XML 포맷으로 변환하십시오.

자세한 정보는 292 페이지의 『Text Transform 명령』을 참조하십시오.

- **XML Transform 명령**

XML 변환기는 XML 문서의 데이터를 대체 XML 포맷으로 변환합니다. 이는 XSL(Extensible Stylesheet Language)을 사용하여 변환에 대한 맵핑 규칙을 정의합니다.

이 명령을 사용하여 데이터를 로드하려는 대상 WebSphere Commerce 데이터베이스의 스키마로 직접 맵핑하는 포맷으로 XML 데이터를 변환하십시오.

자세한 정보는 293 페이지의 『XML Transform 명령』을 참조하십시오.

이 명령은 이 장의 주용 내용이 아닙니다. 이 도구에 대한 자세한 정보는 *IBM WebSphere Commerce 5.4 Catalog Manager 사용자 안내서*의 최신 버전을 참조하십시오.

또한, WebSphere Commerce Catalog Manager는 데이터 관리 기능의 관리에서 보조하는 다음 도구를 포함합니다.

- **텍스트 변환 도구**

텍스트 변환 도구는 텍스트 변환 명령을 사용하여 문자 분리 변수 포맷 및 XML 데이터 포맷 사이의 데이터 변환을 처리하도록 도와줍니다.

- **XSL 편집기**

XSL 편집기에서는 XML 변환기가 사용할 수 있는 XSL 파일을 편집하기 위한 비주얼 인터페이스를 제공합니다. XSL 편집기를 사용하여 소스 DTD의 요소에서 대상 DTD의 요소로 연관을 XML 포맷 사이의 데이터를 변환하기 위한 맵핑 규칙을 정의할 때 설정할 수 있습니다.

- 웹 편집기

웹 편집기를 사용하여 웹 브라우저에서 데이터베이스의 데이터를 작성, 수정 및 삭제할 수 있습니다.

이 도구는 이 장의 주요 내용이 아닙니다. 이 도구에 대한 자세한 정보는 *IBM WebSphere Commerce 5.4 Catalog Manager 사용자 안내서*의 최신 버전을 참조하십시오.

## 상점 데이터를 로드하기 위한 로더 패키지 명령

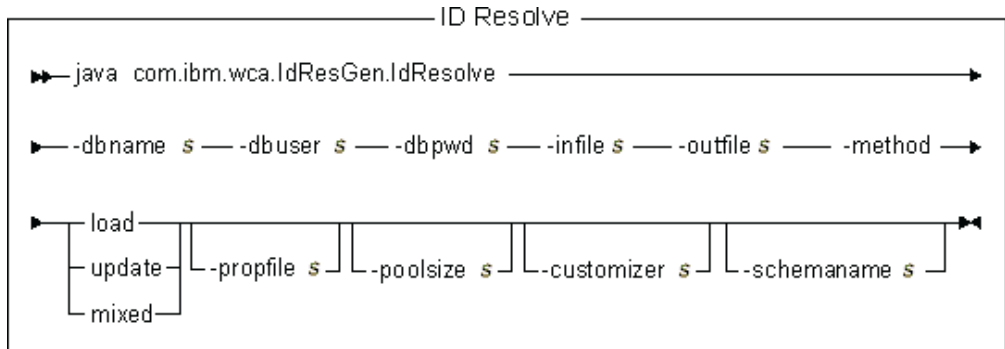
### ID Resolve 명령

이 명령은 데이터베이스로 요소를 로드하기 전에, 식별자를 필요로 하는 XML 데이터 요소에 대한 고유 식별자를 생성합니다. 소스 XML 데이터가 이미 필수 고유 식별자를 제공하는 경우, ID 분석기를 실행할 필요가 없습니다.

WebSphere Commerce 데이터베이스 스키마가 테이블 사이의 다양한 관계를 표시하는 데 사용된 테이블 사이의 기본 및 foreign key를 정의합니다. 이러한 이유로, WebSphere Commerce XML 요소는 고유 식별자와 일치하는 속성을 포함해야 합니다. WebSphere Commerce 서버 데이터베이스에서, 분석되어야 하는 테이블의 식별자는 KEYS 및 SUBKEYS 테이블에 정의된 테이블 식별자이어야 합니다. 이 테이블은 WebSphere Commerce에서 기본 테이블이라 명명됩니다. KEYS 및 SUBKEYS 테이블에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오.

주: KEYS나 SUBKEYS에 정의되지 않은 테이블에 대한 식별자를 분석하는 것이 필요한 경우, ID 분석기를 실행하기 전에 SUBKEYS 테이블에 해당 테이블을 추가하십시오.

WebSphere Commerce XML 요소 및 속성은 데이터베이스 및 데이터베이스 인스턴스에서 휴대용으로 사용됩니다. 식별자는 보통 내부 별명을 사용하여 표시됩니다. 데이터가 WebSphere Commerce 서버 데이터베이스로 로드되기 전에, 이 별명은 올바른 수치 식별자로 분석되어야 합니다. 자세한 정보는 361 페이지의 부록 B 『데이터 작성』을 참조하십시오.



주:

1. 위 도표는 명령 매개변수에 대한 참조용입니다. 이 명령에 대해 제공되고 296 페이지의 『로더 패키지 명령 및 스크립트 사용』 아래 나열된 명령 파일 또는 스크립트는 실제 Java 명령에 래퍼로써 수행되고 동일한 매개변수를 승인합니다. 따라서 사용자가 명령 파일 또는 스크립트를 사용하는 것이 Java 명령을 직접 호출하는 것보다 권장됩니다.
2. 이 명령에 대한 매개변수로서 지정된 파일 이름은 상대 또는 절대 경로가 앞에 올 수 있습니다.

매개변수 값:

**-dbname**

대상 데이터베이스 이름

**-dbuser**

데이터베이스에 연결 중인 사용자의 이름

**-dbpwd**

데이터베이스에 연결 중인 사용자의 암호

**-infile** 테이블 레코드를 포함하는 입력 XML 문서 이름

**-outfile**

생성되는 출력 XML 파일의 이름으로, 이 파일은 로더에 대한 입력으로 사용될 수 있습니다.

**-method**

입력 파일 처리에 사용되는 메소드

- 파일의 모든 레코드가 데이터베이스에 없는 경우, load 메소드를 사용하여 입력 파일을 처리하십시오.
- 파일의 모든 레코드가 데이터베이스에 있는 경우, update 메소드를 사용하여 입력 파일을 처리하십시오.



- 파일의 일부 레코드가 데이터베이스에 있는 경우, mixed 메소드를 사용하여 입력 파일을 처리하십시오.

기본 메소드는 로드입니다.

#### **-propfile**

이름=값 쌍의 양식으로 Java 특성이 들어 있는 텍스트 파일. 이 특성 파일은 ID 분석기가 식별자를 분석하는 방법을 설정합니다. 이것은 어떤 기본 항목의 열이 기본 행의 식별자가 필요한 테이블을 찾는 데 사용되어야 하는지를 설명 하는데 사용됩니다. 이 파일은 foreign key 식별자 찾아보기에 대한 열 이름을 정의하며 기본 테이블(CATEGORY 및 PRODUCT) 조회의 술부를 선택합니다. 식별자를 포함하지 않는 고유 색인을 정의한 테이블에 대해 이 파일의 항목을 생략할 수 있습니다. 이 매개변수는 선택적입니다.

IdResolveKeys.properties는 기본 파일입니다. 이 특성 파일은 다음 예 중 하나처럼 지정될 수 있습니다.

```
-propfile d:\WebSphere\CommerceServer\prop\idresprop.properties
-propfile d:\WebSphere\CommerceServer\prop\idresprop
```

이 파일이 현재 디렉토리에 있는 경우, 동일한 파일이 다음 예처럼 지정될 수 있습니다.

```
-propfile idresprop.properties
```

클래스 경로 시스템 환경 변수에 지정된 디렉토리에 파일이 존재하면, 동일한 파일이 다음 예처럼 지정될 수 있습니다.

```
-propfile idresprop
```

ID 분석기로 새 특성 파일 작성 및 지정에 대한 자세한 정보는 *IBM WebSphere Commerce 5.4 Catalog Manager 사용자 안내서*의 최신 버전을 참조하십시오.

#### **-poolsize**

예약될 식별자의 수. 이 매개변수는 선택적입니다. 기본 수는 50입니다.

#### **-customizer**

사용될 조정자 특성 파일의 이름. 이 매개변수는 선택적입니다. 사용자 정의 특성 파일은 ID 분석기가 기능하는 방법을 설정합니다.

DB2ConnectionCustomizer.properties 파일이 기본 파일입니다. 사용자 정의 프로그램 특성 파일은 다음 예 중 하나처럼 지정될 수 있습니다.

```
-customizer d:\WebSphere\CommerceServer\prop\idres.properties
-customizer d:\WebSphere\CommerceServer\prop\idres
```

이 파일이 현재 디렉토리에 있는 경우, 일부 파일은 다음 예처럼 지정될 수 있습니다.

-customizer idres.properties

클래스 경로 시스템 환경 변수에 지정된 디렉토리에 파일이 존재하면, 동일한 파일이 다음 예처럼 지정될 수 있습니다.

-customizer idres

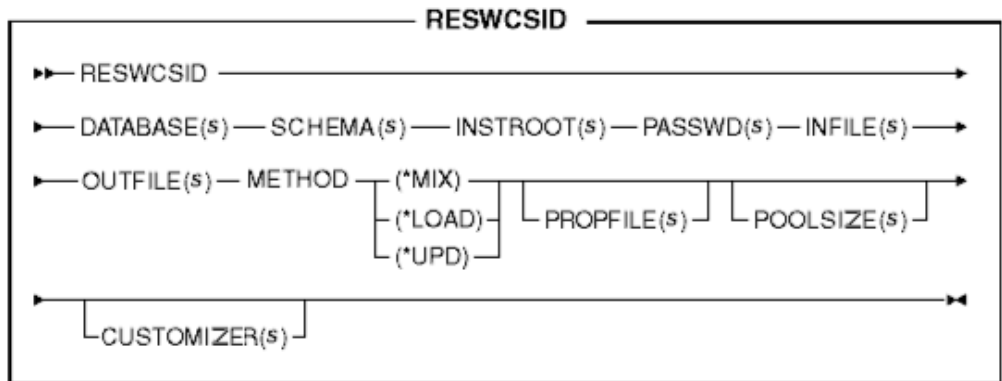
새 사용자 정의 특성 파일 작성 및 지정에 대한 자세한 정보는 *IBM WebSphere Commerce 5.4 Catalog Manager 사용자 안내서*의 최신 버전을 참조하십시오.

**-schemaname**

대상 데이터베이스 스키마의 이름. 이 매개변수는 선택적입니다.

명령을 실행할 때 이 매개변수가 지정되지 않은 경우, 이 명령은 SchemaName의 값을 지정하는 사용자 정의 특성 파일에서 이름=값 쌍을 찾습니다. 이 쌍이 이 특성 파일에 존재할 경우 명령은 지정된 값을 사용합니다. 명령행 또는 특성 파일 스펙 모두에 이 매개변수가 존재하지 않는 경우, 이 명령은 데이터베이스 테이블의 스키마 이름을 기본값으로 합니다.

▶ 400



주: 이 명령에 대한 매개변수로서 지정된 파일 이름은 상대 또는 절대 경로가 앞에 올 수 있습니다.

**매개변수 값:**

**DATABASE**

관계형 데이터베이스 디렉토리에 표시된 대로의 대상 데이터베이스의 이름

**SCHEMA**

대상 데이터베이스 스키마의 이름으로, 인스턴스 이름과 같습니다.

**INSTROOT**

/QIBM/UserData/WebCommerce/instances/instance\_name과 같은 WebSphere Commerce 인스턴스 루트 경로의 전체 이름

**PASSWD**

WebSphere Commerce 인스턴스의 암호

**INFILE**

테이블 레코드를 포함하는 입력 XML 문서 이름

**OUTFILE**

생성되는 출력 XML 파일의 이름으로, 이 파일은 로더에 대한 입력으로 사용될 수 있습니다.

**METHOD**

입력 파일 처리에 사용되는 메소드.

- 파일의 모든 레코드가 데이터베이스에 없는 경우, Load 메소드(\*LOAD)를 사용하여 입력 파일을 처리하십시오.
- 파일의 모든 레코드가 데이터베이스에 있는 경우, update 메소드(\*UPD)를 사용하여 입력 파일을 처리하십시오.
- 파일의 일부 레코드가 데이터베이스에 있는 경우, mixed 메소드(\*MIX)를 사용하여 입력 파일을 처리하십시오.

**PROFILE**

이름=값 쌍의 양식으로 Java 특성이 들어 있는 텍스트 파일. 이 특성 파일은 ID 분석기가 식별자를 분석하는 방법을 설정합니다. 이것은 어떤 기본 항목의 열이 기본 행의 식별자가 필요한 테이블을 찾는 데 사용되어야 하는지를 설명하는데 사용됩니다. 이 파일은 foreign key 식별자 찾아보기에 대한 열 이름을 정의하며 기본 테이블(CATEGORY 및 PRODUCT) 조회의 술부를 선택합니다. 식별자를 포함하지 않는 고유 색인을 정의한 테이블에 대해 이 파일의 항목을 생략할 수 있습니다. 이 매개변수는 선택적입니다.

IdResolveKeys.properties는 기본 파일입니다. 이 특성 파일은 다음 예 중 하나처럼 지정될 수 있습니다.

```
PROFFILE(/wc/prop/idresprop.properties)
```

```
PROFFILE(/wc/prop/idresprop)
```

이 파일이 현재 디렉토리에 있는 경우, 동일한 파일이 다음 예처럼 지정될 수 있습니다.

```
PROFFILE(idresprop.properties)
```

클래스 경로 시스템 환경 변수에 지정된 디렉토리에 파일이 존재하면, 동일한 파일이 다음 예처럼 지정될 수 있습니다.

```
PROFFILE(idresprop)
```

ID 분석기로 새 특성 파일 작성 및 지정에 대한 자세한 정보는 *IBM WebSphere Commerce 5.4 Catalog Manager 사용자 안내서*의 최신 버전을 참조하십시오.

## POOLSIZE

예약될 식별자의 수. 이 매개변수는 선택적입니다. 기본 수는 50입니다.

## CUSTOMIZER

사용될 조정자 특성 파일의 이름. 이 매개변수는 선택적입니다. 사용자 정의 특성 파일은 ID 분석기가 기능하는 방법을 설정합니다. 기본 파일은 ISeries\_RESWCSID\_Customizer.properties입니다. 사용자 정의 프로그램 특성 파일은 다음 예 중 하나처럼 지정될 수 있습니다.

```
CUSTOMIZER(/wc/prop/idres.properties)
```

```
CUSTOMIZER(/wc/prop/idres)
```

이 파일이 현재 디렉토리에 있는 경우, 일부 파일은 다음 예처럼 지정될 수 있습니다.

```
CUSTOMIZER(idres.properties)
```

클래스 경로 시스템 환경 변수에 지정된 디렉토리에 파일이 존재하면, 동일한 파일이 다음 예처럼 지정될 수 있습니다.

```
CUSTOMIZER(idres)
```

새 사용자 정의 특성 파일 작성 및 지정에 대한 자세한 정보는 *IBM WebSphere Commerce 5.4 Catalog Manager 사용자 안내서*의 최신 버전을 참조하십시오.

## 분석 기술:

ID 분석기는 특성 파일을 사용 여부와 관계없이 다음 기술의 두 개 또는 세 개의 조합을 사용하여 식별자를 분석합니다.

- 내부 별명 분석

내부 별명 ID 분석을 사용할 때, 별명이 소스 XML 문서의 고유 키(식별자) 대신 대체됩니다. 그런 다음 이 별명은 XML 파일의 다른 곳에서 사용되어 해당 요소를 참조합니다.

내부 별명은 XML 파일을 통해 연속적으로 사용되어야 합니다. 예를 들어, 주소록 ID, ADDRBOOK\_ID의 별명이 @addrbook\_1인 경우, 파일에서 해당 ID에 대한 모든 foreign-key는 @addrbook\_1을 사용해야 합니다.

특정 XML 파일에 대한 별명은 일시적임에 주의하십시오. 별명은 저장되지 않으며 다시 별명을 도입하지 않고는 별도의 XML 파일에서 사용될 수 없습니다. 상점 서비스에서 공개하는 동안 공개는 XML 파일을 연결하므로 모든 데이터에서 분석이 발생할 수 있습니다.

- 고유 색인 분석

또한, ID 분석기는 데이터베이스 스키마를 분석하여 요구사항을 실행하는 고유 색인의 존재 여부를 판별합니다. 분석 중인 테이블에 대한 특성 파일의 항목이 없거나 특성 파일이 없을 때에만, ID 분석기는 고유 색인을 찾습니다. 이 조건이 true인 경우, 고유 색인 확인이 수행됩니다. 고유 색인이 존재하고 테이블에 대한 기본 키를 포함하는 경우, 고유 색인이 올바른 것으로 간주됩니다.

- 특성 파일 스펙

ID 분석기를 통해 대체 Java 특성 파일을 사용하여 기본 행의 식별자를 필요로 하는 테이블에 대한 찾아보기로써 사용되어 하는 기본 항목의 열을 판별합니다.

WebSphere Commerce와 함께 제공된 견본 상점 아카이브는 XML 파일에서 내부 별명을 사용합니다. 이것은 상점 아카이브가 데이터베이스에서 휴대 가능하도록 합니다. 고유 색인 및 특성 파일 스펙 기술 또한 데이터베이스에서 휴대성을 허용하고 사용자가 항상 고유 열을 변경할 수 있지만, 이 기술이 ID 해상도에 대해 이후에 사용된 경우 문제점의 원인이 됩니다. 예를 들어, 사용자가 고유 열을 변경할 경우, 열 이름은 특성 파일 정의에서 변경되어야 합니다. 그러나, 내부 별명 기술을 사용하면 데이터베이스 변경시 XML 또는 특성 파일을 반드시 변경해야 할 필요는 없습니다. 상점 서비스에서 공개 또는 로더 패키지를 사용하는 동안 ID 분석기는 별명을 고유한 값으로 교체됩니다. 일단, 데이터가 로드되면, 별명이 사용자에게 명료해집니다. 자세한 정보는 361 페이지의 부록 B 『데이터 작성』을 참조하십시오.

ID 분석기는 다음 프로세스를 사용합니다.

- 입력 XML 데이터가 이미 하드 코딩된 식별자(예: "12345")가 있는 기본 테이블에서 요소를 가지고 있는 경우, ID 분석기는 해당 요소에 대한 새 식별자를 생성하지 못합니다.
- 입력 XML 데이터가 식별자가 없는 기본 테이블에서 요소를 가지고 있는 경우, ID 분석기는 해당 요소에 대해 열이 있는지 여부를 보기 위해 데이터베이스를 찾습니다.

데이터베이스의 요소를 조사하려면 사용된 요소의 다른 열이 고유한 키를 형성하기 위해 사용되어야 합니다. 다른 열이 특성 파일에 지정될 수 있습니다. 또는 ID 분석기는 사용할 열을 결정하기 위해 허용될 수 있습니다.

- 특성 파일이 사용되는 경우, 분석 중인 파일에 대한 특성 파일의 항목인 경우, ID 분석기는 특성 파일에서 지정된 열을 사용하여 고유한 키를 형성합니다.
- 사용 중인 특성 파일이 없거나 분석 중인 테이블에 대한 특성 파일에 항목이 없는 경우, ID 분석기는 고유 색인 분석을 사용합니다.

고유 색인 분석은 식별자 찾는 수단으로써 테이블의 지정된 모든 고유 색인을 사용합니다. 예를 들어, MEMBER\_ID 플러스 IDENTIFIER가 CATALOG 테이블의 고유 색인이고 CATALOGDSC 테이블의 foreign key에 분석 포인트로써 사용될 수 있습니다.

동일한 고유 키에 행이 있는 경우, 요소는 데이터베이스에 이미 존재한다고 간주합니다. 그렇지 않은 경우, 데이터의 새 가격으로 보입니다.

- 요소가 데이터베이스의 행으로 이미 존재하는 경우, 식별자는 검토되고 나중에 사용하기 위해 저장됩니다. 그렇지 않은 경우, 새 식별자는 KEYS 또는 SUBKEYS 테이블의 사용 가능한 값을 사용하여 ID 분석기에 의해 작성됩니다.
- XML 문서에서 요소에 대한 내부 별명(예를 들어, "@store\_id\_1")을 지정하는 경우, 그 별명은 나중에 동일한 내부 별명을 사용하여 조사하기 위해 식별자와 관련되어 있습니다.
- 기본 테이블에서의 요소를 참조할 필요가 있는 다음 XML 문서 요소는 기본 요소를 하나 가지고 있을 경우(예를 들어, "@store\_id\_1"), 내부 별명을 사용하고 그렇지 않은 경우(예를 들어, "@WC2001@100"), 찾아보기 열의 값을 사용합니다. 다른 경우, 지정된 값은 실제 식별자를 조사하기 위해 사용되고 그 식별자로 교체됩니다.
- 출력 XML 문서가 작성될 때, 모든 1차 테이블 요소는 실제 식별자를 가지고 있고 1차 테이블 요소를 참조하는 모든 요소는 실제 식별자를 사용하여 참조합니다. 내부 별명 또는 위에서 언급된 찾아보기 열 값도 아닙니다. 이는 완전히 분석된 XML 문서입니다.

#### **ID Resolve 명령에 대한 메소드:**

ID Resolve 명령은 입력 파일을 처리하는 load, update, 또는 mixed 메소드를 선택하게 합니다.

#### **Load 메소드:**

ID 분석기에 대한 load 메소드는 데이터베이스로 로드되는 모든 새 레코드에 대한 새 식별자를 생성하기 위해 사용됩니다.

주: ID 분석기에 대한 load 메소드를 지정하는 경우, 입력 파일의 레코드는 데이터베이스에 없어야 합니다. load 메소드가 ID 분석으로 사용되고 소스 XML 파일의 레코드가 이미 대상 데이터베이스에 있는 경우, 로더는 데이터를 로드할 때 오류를 생

성합니다. ID 분석기는 ID 분석하는 동안 새 1차 키를 XML 파일의 레코드에 지정하지만, 데이터베이스로 데이터를 로드할 때, 오류가 생성됩니다. 로더는 복제 작성 레코드를 처리하는 지점에서 중지되지 않지만, 이는 오류를 보고하며 복제 작성 레코드는 데이터베이스로 로드되지 않습니다.

다음 예제는 데이터베이스에서 새로운 데이터 요소에 대한 식별자를 생성하기 위해 사용됩니다.

- ▶ Windows

```
idresgen -dbname db -dbuser user -dbpwd pwd -infile input.xml
-outfile output.xml -method load -customizer customizer -schemaname wcsadmin
```

- ▶ AIX ▶ Solaris ▶ Linux

```
./idresgen.sh -dbname db -dbuser user -dbpwd pwd -infile input.xml
-outfile output.xml -method load -customizer customizer -schemaname wcsadmin
```

- ▶ 400

```
QWEBCOMM/RESWCSID DATABASE(DATABASE_NAME) SCHEMA(WCSADMIN)
INSTROOT(/QIBM/UserData/WebCommerce/instances/mser)
PASSWD(mypassword) INFILE(input.xml) OUTFILE(output.xml)
METHOD(*LOAD)
```

주: 적절한 ID Resolve 명령 또는 스크립트의 위치는 296 페이지의 『로더 패키지 명령 및 스크립트 사용』을 참조하십시오.

**Update 메소드:** ID 분석기에 대한 update 메소드를 지정하는 경우, 입력 파일의 레코드는 데이터베이스에 이미 있어야 합니다. ID 분석기는 275 페이지에서 설명된 것과 같이 데이터베이스에 식별자를 위치 지정합니다. 레코드가 데이터베이스에 없는 경우, ID 분석기가 이 레코드에 대해 분석기를 분석할 수 없고 오류가 발생함을 표시합니다. 다음 예제는 데이터베이스에 이미 있는 데이터 요소에 대한 식별자를 찾기 위해 사용됩니다.

- ▶ Windows

```
idresgen -dbname db -dbuser user -dbpwd pwd -infile input.xml
-outfile output.xml -method update -customizer customizer -schemaname wcsadmin
```

- ▶ AIX ▶ Solaris ▶ Linux

```
./idresgen.sh -dbname db -dbuser user -dbpwd pwd -infile input.xml
-outfile output.xml -method update -customizer customizer -schemaname wcsadmin
```

- ▶ 400

```
QWEBCOMM/RESWCSID DATABASE(DATABASE_NAME) SCHEMA(WCSADMIN)
INSTROOT(/QIBM/UserData/WebCommerce/instances/mser) PASSWD(mypassword)
INFILE(input.xml) OUTFILE(output.xml) METHOD(*UPD)
```

주: 적절한 ID 분석 명령 또는 스크립트의 위치는 296 페이지의 『로더 패키지 명령 및 스크립트 사용』을 참조하십시오.

### **mixed 메소드:**

입력 데이터 파일은 데이터베이스에 이미 있는 레코드 뿐만 아니라 새로운 일부 레코드를 포함하는 경우, ID 분석기는 mixed 메소드를 사용하여 실행되어야 합니다. 이 메소드로 레코드가 데이터베이스에 없는 경우에만, ID 분석기는 레코드에 대한 새 식별자를 작성합니다. 그렇지 않은 경우, 기존 식별자를 데이터베이스에서 얻을 수 있습니다. 다음 예제는 데이터베이스에 이미 있는 데이터 요소에 대한 식별자를 생성하기 위해 사용됩니다.

- ▶ Windows

```
idresgen -dbname db -dbuser user -dbpwd pwd -infile input.xml
-outfile output.xml -method mixed -customizer customizer -schemaname wcsadmin
```

- ▶ AIX ▶ Solaris ▶ Linux

```
./idresgen.sh -dbname db -dbuser user -dbpwd pwd -infile input.xml
-outfile output.xml -method mixed -customizer customizer -schemaname wcsadmin
```

- ▶ 400

```
QWEBCOMM/RESWCSID DATABASE(DATABASE_NAME) SCHEMA(WCSADMIN)
INSTROOT(/QIBM/UserData/WebCommerce/instances/mser) PASSWD(mypassword)
INFILE(input.xml) OUTFILE(output.xml) METHOD(*MIX)
```

### **주:**

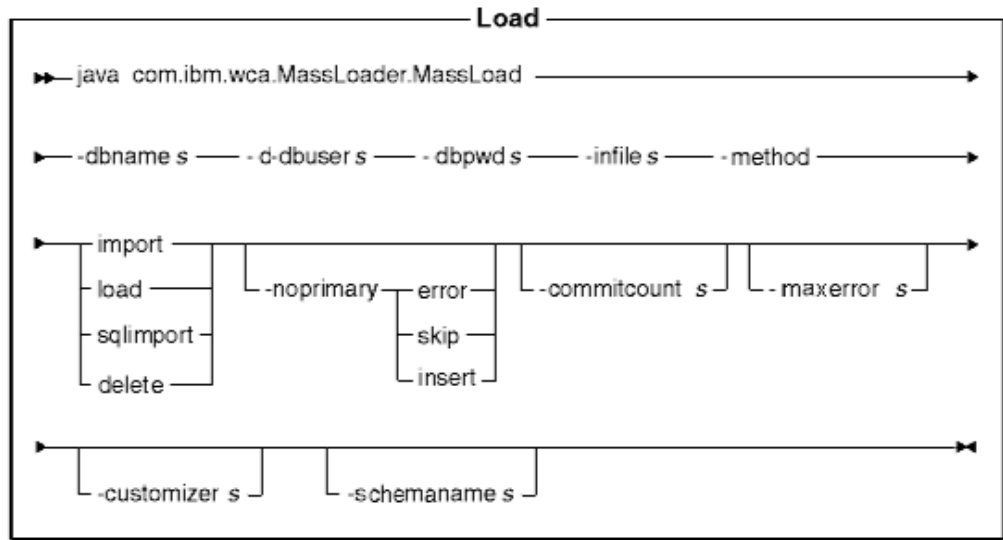
1. 적절한 ID 분석 명령 또는 스크립트의 위치는 296 페이지의 『로더 패키지 명령 및 스크립트 사용』을 참조하십시오.
2. mixed 메소드는 상점 서비스에 대해 권장되는 메소드입니다.

이 명령을 실행하기 위해 사용된 파일을 설치하고 사용자 정의할 수 있는 자세한 정보에 대해서는 *IBM WebSphere Commerce 5.4 Catalog Manager 사용자 안내서*의 최신 버전을 참조하십시오.

### **load 명령**

이 명령은 XML 입력 파일을 대상 데이터베이스로 로드합니다.





주:

1. 위 도표는 명령 매개변수에 대한 참조용입니다. 이 명령에 대해 제공되고 296 페이지의 『로더 패키지 명령 및 스크립트 사용』 아래 나열된 명령 파일 또는 스크립트는 실제 Java 명령에 래퍼로써 수행되고 동일한 매개변수를 승인합니다. 따라서 사용자가 명령 파일 또는 스크립트를 사용하는 것이 Java 명령을 직접 호출하는 것보다 권장됩니다.
2. 이 명령에 대한 매개변수로서 지정된 파일 이름은 상대 또는 절대 경로가 앞에 올 수 있습니다.

**매개변수 값:**

**-dbname**

대상 데이터베이스 이름

**-dbuser**

데이터베이스에 연결 중인 사용자의 이름

**-dbpwd**

데이터베이스에 연결 중인 사용자의 암호

**-infile** 입력 XML 파일 이름

**-method**

데이터베이스로 데이터를 삽입할 때 사용할 로더의 조작 모드

- Load 메소드는 데이터베이스 공급업체로의 원시 로더를 사용합니다. 로컬 및 원격 Oracle 데이터베이스 모두에 대하여 Load 메소드를 사용할 수 있습니다. 하지만 오직 로컬 DB2 데이터베이스에 대해서만 Load 메소드를 사용할 수 있습니다.

- 반입 방법이 로컬 또는 원격 데이터베이스로 데이터를 로드하는데 사용될 수 있지만, 이것은 보통 원격 DB2 데이터베이스로의 데이터 로드를 위해 사용됩니다. 데이터베이스 공급업체에서 사용 가능한 경우 이 메소드는 반입 또는 갱신 옵션을 사용합니다. 반입 또는 갱신 옵션이 사용 가능하지 않은 데이터베이스에 대해 이 방법을 지정한 경우, JDBC를 사용하는 Oracle 및 SQL 명령문이 데이터베이스에서 갱신을 위해 사용됩니다.
- SQL Import(sqlimport) 메소드는 로컬 및 원격 데이터베이스 둘 다와 함께 사용될 수 있습니다.
- 삭제 메소드는 데이터베이스에서 데이터를 삭제합니다.

#### **-noprimary**

1차 키가 입력 파일에 있는 레코드에 대해 누락되었을 때 로더가 취해야 하는 조치

- 오류 옵션은 누락된 1차 키를 오류로 보고하며 종료해야 함을 나타냅니다.
- 건너뛰기 옵션은 1차 키가 없는 입력 파일에서 임의의 레코드를 건너뛵니다.
- 삽입 옵션은 데이터를 삽입 또는 삭제하려고 시도합니다.

이 매개변수는 선택적입니다. 기본 조치는 오류입니다.

#### **-commitcount**

SQL Import 조작 메소드를 사용할 때 데이터베이스 확약이 발생하기 전에 처리되는 레코드 수. 이 매개변수는 선택적입니다. 기본 수는 1입니다.

#### **-maxerror**

SQL Import 조작 메소드에서 로더가 종료한 이후의 오류 수. 이 매개변수는 선택적입니다.

#### **-customizer**

사용될 조정자 특성 파일의 이름. 이 매개변수는 선택적입니다. 사용자 정의 특성 파일은 로더가 기능하는 방법을 설정합니다.

MassLoadCustomizer.properties는 기본 파일입니다. 사용자 정의 프로그램 특성 파일은 다음 예처럼 지정될 수 있습니다.

```
-customizer d:\WebSphere\CommerceServer\prop\ml.properties
```

클래스 경로 시스템 환경 변수에 지정된 디렉토리에 파일이 존재하면, 동일한 파일이 다음 예처럼 지정될 수 있습니다.

```
-customizer ml
```

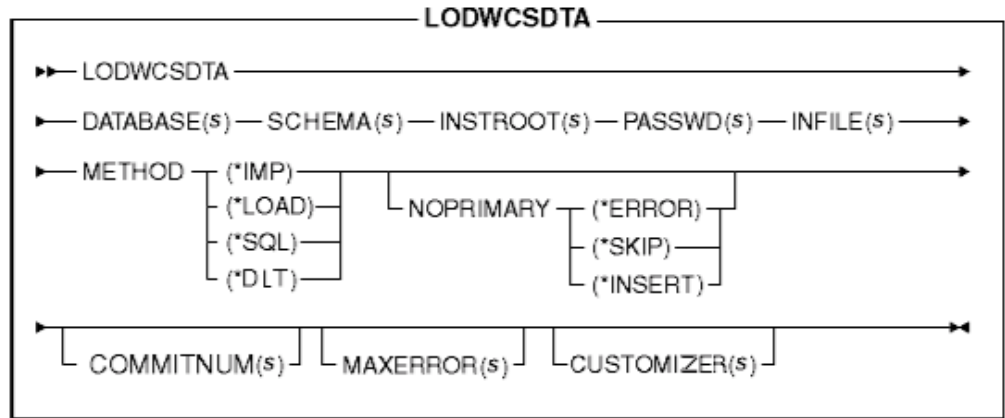
새 사용자 정의 특성 파일 작성 및 지정에 대한 자세한 정보는 *IBM WebSphere Commerce 5.4 Catalog Manager 사용자 안내서*의 최신 버전을 참조하십시오.

#### **-schemaname**

대상 데이터베이스 스키마의 이름. 이 매개변수는 선택적입니다.

명령을 실행할 때 이 매개변수가 지정되지 않은 경우, 이 명령은 SchemaName의 값을 지정하는 사용자 정의 특성 파일에서 이름=값 쌍을 찾습니다. 이 쌍이 이 특성 파일에 존재할 경우 명령은 지정된 값을 사용합니다. 명령행 또는 특성 파일 스펙 모두에 이 매개변수가 존재하지 않는 경우, 이 명령은 데이터베이스 테이블의 스키마 이름을 기본값으로 합니다.

▶ 400



주: 이 명령에 대한 매개변수로서 지정된 파일 이름은 상대 또는 절대 경로가 앞에 올 수 있습니다.

### 매개변수 값:

#### DATABASE

관계형 데이터베이스 디렉토리에 표시된 대로의 대상 데이터베이스의 이름

#### SCHEMA

대상 데이터베이스 스키마의 이름으로, 인스턴스 이름과 같습니다.

#### INSTROOT

/QIBM/UserData/WebCommerce/instances/*instance\_name*과 같은 WebSphere Commerce 인스턴스 루트 경로의 전체 이름

#### PASSWD

WebSphere Commerce 인스턴스의 암호

#### INFILE

입력 XML 파일 이름

#### METHOD

데이터베이스로 데이터를 삽입할 때 사용할 로더의 조작 모드

- Load 메소드(\*LOAD)는 데이터베이스 공급업체의 원시 로더를 사용합니다. 로컬 및 원격 Oracle 데이터베이스 모두에 대하여 Load 메소드(\*LOAD)를

사용할 수 있습니다. 하지만 오직 로컬 DB2 데이터베이스에 대해서만 Load 메소드(\*LOAD)를 사용할 수 있습니다.

- 반입 방법(\*IMP)이 로컬 또는 원격 데이터베이스로 데이터를 로드하는데 사용될 수 있지만, 이것은 보통 원격 DB2 데이터베이스로의 데이터 로드를 위해 사용됩니다. 데이터베이스 공급업체에서 사용 가능한 경우 이 메소드는 반입 또는 갱신 옵션을 사용합니다. 반입 또는 갱신 옵션이 기재되지 않은 경우, JDBC를 사용하는 SQL 문은 데이터베이스를 갱신하기 위해 사용됩니다.
- SQL Import(\*SQL) 메소드는 로컬 및 원격 데이터베이스 둘 다와 함께 사용될 수 있습니다.
- 삭제(\*DLT) 메소드는 데이터베이스에서 데이터를 삭제합니다.

### **NOPRIMARY**

1차 키가 입력 파일에 있는 레코드에 대해 누락되었을 때 로더가 취해야 하는 조치

- 오류 옵션(\*ERROR)은 누락된 1차 키를 오류로 보고하며 종료해야 함을 나타냅니다.
  - 건너뛰기 옵션(\*SKIP)은 1차 키가 없는 입력 파일에서 임의의 레코드를 건너뛵니다.
  - 삽입 옵션(\*INSERT)은 데이터를 처리(삽입 또는 삭제)하려고 시도합니다.
- 이 매개변수는 선택적입니다. 기본 조치는 오류입니다.

### **COMMITNUM**

SQL Import 조작 메소드를 사용할 때 데이터베이스 확약이 발생하기 전에 처리되는 레코드 수. 이 매개변수는 선택적입니다. 기본 수는 1입니다.

### **MAXERROR**

SQL Import 조작 메소드에서 로더가 종료한 이후의 오류 수. 이 매개변수는 선택적입니다.

### **CUSTOMIZER**

사용될 조정자 특성 파일의 이름. 이 매개변수는 선택적입니다. 사용자 정의 특성 파일은 ID 분석기가 기능하는 방법을 설정합니다. 기본 파일은 ISeries\_LODWCSDTA\_Customizer.properties입니다. 사용자 정의 프로그램 특성 파일은 다음 예처럼 지정될 수 있습니다.

```
CUSTOMIZER(/wc/prop/m1.properties)
```

클래스 경로 시스템 환경 변수에 지정된 디렉토리에 파일이 존재하면, 동일한 파일이 다음 예처럼 지정될 수 있습니다.

```
CUSTOMIZER(m1)
```

새 사용자 정의 특성 파일 작성 및 지정에 대한 자세한 정보는 *IBM WebSphere Commerce 5.4 Catalog Manager 사용자 안내서*의 최신 버전을 참조하십시오.

### load 명령에 대한 메소드:

데이터를 로드하기 전에, 최적의 결과를 얻기 위한 세 가지 프로세스의 메소드를 결정해야 합니다.


### Load 메소드:

다음 경우에는 Load 메소드를 고려해야 합니다.



- 소스 데이터가 지워지고 데이터베이스는 모든 데이터를 포함하지 않습니다.

주: 제거 데이터는 로드 중인 데이터로 테이블의 제한자를 어기는 데이터입니다.





- 소스 데이터가 지워지고 데이터베이스는 로드 중인 데이터를 포함하지 않습니다.
- 소스 데이터가 지워지고 하나 이상의 대상 테이블은 기본 키를 포함하지 않고, 데이터베이스는 로드 중인 데이터를 포함하지 않습니다.
- 데이터베이스는 로컬 DB2 데이터베이스입니다.
- 로컬 및 원격 Oracle 데이터베이스입니다.
- 데이터베이스는 로드가 발생하는 동안 기타 사용자 또는 응용프로그램에 의해 액세스할 수 없습니다.

 Load 메소드를 사용하고, 데이터는 데이터베이스로 로드됩니다. 데이터가 이미 있는 경우, 명령은 중복 키 오류 및 중복 오류 메시지의 결과로 실패합니다.

다음 제한 사항은 Load 메소드를 사용하여 존재합니다.

- Load 메소드는 비트 데이터 필드에 데이터를 삽입 또는 갱신할 수 없습니다.
-  Load 메소드를 사용하고, 새 레코드만이 데이터베이스로 삽입됩니다. 기존 레코드는 갱신되지 않습니다.
-  로드 방법은 원격이 아닌 로컬 DB2 데이터베이스에 대해서만 사용될 수 있습니다.

### Import 메소드:

    DB2용 Import 메소드를 사용하고, 또한 데이터는 데이터베이스로 로드됩니다. 데이터가 이미 있는 경우, 삭제되지는 않지만 새 값으로 갱신됩니다. 다음 경우에는 이 메소드를 고려해야 합니다.

- 데이터베이스 관리 시스템이 DB2입니다.
- 데이터가 지워지는지 여부를 모릅니다.
- 열 레벨에서 거대한 동일 데이터의 세트를 갱신해야 합니다.
- 데이터로 반입 중인 모든 테이블에는 기본 키가 있습니다.

▶ 400 Import 메소드를 사용하고, 데이터는 데이터베이스로 또한 로드됩니다. 데이터가 이미 있는 경우, 삭제되지는 않지만 새 값으로 갱신됩니다. 다음 경우에는 이 메소드를 고려해야 합니다.

- 데이터가 지워지는지 여부를 모릅니다.
- 데이터는 데이터베이스에 이미 있습니다.
- 데이터로 반입 중인 모든 테이블에는 기본 키가 있습니다.

다음 제한 사항은 Import 메소드를 사용할 때 존재합니다.

- 데이터베이스 관리 시스템은 Import 메소드를 사용하기 위해 DB2이어야 합니다.
- Import 메소드는 비트 데이터 필드에 데이터를 삽입 또는 갱신할 수 없습니다.
- Import 메소드를 사용하고, 로더는 테이블에 정의된 1차 키를 가진 테이블을 삽입 또는 갱신만 합니다. Import 메소드는 1차 키를 없는 테이블에 데이터를 삽입 또는 갱신할 수 없습니다. 입력 레코드가 1차인 열에 대한 값을 가지고 있는 경우, 레코드는 거부됩니다.

#### **SQL Import 메소드:**

SQL 입력 메소드로, JDBC 또는 SQL 명령문이 데이터베이스로 데이터를 삽입 또는 갱신하기 위해 사용됩니다. 데이터가 이미 있는 경우, 데이터가 삽입되고 기존 데이터가 갱신됩니다. 다음 경우에는 이 메소드를 고려해야 합니다.

- 기존 데이터를 갱신 중이고 열 레벨 갱신이 필요합니다.
- 일부 데이터는 지워지지 않습니다.
- 데이터가 로컬이 아닙니다.

주: 상품 어드바이저 검색 공간 동기화를 사용하려는 경우, 데이터 로드에는 SQL Import 메소드를 사용해야 합니다.

#### **삭제 메소드:**

삭제 메소드는 데이터베이스에서 입력 XML 문서에 있는 데이터를 삭제하기 위해 사용됩니다. 요소는 1차 키에 대한 값 또는 테이블에 대한 고유 색인을 포함해야 합니다. 삭제 중인 데이터가 사용 가능한 "삭제 시 계단식"으로 종속적인 다른 테이블에 데이터를 가지고 있는 경우, 종속 데이터 또한 삭제됩니다.

#### **메소드 비교:**

##### **• SQL Import 및 Load 메소드 비교**

외부 참조를 포함하는 SQL Import 메소드는 데이터 일관성을 확인하고 기존 데이터를 갱신할 수 있습니다. Load 메소드는 다음을 수행할 수 없습니다.

##### **• Import 및 SQL Import 메소드 비교**

Import 및 SQL Import 메소드는 유사한 기능을 수행합니다. Import 메소드는 일반적으로 더 빠르지만, 임시 파일에 대한 디스크 공간이 필요합니다.

Import 메소드는 테이블에서 정의된 1차 키가 있는 테이블을 삽입 또는 갱신만 할 수 있습니다. 하지만, SQL Import 메소드에는 테이블에 1차 키가 있어야 합니다.

- 사용되는 데이터베이스 상품에 기반하여 메소드 비교

Import 및 Load 메소드는 DB2에 최적화된 원시 유틸리티를 사용하는 반면, SQL Import 메소드는 JDBC 호출(많은 데이터베이스 상품에 일반적인)을 사용합니다.

**성능 고려사항:**

데이터베이스로 큰 문서를 로드하기 위해 로더를 사용할 때, 다음 항목을 고려해야 합니다.

- JVM(Java Virtual Machine) 힙 크기

기본값으로, JVM 힙에 할당된 최대 메모리 양은 64MB입니다. 메모리 양이 증가되지 않는 경우, 로드 프로세스 동안 결과적으로 JVM은 메모리가 부족할 수 있습니다. Java 힙에 할당된 최대 메모리 양은 Java 명령에서 JVM-mx 옵션을 사용하여 다양해질 수 있습니다.

- 추적 로그

추적 로그 프로그램은 큰 XML 문서를 로드할 때, JVM 힙을 소비합니다. 실행이 실패하는 경우, 추적 정보는 실행을 디버그하기 위해 사용됩니다. 로드 프로세스 추적이 필요하지 않는 경우, 추적이 꺼져 있어야 합니다. 추적이 꺼져 있을 때, 다시 성능이 커집니다. 로깅 문서 XML 문서를 수정해서 추적을 끌 수 있습니다. 로깅 구성 XML 문서 수정 정보에 대해서는 *IBM WebSphere Commerce 5.4 Catalog Manager* 사용자 안내서의 최신 버전을 참조하십시오.

- 확약 계수

SQL Import 모드에서 작동할 때, 로더에 대한 기본 확약 계수는 1입니다. 기본값으로, 트랜잭션은 모든 갱신에서 확약되거나 데이터베이스로 삽입됩니다. 큰 문서에 대한 로더의 성능을 향상시키려면, 확약 계수는 증가되어야 합니다. 값 "100"이 제공되었지만 서버의 실제 메모리의 양, DBMS 트랜잭션 로그 크기 기타 여러 가지에 따라 더 높아질 수 있습니다.

로더에 대한 확약 계수는 load 명령에 대한 -commitcount 계수 옵션을 사용하여 변경됩니다(여기서, 계수는 트랜잭션이 확약되기 전에 실행 명령문의 수입니다).

- 로깅 구성

데이터 로드가 천천히 진행되는 경우 다음 중 하나가 원인일 수 있습니다.

- 로더를 호출하는 사용자는 디렉토리에 쓰기 권한 또는 로깅 구성 문서에서 지정된 파일을 갱신할 권한이 없습니다.
- 로깅 구성 문서의 파일 위치로 지정된 디렉토리는 없습니다.
- 로깅 구성 문서의 파일 위치로 지정된 드라이브에 충분한 공간이 없습니다.

문제점을 수정할 때, 파일의 지정된 위치를 변경해서 로깅 구성 문서를 수정해야 합니다(기본값으로, WCALoggerConfig.xml). 로깅 구성 XML 문서 수정 정보에 대해서는 *IBM WebSphere Commerce 5.4 Catalog Manager 사용자 안내서*의 최신 버전을 참조하십시오.







이 명령을 실행하기 위해 사용된 파일을 설치하고 사용자 정의할 수 있는 자세한 정보에 대해서는 *IBM WebSphere Commerce 5.4 Catalog Manager 사용자 안내서*의 최신 버전을 참조하십시오.

## 데이터를 변환 및 추출하기 위한 로더 패키지 명령

### DTD Generate 명령

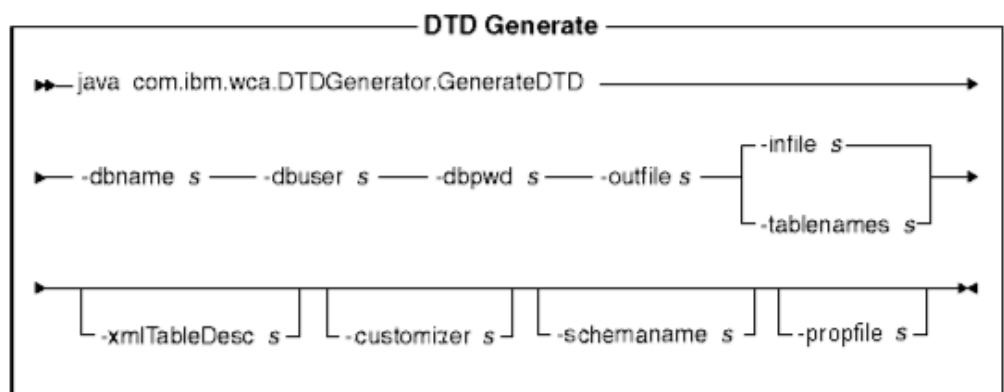
이 명령은 로더 패키지로 사용하기 위해 DTD를 작성합니다. 이 DTD는 데이터 로드 프로세스를 통해 사용됩니다. 명령을 호출하는 방법에 따라, DTD 작성기는 DTD 또는 XML 스키마가 있는 DTD를 작성합니다.

DTD 작성기는 WebSphere Commerce 데이터베이스 스키마를 기반으로 DTD를 작성할 수 있습니다. 견본 상점 아카이브가 제공된 DTD를 사용하고 데이터베이스 스키마를 수정하지 않는 경우, DTD 작성기를 사용하여 DTD를 생성할 필요가 없습니다. 제공된 DTD는 다음 디렉토리에 있습니다.

-  `drive:\WebSphere\CommerceServer\xml\sar`
-  `drive:\Program Files\WebSphere\CommerceServer\xml\sar`
-  `/usr/WebSphere/CommerceServer/xml/sar`
-   `/opt/WebSphere/CommerceServer/xml/sar`
-  `/QIBM/ProdData/WebCommerce/xml/sar`

제공된 DTD를 사용하는 것이 권장됩니다. 하지만, 데이터베이스 스키마를 사용자 정의하는 경우 제공된 DTD를 편집하여 변경사항과 일치시키거나 새 DTD를 작성하여야 합니다.





주:

1. 위 도표는 명령 매개변수에 대한 참조용입니다. 이 명령에 대해 제공되고 296 페이지의 『로더 패키지 명령 및 스크립트 사용』 아래 나열된 명령 파일 또는 스크립트는 실제 Java 명령에 래퍼로써 수행되고 동일한 매개변수를 승인합니다. 따라서 사용자가 명령 파일 또는 스크립트를 사용하는 것이 Java 명령을 직접 호출하는 것보다 권장됩니다.
2. 이 명령에 대한 매개변수로서 지정된 파일 이름은 상대 또는 절대 경로가 앞에 올 수 있습니다.

#### 매개변수 값:

##### **-dbname**

대상 데이터베이스 이름

##### **-dbuser**

데이터베이스에 연결 중인 사용자의 이름

##### **-dbpwd**

데이터베이스에 연결 중인 사용자의 암호

##### **-outfile**

출력 DTD 파일 이름(.dtd 확장자를 가짐)

**-infile** 각 행에 데이터베이스 테이블을 포함하는 입력 파일 이름

##### **-tablenamees**

섬표와 큰따옴표(" ")로 분리된 테이블 이름

##### **-xmltabledesc**

작성된 XML 스키마 파일의 파일 경로. 이 매개변수는 선택적입니다.

##### **-customizer**

사용될 조정자 특성 파일의 이름. 이 매개변수는 선택적입니다. 사용자 정의 특성 파일은 DTD 작성기가 기능하는 방법을 설정합니다.

DB2ConnectionCustomizer.properties 파일이 기본 파일입니다. 사용자 정의 프로그램 특성 파일은 다음 예처럼 지정될 수 있습니다.

```
-customizer d:\WebSphere\CommerceServer\prop\dttdgen.properties
```

클래스 경로 시스템 환경 변수에 지정된 디렉토리에 파일이 존재하면, 동일한 파일이 다음 예처럼 지정될 수 있습니다.

```
-customizer dttdgen
```

새 사용자 정의 특성 파일 작성 및 지정에 대한 자세한 정보는 *IBM WebSphere Commerce 5.4 Catalog Manager 사용자 안내서*의 최신 버전을 참조하십시오.

**-schemaname**

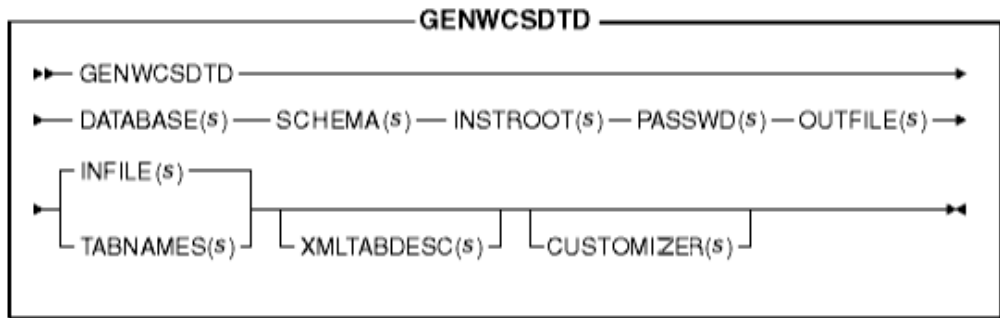
대상 데이터베이스 스키마의 이름. 이 매개변수는 선택적입니다.

명령을 실행할 때 이 매개변수가 지정되지 않은 경우, 이 명령은 SchemaName의 값을 지정하는 사용자 정의 특성 파일에서 이름=값 쌍을 찾습니다. 이 쌍이 이 특성 파일에 존재할 경우 명령은 지정된 값을 사용합니다. 명령행 또는 특성 파일 스펙 모두에 이 매개변수가 존재하지 않는 경우, 이 명령은 데이터베이스 사용자의 이름을 기본값으로 합니다.

**-propfile**

도움말 텍스트, 기본값 및 필드 설명 정보가 설명에서 웹 편집기에 대해 저장될 수 있는 특성 파일. 이 매개변수는 선택적입니다.

▶ 400



주: 이 명령에 대한 매개변수로서 지정된 파일 이름은 상대 또는 절대 경로가 앞에 올 수 있습니다.

**매개변수 값:**

**DATABASE**

관계형 데이터베이스 디렉토리에 표시된 대로의 대상 데이터베이스의 이름

**SCHEMA**

대상 데이터베이스 스키마의 이름으로, 인스턴스 이름과 같습니다.

**INSTROOT**

/QIBM/UserData/WebCommerce/instances/*instance\_name*과 같은 WebSphere Commerce 인스턴스 루트 경로의 전체 이름

**PASSWD**

WebSphere Commerce 인스턴스의 암호

**OUTFILE**

출력 DTD 파일 이름(.dtd 확장자를 가짐)

**INFILE**

각 행에 데이터베이스 테이블을 포함하는 입력 파일 이름

## TABNAMES

쉼표와 따옴표로 분리된 테이블 이름

## XMLTABDESC

작성된 XML 스키마 파일의 파일 경로. 이 매개변수는 선택적입니다.

## CUSTOMIZER

사용될 조정자 특성 파일의 이름. 이 매개변수는 선택적입니다. 사용자 정의 특성 파일은 DTD 작성기가 기능하는 방법을 설정합니다. 기본 파일은 ISeries\_GENWCSDTD\_Customizer.properties입니다. 사용자 정의 프로그램 특성 파일은 다음 예처럼 지정될 수 있습니다.

```
CUSTOMIZER(/wc/prop/dtdgen.properties)
```

클래스 경로 시스템 환경 변수에 지정된 디렉토리에 파일이 존재하면, 동일한 파일이 다음 예처럼 지정될 수 있습니다.

```
CUSTOMIZER(dtdgen)
```

새 사용자 정의 특성 파일 작성 및 지정에 대한 자세한 정보는 *IBM WebSphere Commerce 5.4 Catalog Manager 사용자 안내서*의 최신 버전을 참조하십시오.

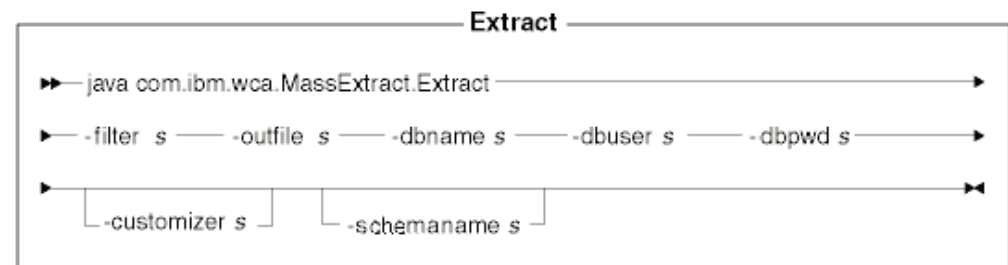
이 명령을 실행하기 위해 사용된 파일을 설치하고 사용자 정의할 수 있는 자세한 정보에 대해서는 *IBM WebSphere Commerce 5.4 Catalog Manager 사용자 안내서*의 최신 버전을 참조하십시오.

## Extract 명령

이 명령은 XML 파일 양식의 데이터베이스에서 데이터의 선택된 서브세트를 추출합니다.

추출기를 사용하여 데이터를 추출하려면, 추출 필터 파일을 사용하여 데이터베이스에서 추출하려는 데이터를 지정해야 합니다. 사용하려는 추출 필터는 추출하려는 데이터 유형에 따라 다릅니다.

Windows AIX Solaris Linux



주:

1. 위 도표는 명령 매개변수에 대한 참조용입니다. 이 명령에 대해 제공되고 296 페이지의 『로더 패키지 명령 및 스크립트 사용』 아래 나열된 명령 파일 또는 스크립트는 실제 Java 명령에 래퍼로써 수행되고 동일한 매개변수를 승인합니다. 따라서 사용자가 명령 파일 또는 스크립트를 사용하는 것이 Java 명령을 직접 호출하는 것보다 권장됩니다.
2. 이 명령에 대한 매개변수로서 지정된 파일 이름은 상대 또는 절대 경로가 앞에 올 수 있습니다.

#### 매개변수 값:

**-filter** 추출 필터 파일 이름

**-outfile**

추출된 데이터가 저장될 출력 XML 파일 이름

**-dbname**

추출 중인 데이터에서 데이터베이스 이름

**-dbuser**

추출 중인 데이터에서 데이터베이스의 데이터베이스 사용자 이름

**-dbpwd**

추출 중인 데이터에서 데이터베이스의 사용자 이름과 관련된 암호

**-customizer**

사용될 조정자 특성 파일의 이름. 사용자 정의 특성 파일은 추출기가 기능하는 방법을 설정합니다.

DB2ConnectionCustomizer.properties는 기본 파일입니다. 사용자 정의 프로그램 특성 파일은 다음 예처럼 지정될 수 있습니다.

```
-customizer d:\WebSphere\CommerceServer\prop\extract.properties
```

클래스 경로 시스템 환경 변수에 지정된 디렉토리에 파일이 존재하면, 동일한 파일이 다음 예처럼 지정될 수 있습니다.

```
-customizer extract
```

새 사용자 정의 특성 파일 작성 및 지정에 대한 자세한 정보는 *IBM WebSphere Commerce 5.4 Catalog Manager 사용자 안내서*의 최신 버전을 참조하십시오.

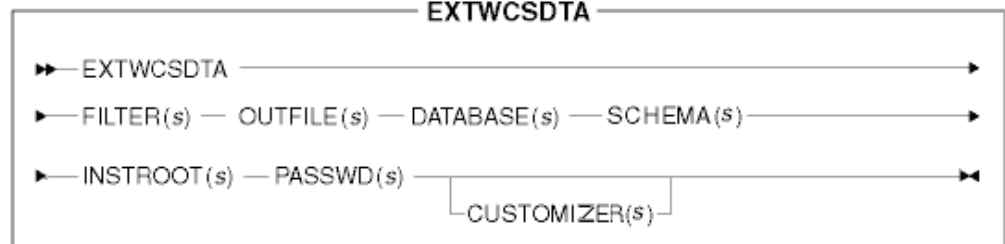
**-schemaname**

추출 중인 데이터에서 데이터베이스 스키마 이름. 이 매개변수는 선택적입니다.

명령을 실행할 때 이 매개변수가 지정되지 않은 경우, 이 명령은 SchemaName의 값을 지정하는 사용자 정의 특성 파일에서 이름=값 쌍을 찾습니다. 이 쌍이 이 특성 파일에 존재할 경우 명령은 지정된 값을 사용합니다. 명령행 또는 특

성 파일 스펙 모두에 이 매개변수가 존재하지 않는 경우, 이 명령은 데이터베이스 테이블의 스키마 이름을 기본값으로 합니다.

▶ 400



주: 이 명령에 대한 매개변수로서 지정된 파일 이름은 상대 또는 절대 경로가 앞에 올 수 있습니다.

**매개변수 값:**

**FILTER**

추출 필터 파일 이름

**OUTFILE**

추출된 데이터가 저장될 출력 XML 파일 이름

**DATABASE**

관련 데이터베이스 디렉토리에서 표시되는 추출 중인 데이터에서 데이터베이스 이름

**SCHEMA**

추출 중인 데이터에서 데이터베이스 스키마 이름으로, 인스턴스 이름과 같습니다.

**INSTROOT**

/QIBM/UserData/WebCommerce/instances/*instance\_name*과 같은 WebSphere Commerce 인스턴스 루트 경로의 전체 이름

**PASSWD**

WebSphere Commerce 인스턴스의 암호

**CUSTOMIZER**

사용될 조정자 특성 파일의 이름. 사용자 정의 특성 파일은 추출기가 기능하는 방법을 설정합니다. 기본 파일은 ISeries\_EXTWCSDTA\_Customizer.properties 입니다. 사용자 정의 프로그램 특성 파일은 다음 예처럼 지정될 수 있습니다.

CUSTOMIZER(/wc/prop/extract.properties)

클래스 경로 시스템 환경 변수에 지정된 디렉토리에 파일이 존재하면, 동일한 파일이 다음 예처럼 지정될 수 있습니다.

CUSTOMIZER(extract)

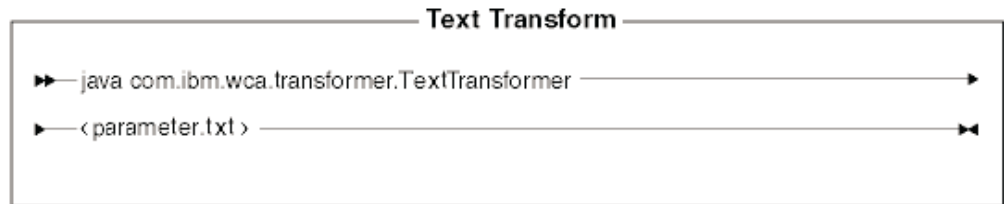
새 사용자 정의 특성 파일 작성 및 지정에 대한 자세한 정보는 *IBM WebSphere Commerce 5.4 Catalog Manager 사용자 안내서*의 최신 버전을 참조하십시오.

이 명령의 자세한 정보에 대해서는 *IBM WebSphere Commerce 5.4 Catalog Manager 사용자 안내서*의 최신 버전을 참조하십시오.

## Text Transform 명령

이 명령은 문자 분리 변수 포맷 및 XML 포맷 사이에 데이터를 변환합니다.

Windows AIX Solaris Linux



주: 위 도표는 명령 매개변수에 대한 참조용입니다. 이 명령에 대해 제공되고 296 페이지의 『로더 패키지 명령 및 스크립트 사용』 아래 나열된 명령 파일 또는 스크립트는 실제 Java 명령에 래퍼로서 수행되고 동일한 매개변수를 승인합니다. 따라서 사용자가 명령 파일 또는 스크립트를 사용하는 것이 Java 명령을 직접 호출하는 것보다 권장됩니다.

### 매개변수 값:

다음 값은 매개변수 파일(*parameter.txt*)에서 쉼표로 분리되고 지정됩니다.

#### 입력 파일

변환될 파일 이름

#### 스키마 파일

변환에서 사용될 XML 스키마 파일 이름

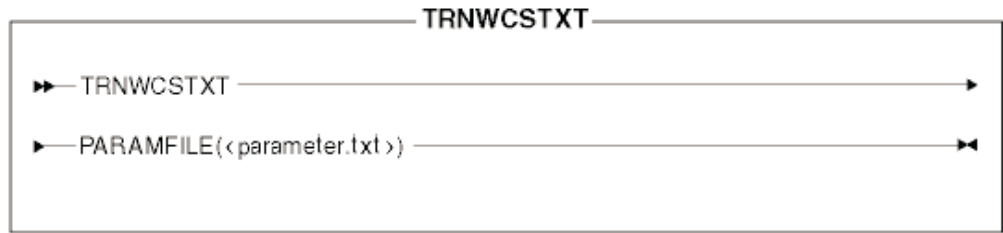
#### 출력 파일

변환된 데이터가 저장될 출력 파일의 이름

#### 변환 메소드

출력 파일로 데이터를 추가시 사용되는 메소드. 새 파일이 작성되는 경우, **create** 을 지정하거나 출력 데이터가 기존 데이터 파일에 첨부된 경우 **appended**를 지정하십시오.

또한, 이 파일을 "적하 목록" 또는 "명령" 파일이라고도 합니다. 네 개의 매개변수 복수 행을 포함할 수 있습니다.



**매개변수 값:**

다음 값은 매개변수 파일(*parameter.txt*)에서 쉼표로 분리되고 지정됩니다.

**입력 파일**

변환될 파일 이름

**스키마 파일**

변환에서 사용될 XML 스키마 파일 이름

**출력 파일**

변환된 데이터가 저장될 출력 파일의 이름

**변환 메소드**

출력 파일로 데이터를 추가 시 사용되는 메소드. 새 파일이 작성되는 경우, **Create**을 지정하거나 출력 데이터가 기존 데이터 파일에 첨부된 경우 **appended**를 지정하십시오.

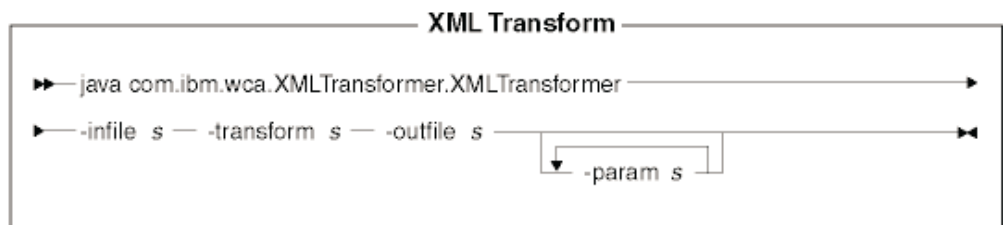
주: 또한, 이 파일을 "적하 목록" 또는 "명령" 파일이라고도 합니다.

이 명령의 자세한 정보에 대해서는 *IBM WebSphere Commerce 5.4 Catalog Manager 사용자 안내서*의 최신 버전을 참조하십시오.

**XML Transform 명령**

이 명령은 XML 파일을 대체 XML 포맷으로 변환합니다.

▶ Windows ▶ AIX ▶ Solaris ▶ Linux



**주:**

1. 위 도표는 명령 매개변수에 대한 참조용입니다. 이 명령에 대해 제공되고 296 페이지의 『로더 패키지 명령 및 스크립트 사용』 아래 나열된 명령 파일 또는 스크립트

는 실제 Java 명령에 래퍼로써 수행되고 동일한 매개변수를 승인합니다. 따라서 사용자가 명령 파일 또는 스크립트를 사용하는 것이 Java 명령을 직접 호출하는 것보다 권장됩니다.

- 이 명령에 대한 매개변수로서 지정된 파일 이름은 상대 또는 절대 경로가 앞에 올 수 있습니다.

**매개변수 값:**

**-infile** 변환될 파일 이름

**-transform**

변환 XSL 맵핑 규칙 파일의 이름

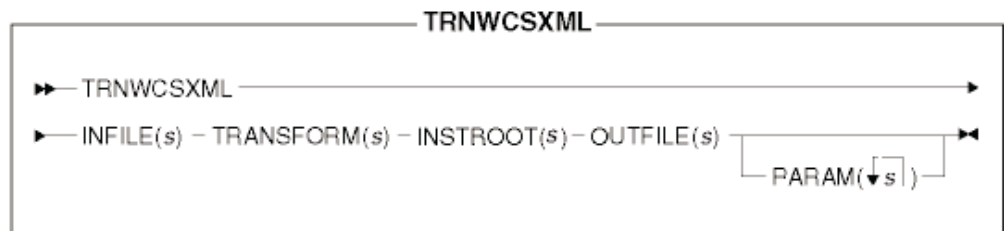
**-outfile**

변환된 데이터가 저장될 출력 XML 파일의 이름

**-param**

XSL 맵핑 규칙 파일로 전달되는 매개변수 *이 매개변수는 선택적입니다.* 이 매개변수를 여러번복수 이름=값 쌍을 전달할 수 있습니다.

▶ 400



주: 이 명령에 대한 매개변수로서 지정된 파일 이름은 상대 또는 절대 경로가 앞에 올 수 있습니다.

**매개변수 값:**

**INFILE**

변환될 파일 이름

**TRANSFORM**

변환 XSL 맵핑 규칙 파일의 이름

**INSTROOT**

/QIBM/UserData/WebCommerce/instances/*instance\_name*과 같은 WebSphere Commerce 인스턴스 루트 경로의 전체 이름

**OUTFILE**

변환된 데이터가 저장될 출력 XML 파일의 이름



## PARAM

XSL 맵핑 규칙 파일로 전달되는 매개변수. 이 매개변수는 선택적입니다. 문자열에 복수 값을 포함하여 복수 이름=값 쌍을 전달할 수 있습니다.

이 명령의 자세한 정보에 대해서는 *IBM WebSphere Commerce 5.4 Catalog Manager 사용자 안내서*의 최신 버전을 참조하십시오.

## 로더 패키지 명령과 관련된 도구

### 텍스트 변환 도구

텍스트 변환 도구는 텍스트 변환 명령을 사용하여 문자 분리 변수 포맷 및 XML 포맷 사이의 데이터 변환을 처리합니다. 다음 보기가 제공됩니다.

1. 텍스트 스키마 편집 보기를 사용하여 변환에서 사용되는 XML 스키마 파일을 작성하고 수정할 수 있습니다.
2. 변환 명령 편집 보기를 사용하여 변환 프로세스를 실행하기 위해 사용되는 실제 명령을 작성하고 변환할 수 있습니다.
3. 변환 명령 프로세스 보기를 사용하여 변환 프로세스를 시작할 수 있습니다.

이 도구의 자세한 정보에 대해서는 *IBM WebSphere Commerce 5.4 Catalog Manager 사용자 안내서*의 최신 버전을 참조하십시오.

### XSL 편집기

XML 변환기는 XSL을 사용하여 XML 파일을 또 다른 XML 파일로 변환하기 위해 규칙을 정의합니다. XSLM 편집기에서 맵핑 기능에서는 소스 DTD의 요소에서 대상 DTD의 요소로 연관을 설정할 수 있는 비주얼 인터페이스를 제공합니다. 두 개의 DTD가 제공된 경우, 1차(소스) DTD에 맞는 XML 파일이 2차(대상) DTD에 맞는 파일로 변환되는 메소드를 결정하는 XSL 규칙을 개발할 수 있습니다.

이 도구의 자세한 정보에 대해서는 *IBM WebSphere Commerce 5.4 Catalog Manager 사용자 안내서*의 최신 버전을 참조하십시오.

### 웹 편집기

웹 편집기를 사용하여 웹 브라우저에서 카탈로그 데이터를 작성, 삭제 및 변경할 수 있습니다. 정보 보기 또는 갱신에 대한 데이터 입력 양식이 웹 편집기의 핵심입니다. 가장 간단한 경우, 양식은 WebSphere Commerce 서버 데이터베이스의 테이블에 해당합니다. 상점 개발자는 제공된 기본 양식을 사용하거나 사용 가능한 양식을 사용자 정의하도록 선택할 수 있습니다.

이 도구의 자세한 정보에 대해서는 *IBM WebSphere Commerce 5.4 Catalog Manager 사용자 안내서*의 최신 버전을 참조하십시오.

## 상점 데이터 로드

이 절에서는 로더 패키지 명령행 유틸리티를 사용하여 WebSphere Commerce 서버 데이터베이스로 상점 데이터를 로드하는 방법의 예제를 제공합니다.







주:

1. 이 절의 예제는 Windows NT 환경에서 수행됩니다. 기타 환경에서 이 명령 실행에 대한 정보는 *IBM WebSphere Commerce 5.4 Catalog Manager 사용자 안내서*의 최신 버전을 참조하십시오.
2. 로더 패키지 명령행 유틸리티는 DB2, iSeries용 DB2 및 Oracle 데이터베이스를 지원하지만, DB2용 명령 및 옵션만이 다음 예제에 포함됩니다. DB2 이외의 데이터베이스를 사용하는 경우, *IBM WebSphere Commerce 5.4 Catalog Manager 사용자 안내서*의 최신 버전에서 설명된 사용자 정의 프로그램 특성 파일을 수정해야 합니다.





WebSphere Commerce 데이터베이스 자원 그룹의 로드 프로세스에 대한 정보에 대해서는 307 페이지의 제 28 장 『WebSphere Commerce 데이터베이스 자원 그룹 로드』 및 316 페이지의 『데이터베이스 자원 그룹 로드』를 참조하십시오.

### 로더 패키지 명령 및 스크립트 사용

Loader package 명령을 실행하려면, 다음 WebSphere Commerce 디렉토리에서 정의된 스크립트 또는 명령을 사용하십시오.

-  NT `drive:\WebSphere\CommerceServer\bin`
-  2000 `drive:\Program Files\WebSphere\CommerceServer\bin`
-  AIX `/usr/WebSphere/CommerceServer/bin`
-  Solaris  Linux `/opt/WebSphere/CommerceServer/bin`
-  400 QWEBCOMM native library

스크립트 및 명령은 다음과 같습니다.

-  Windows
  - dtdgen.cmd** DTD Generate 명령
  - idresgen.cmd** ID Resolve 명령
  - massload.cmd** Load 명령
  - massextract.cmd** Extract 명령
  - txttransform.cmd** Text Transform 명령
  - xmltransform.cmd** XML Transform 명령
-  AIX  Solaris  Linux
  - dtdgen.sh** DTD 생성 셸 스크립트
  - idresgen.sh** ID 분석 셸 스크립트

<b>massload.sh</b>	로드 셸 스크립트
<b>massextract.sh</b>	추출 셸 스크립트
<b>txttransform.sh</b>	텍스트 변환 셸 스크립트
<b>xmltransform.sh</b>	XML 변환 셸 스크립트

• **400**

<b>GENWCSDTD</b>	DTD Generate 명령
<b>RESWC SID</b>	ID Resolve 명령
<b>LODWCS DTA</b>	Load 명령
<b>EXTWCSDTA</b>	Extract 명령
<b>TRNWCSTXT</b>	Text Transform 명령
<b>TRNWC SXML</b>	XML Transform 명령

## 식별자를 분석하는 예제

이 절에서 설명된 식별자 분석 예제는 **Business** ToolTech 견본 상점에서 상점 자원 과일을 사용합니다.

이 예제는 WebSphere Commerce Server 데이터베이스로 새 데이터를 로드하는 것을 기본으로 하므로 이 Load 메소드를 사용합니다.


이후에 XML 문서 내의 특정 요소를 수정할 필요가 있는 경우, update 메소드를 사용하여 수행할 수 있습니다. update 메소드로 할당되는 새 식별자가 없기 때문에 update 메소드를 Load 메소드보다 빨리 실행해야 합니다. update 메소드를 사용하여, 데이터 베이스 조회를 수행하여 식별자를 찾을 수 있고 식별자를 찾을 수 없는 경우, 오류가 보고됩니다. 이 처리 작동하는 방법에 대한 자세한 정보는 275 페이지의 처음 부분의 설명을 참조하십시오.

입력 XML 파일이 데이터베이스에 이미 존재하는 요소 뿐만 아니라 존재하지 않는 요소를 포함하는 경우, mixed 메소드를 사용하십시오. mixed 메소드를 사용하고, 데이터 베이스 찾아가기가 먼저 수행되고 레코드를 찾을 수 없는 경우 식별자가 요소에 지정됩니다. 의심스러우면, mixed 메소드를 사용하십시오. 로드 및 update 메소드가 mixed 메소드보다 더 빠른 성능을 제공하는 경우, mixed 메소드를 사용해서 작성된 분석 XML 파일은 오류 없는 로드 가능성이 더 큽니다.

ID 분석기 작동 방법에 대한 설명은 276 페이지의 『ID Resolve 명령에 대한 메소드』를 참조하십시오.

### 내부 별명으로 XML 파일에서 식별자 분석

WebSphere Commerce Server 데이터베이스로 데이터를 로드하기 전에 내부 명령을 사용하여 식별자를 분석하려면, 다음 예에서 표시된 것처럼 ID Resolve 명령을 실행하십시오.

1. 경로에는 296 페이지의 『로더 패키지 명령 및 스크립트 사용』에서 나열된 것처럼 적절한 ID Resolve 명령 또는 스크립트를 포함하는 디렉토리가 있습니다.  
이 예제에 대해 ID Resolve 명령 idresgen.cmd를 사용할 수 있습니다.
2. 작업 디렉토리 작성  
이 예제에 대해서는 c:\WebSphere\CommerceServer\runtime\test 디렉토리를 작성하십시오.
3. 입력 XML 파일뿐 아니라 모든 참조 DTD 파일이 ID 분석기가 찾을 수 있는 위치에 있습니다.  
이 예제에 대해서는 다음을 수행하십시오.
  - a. c:\WebSphere\CommerceServer\samples\ToolTech\ToolTech\_en\_US\_fr\_FR.sar 파일을 c:\WebSphere\CommerceServer\runtime\test로 복사하십시오.
  - b. Windows 명령 프롬프트에서 다음 명령을 입력하십시오.  
cd c:\WebSphere\CommerceServer\runtime\test
  - c. Windows 명령 프롬프트에서 다음 명령을 입력하십시오.  
jar -xvf ToolTech\_en\_US\_fr\_FR.sar  
  
이는  Business ToolTech 견본 상점 XML 파일을 c:\WebSphere\CommerceServer\runtime\test\data로 추출합니다.
  - d. Windows 명령 프롬프트에서 다음 명령을 입력하십시오.  
cp c:\WebSphere\CommerceServer\xml\sar\store.dtd  
c:\WebSphere\CommerceServer\runtime\test\data  
  
이는 store.dtd 파일을 c:\WebSphere\CommerceServer\runtime\test\data로 복사합니다.
  - e. Windows 명령 프롬프트에서 다음 명령을 입력하십시오.  
cp c:\WebSphere\CommerceServer\xml\sar\DBLoadMacros.dtd  
c:\WebSphere\CommerceServer\runtime\test\data  
  
이는 DBLoadMacros.dtd 파일을 c:\WebSphere\CommerceServer\runtime\test\data로 복사합니다.
  - f. Windows 명령 프롬프트에서 다음 명령을 입력하십시오.  
cp c:\WebSphere\CommerceServer\xml\sar\fulfillment.dtd  
c:\WebSphere\CommerceServer\runtime\test\data  
  
이는 fulfillment.dtd 파일을 c:\WebSphere\CommerceServer\runtime\test\data로 복사합니다.
4. WebSphere Commerce 스키마가 적절한 WebSphere Commerce Server 데이터베이스 인스턴스를 작성하여 필요한 부트스트랩 데이터와 함께 사용자의 데이터베이스에 로드되는지 확인하십시오.

주: 인스턴스 작성에 대한 정보는 사용자 운영체제용 WebSphere Commerce 설치 안내서를 참조하십시오.

이 예제가 사용하는 WebSphere Commerce 서버 데이터베이스 인스턴스는 상가로 불립니다. 1차 및 foreign keys는 이 데이터베이스의 KEYS 및 SUBKEYS 테이블에서 얻을 수 있습니다. 따라서 데이터베이스가 올바르게 로드되지 않는 경우 ID 분석기는 식별자를 분석할 수 없습니다.

5. store.xml 파일에서, 다음 요소를 찾을 수 있습니다.

```
<storeent
 STOREENT_ID="@storeent_id_1"
 MEMBER_ID("&MEMBER_ID;")
 TYPE="S"
 IDENTIFIER="ToolTech"
 SETCURR="USD"
/>
```

store.xml의 이 요소는 데이터베이스의 storeent 테이블로 맵핑되며 STOREENT\_ID MEMBER\_ID, TYPE, IDENTIFIER 및 SETCURR 속성이 해당 테이블의 열로 맵핑됩니다. @storeent\_id\_1 스펙은 STOREENT\_ID 속성값에 대한 내부 별명이고, &MEMBER\_ID;는 엔티티 매개변수입니다.

&MEMBER\_ID; 엔티티 값은 로더를 사용하여 로드될 수 있기 전에 대체되어야 합니다. &MEMBER\_ID;의 값이 DBLoadMacros.dtd 매크로 파일에 정의됩니다. 그리고 이 값이 해당 파일에서 대체됩니다. ID 분석기가 @storeent\_id\_1을 발견하면, 이는 기본 테이블의 캐시를 조사하여 storeent가 있는지 확인합니다. 1차 테이블이기 때문에 storeent가 있습니다. ID 분석기가 해당 테이블에 대한 카운터를 증가시켜서 가져오고 내부 별명을 그 결과와 바꿉니다. store.xml 파일의 모든 기타 항목이 동일한 방법으로 처리됩니다.

6. Windows 명령 프롬프트에서 다음 명령을 입력하십시오.

```
idresgen -dbname mall -dbuser db2admin -dbpwd db2admin -infile store.xml
-outfile c:\WebSphere\CommerceServer\runtime\test\data\store1.xml -method load
```

store1.xml에서 1차 출력 XML 단편은 다음과 같습니다.

```
<storeent
 STOREENT_ID="10001"
 MEMBER_ID("-2001"
 TYPE="S"
 IDENTIFIER="ToolTech"
 SETCCURR="USD"
/>
```

store1.xml에서 2차 출력 XML 단편은 다음과 같습니다.

```
<store
 STORE_ID="10001"
 DIRECTORY="ToolTech"
 FFMCENTER_ID=""
 LANGUAGE_ID="-1"
```

```

STOREGRP_ID="-1"
ALLOCATIONGOODFOR="43200"
BOPMPADFACTOR="0"
DEFAULTBOOFFSET="2592000"
FFMCSELECTIONFLAGS="0"
MAXBOOFFSET="7776000"
REJECTEDORDEXPIRY="259200"
RTNFFMCTR_ID=""
PRICEREFFLAGS="0"
STORETYPE="B2B"
/>

```

FFMCENTER\_ID 및 RTNFFMCTR\_ID 속성을 분석할 수 없습니다.

@ffmcenter\_id\_1 내부 별명이 분석되는 대신 비어있는 큰 따옴표("")로 대체됩니다. 이는 오류입니다. foreign key 제한자 위반으로 인해 로더를 사용하여 데이터를 적절하게 로드할 수 없습니다. 참조하는 별명을 사용하기 전에 FFMCENTER 테이블이 처리되지 않기 때문에 ID 분석기가 이 내부 별명을 분석할 수 없습니다. 이 문제를 해결하려면, 다음 세 가지 옵션 중 하나를 선택하십시오.

- 옵션 1:

- a. 다음 명령을 입력하여 fulfillment.xml 파일(FFMCENTER 테이블이 정의되는 곳)에 대한 ID 분석기를 실행하십시오.

```

idresgen -dbname mall -dbuser db2admin -dbpwd db2admin
-infile fulfillment.xml -outfile
c:\WebSphere\CommerceServer\runtime\test\data\fulfillment1.xml
-method load

```

fulfillment1.xml 출력 파일에서 분석된 요소는 다음과 같습니다.

```

<fulfillment-asset>
<ffmcenter
 FFMCENTER_ID="10001"
 MEMBER_ID="-2001"
 NAME="ToolTech Home"
 DEFAULTSHIPOFFSET="0"
 MARKFORDELETE="0"
/>
</fulfillment-asset>

```

- b. 결과 출력 파일 FFMCENTER\_ID(fulfillment1.xml)에서 FFMCENTER\_ID를 가져오고

c:\WebSphere\CommerceServer\runtime\test\data에서 store.xml 작업 사본의 @ffmcenter\_id\_1의 모든 발생에 대해 해당 키를 대체하십시오.

- c. 다음 명령을 입력하십시오.

```

idresgen -dbname mall -dbuser db2admin -dbpwd db2admin -infile store.xml
-outfile c:\WebSphere\CommerceServer\runtime\test\data\store1.xml
-method load

```

store1.xml 출력 파일의 전체 분석 요소는 다음과 같습니다.

```

<store-asset>
<storeent
 STOREENT_ID="10151"
 MEMBER_ID="-2001"
 TYPE="S"
 IDENTIFIER="ToolTech"
 SETCCURR="USD"
/>
<store
 STORE_ID="10151"
 DIRECTORY="ToolTech"
 FFMCENTER_ID="10001"
 LANGUAGE_ID="-1"
 STOREGRP_ID="-1"
 ALLOCATIONGOODFOR="43200"
 BOPMPADFACTOR="0"
 DEFAULTBOOFFSET="2592000"
 FFMSELECTIONFLAGS="0"
 MAXBOOFFSET="7776000"
 REJECTEDORDEXPIRY="259200"
 RTNFFMCTR_ID="10001"
 PRICEREFFLAGS="0"
 STORETYPE="B2B"
/>
<vendor
 VENDOR_ID="10001"
 STOREENT_ID="10151"
 VENDORNAME="Tooltech Vendor"
 MARKFORDELETE="0"
/>
<dispentrel
 AUCTIONSTATE="0"
 CATENTRY_ID="0"
 CATENTTYPE_ID="ProductBean"
 DEVICEFMT_ID="-1"
 DISPENTREL_ID="10001"
 MBRGRP_ID="0"
 PAGENAME="CatalogProductDisplay.jsp"
 STOREENT_ID="10151"
 RANK="0"
/>
<dispentrel
 AUCTIONSTATE="0"
 CATENTRY_ID="0"
 CATENTTYPE_ID="ItemBean"
 DEVICEFMT_ID="-1"
 DISPENTREL_ID="10002"
 MBRGRP_ID="0"
 PAGENAME="CatalogItemDisplay.jsp"
 STOREENT_ID="10151"
 RANK="0"
/>
<dispcgprel
 CATGROUP_ID="0"
 DEVICEFMT_ID="-1"
 DISPCGPREL_ID="10001"
 MBRGRP_ID="0"
 PAGENAME="CatalogCategories.jsp"

```

```

 STOREENT_ID="10151"
 RANK="0"
 />
<invadjcode
 ADJUSTCODE="PCNT"
 INVADJCODE_ID="10001"
 MARKFORDELETE="0"
 STOREENT_ID="10151"
/>
<invadjcode
 ADJUSTCODE="SPLG"
 INVADJCODE_ID="10002"
 MARKFORDELETE="0"
 STOREENT_ID="10151"
/>
<invadjcode
 ADJUSTCODE="DISC"
 INVADJCODE_ID="10003"
 MARKFORDELETE="0"
 STOREENT_ID="10151"
/>
<rtreason
 REASONTYPE="C"
 RTNREASON_ID="10001"
 STOREENT_ID="10151"
 MARKFORDELETE="0"
 CODE="WPR"
/>
<rtreason
 REASONTYPE="B"
 RTNREASON_ID="10002"
 STOREENT_ID="10151"
 MARKFORDELETE="0"
 CODE="DEF"
/>
<rtreason
 REASONTYPE="M"
 RTNREASON_ID="10003"
 STOREENT_ID="10151"
 MARKFORDELETE="0"
 CODE="ERR"
/>
<rtreason
 REASONTYPE="M"
 RTNREASON_ID="10004"
 STOREENT_ID="10151"
 MARKFORDELETE="0"
 CODE="WPS"
/>
</store-asset>

```

- 옵션 2:
  - a. fulfillment.xml 파일과 store.xml 파일을 fulfillment.xml 파일 (fulfillment.dtd 참조 포함)에 고유한 임의의 콘텐츠를 추가하여 store.xml로 병합하되, 아래에 있는 ffmcenter 요소가 상점 요소 앞에 오도록 하십시오.



```

<ffmcenter
 FFMCENTER_ID="@ffmcenter_id_1"
 MEMBER_ID("&MEMBER_ID;"
 NAME="ToolTech Home"
 DEFAULTBOFFSET="0"
 MARKFORDELETE="0"
/>










```

b. 병합된 파일에 대한 ID 분석기 실행


- 옵션 3: 316 페이지의 『데이터베이스 자원 그룹 로드』에 설명된 처리를 사용하여 상점 자원 데이터 그룹을 로드하십시오.

### ID 분석기를 가진 특성 파일 지정

-propfile 매개변수를 사용하여 ID 분석기가 식별자를 분석하는 방법을 수정할 수 있습니다. 기본 특성 파일은 IdResolveKeys.properties이지만, 이를 수정하거나 ID 분석 명령을 호출할 때 사용자 고유 파일을 지정할 수 있습니다.

-     IdResolveKeys.properties 파일은 다음 디렉토리에 있습니다.
  -  *drive:\WebSphere\CommerceServer\properties*
  -  *drive:\Program Files\WebSphere\CommerceServer\properties*
  -  */usr/WebSphere/CommerceServer/properties*
  -   */opt/WebSphere/CommerceServer/properties*

ID 분석기가 실행할 때, 현재 디렉토리에 이 파일을 넣지 않는 경우, 이를 classpath 환경 변수에서 정의된 디렉토리로 위치 지정할 수 있습니다. 또한, 이 파일에 전체 경로를 지정할 수 있습니다.

-  IdResolveKeys.properties 파일을 변경하려면, /QIBM/ProdData/WebCommerce/properties 디렉토리에서 이 파일을 복사하여 */instroot/xml* 디렉토리로 저장한 후, 새 파일에 대하여 필요한 변경사항을 수행하십시오.

주: 위 디렉토리는 RESWCSID 명령에 의해 사용된 classpath에 있습니다.

특성 파일 스펙은 내부 별명 사용에서 우선순위를 가집니다.

store.xml 파일에 동일한 XML 단편이 있습니다.

```

<store
 STORE_ID="@storeent_id_1"
 DIRECTORY="ToolTech"
 FFMCENTER_ID="@ffmcenter_id_1"
 LANGUAGE_ID("&en_US;"
 STOREGRP_ID="-1"
 ALLOCATIONGOODFOR="43200"
 BOPMPADFACTORr="0"
 DEFAULTBOFFSET="2592000"
 FFMCELECTIONFLAGS="0"
 MAXBOFFSET="7776000"

```

```
REJECTEDORDEXPIRY="259200"
RTNFFMCTR_ID="@ffmcenter_id_1"
PRICEREFFLAGS="0"
STORETYPE="B2B"
/>
```

c:\WebSphere\CommerceServer\runtime\test\data\myPropFile로 지정된 -propfile을 가진 ID 분석기 및 지정된 파일 myPropFile.properties를 실행하는 경우, 다음 항목을 포함합니다.

```
NAMEDELIMITER=@
SELECTDELIMITER=:
FFMCENTER=@FFMCENTER_ID@MEMBER_ID:10051 -2001
```

ID 분석기는 상점 요소가 진행되는 동안, 10051 및 -2001 절이 있는 FFMCENTER 테이블에 대한 데이터베이스를 조회합니다. 이 값에 대해 리턴된 색인이 FFMCENTER\_ID에 대해 식별자를 분석하기 위해 사용됩니다.

이 명령 사용에 대한 자세한 정보는 269 페이지의 『ID Resolve 명령』을 참조하십시오.

## 데이터 로드 예제

필요에 의해 XML 파일에서 식별자를 분석한 경우, 데이터를 WebSphere Commerce 서버 데이터베이스로 로드할 준비가 되어 있습니다.

주: 사용자의 XML 데이터에서 식별자를 적절하게 분석한 경우, 사용자의 소스 XML 파일은 다음을 포함하지 않아야 합니다.

- (@) 기호로 시작되는 단어
- 앰퍼센트(&) 기호로 시작되는 단어
- 비어있는 따옴표("")를 갖는 식별자

이들 중 하나라도 있는 경우, 사용자의 XML 파일이 로드될 준비가 되지 않았음을 나타냅니다.

이 절에 설명된 데이터 로드 예제는 297 페이지의 『식별자를 분석하는 예제』에서 분석된 fulfillment1.xml 파일을 사용합니다.

데이터를 WebSphere Commerce 서버 데이터베이스로 로드하려면, 다음 예제에서 나온 명령을 실행하십시오.

1. 경로에는 296 페이지의 『로더 패키지 명령 및 스크립트 사용』에서 나열된 것처럼 적절한 load 명령 또는 스크립트를 포함하는 디렉토리가 있습니다.

이 예제에 대해, massload.cmd 파일을 사용할 수 있습니다.

2. 작업 디렉토리 작성

297 페이지의 『식별자를 분석하는 예제』에서 작성한

c:\WebSphere\CommerceServer\runtime\test\data 디렉토리를 사용하십시오.

3. 입력 XML 파일은 로더가 찾을 수 있는 위치에 있어야 합니다.

이 예제에서는 297 페이지의 『식별자를 분석하는 예제』에서 작성한 fulfillment1.xml 출력 파일이

c:\WebSphere\CommerceServer\runtime\test\data에 있어야만 합니다.

4. 작업 디렉토리를 변경하십시오.

예를 들어, Windows 명령 프롬프트에 다음 명령을 입력하십시오.

```
cd c:\WebSphere\CommerceServer\runtime\test\data
```

5. WebSphere Commerce Server 데이터베이스를 백업하여, 복구할 수 없는 오류가 발생했을 경우 이 백업에서 데이터베이스를 복원할 수 있도록 하십시오.

6. 대상 데이터베이스로 사용자의 데이터를 로드하기 위해 분석된 사용자의 XML 파일에 대해 Load 명령을 실행하십시오.

예를 들어, Windows 명령 프롬프트에 다음 명령을 입력하십시오.

```
massload -dbname mall -dbuser db2admin -dbpwd db2admin -infile
c:\WebSphere\CommerceServer\runtime\test\data\fulfillment1.xml
-method sqlimport -commitcount 50
```

50개 미만의 요소가 로드된다 해도 이 예제에서는 -commitcount에 대하여 값 50을 지정합니다. 이는 성능 이유입니다. 기본값으로, 확약 계수는 1입니다. 이 기본값을 사용하는 경우 데이터베이스에서 작성된 각 레코드에 대하여 확약 조사가 발생합니다. 위의 예제에서, 번호를 50으로 지정하면 로드가 성공하는 경우에만 데이터베이스 입출력이 발생하고 오류가 발생하는 경우 데이터베이스에서 작성되는 것이 없도록 합니다. 많은 양의 데이터를 로드해야 하는 경우, 다음 이유로 요소의 수 만큼 확약 계수를 설정하지 않을 것을 권장합니다.

- 확약 계수 값이 많으면 메모리도 많이 필요합니다.
- 확약 계수 값이 요소의 개수보다 작을 경우, 적어도 일부 데이터가 데이터베이스에 작성됩니다. -maxerror의 값에 따라, -commitcount에 대해 더 작은 값을 지정하면 일부 데이터는 최대 오류 갯수가 초과되기 전에 데이터베이스에서 작성되고 도구가 완료됩니다. -maxerror에 대한 기본값은 1입니다.

-noprimary 옵션은 1차 키가 누락될 때, 도구가 오류를 보고하고 종료되기 위해 오류가 기본값이 됩니다.

이 예제가 상점 서비스에 의해 사용되는 순서에서 상점 자원을 로드하지 않고 308 페이지의 『데이터베이스 자원 로드 순서』에서 설명되지 않기 때문에, 297 페이지의 『식별자를 분석하는 예제』에서 분석된 store1.xml 파일은 일부 테이블의 무결성 제한자를 유효화합니다. load 메소드를 사용하여 수정하지 않고 store1.xml 파일을 로드하려는 경우, 제한자 위반은 데이터베이스를 보류 상태로 되도록 야기시킬 수 있습니다. 그래서, 단순화하기 위해 load 명령을 사용하는 이 예제는 fulfillment.xml 파일의

분석 버전을 기반으로 하고, 파일의 foreign key는 단순한 상점에서 정의된 MEMBER\_ID입니다. 이 예제는 297 페이지의 『식별자를 분석하는 예제』의 출력값인 분석된 fulfillment1.xml 파일을 로드하고 SQL 메소드를 사용합니다. XML 파일의 콘텐츠가 정리되었는지 확실하지 않은 경우, 이 예에서 처럼 -commitcount 및 -maxerror 매개변수 세트를 적절하게 사용하고 SQL Import 메소드를 사용하여 모든 데이터베이스 제한자 위반이 데이터베이스 변경이나 데이터베이스 무결성 위협 없이 보고되도록 하십시오.

이 명령 사용에 대한 자세한 정보는 278 페이지의 『load 명령』을 참조하십시오.

---

## 제 28 장 WebSphere Commerce 데이터베이스 자원 그룹 로드

모든 데이터베이스 자원을 작성하지 않고 공개 이전에 상점 아카이브로 패키징할 경우, WebSphere Commerce 로더 패키지를 사용하여 데이터베이스 자원 그룹을 로드할 수 있습니다.

이 장의 첫 번째 부분에서는 WebSphere Commerce 데이터베이스 자원 그룹 및 그룹화가 결정되는 방법을 설명합니다. 두 번째 부분에서는 WebSphere Commerce 데이터베이스로 이 데이터베이스 자원 그룹을 로드하는 프로세스를 설명합니다. 이 장을 읽기 전에 로더 패키지로 데이터베이스 자원 그룹을 로드하기 위해 필요한 사항을 이해하도록 265 페이지의 제 27 장 『상점 데이터 로드 개요』, 정보를 전체적으로 검토해야 합니다.

---

### 데이터베이스 자원 그룹

데이터베이스 자원은 작성 및 로드 프로세스를 단순화하는 그룹으로 분리됩니다. 이 데이터베이스 자원 그룹에는 논리적으로 관련된 테이블 세트를 포함합니다. 데이터 간의 관계를 로드하기 전에 데이터가 존재해야 하기 때문에 데이터베이스 자원 그룹이 구성되는 순서는 로드를 위해 중요합니다.

전체 상점에 대한 데이터베이스 자원 세트를 로드하려면 308 페이지의 『데이터베이스 자원 로드 순서』를 따라야 합니다. 데이터베이스 자원의 단일 그룹을 로드하려면, 이 그룹이 논리적으로 완료되어야 합니다. 예를 들어 상점 아카이브 공개시, 나중에 공개될 수 있도록 카탈로그 데이터베이스 자원을 생략할 수 있습니다. 이러한 경우, 카탈로그 (재고, 가격 목록 및 일부 운송 및 세금부와 데이터)에서 모든 데이터베이스 자원 종속이 공개되지 않고 보존됩니다. 생략된 데이터를 공개하려면, 해당 카탈로그 데이터베이스 자원이 논리적으로 완료되었는지 확인하십시오. 즉, 기본 항목, 카탈로그 항목, 속성 등이 제공되어야 합니다. 또한, 논리적으로 완료되어야 하는 종속적인 데이터베이스 자원을 공개해야 합니다. 다시 말하면, 각 SKU는 정의된 적절한 재고, 가격, 운송 및 세금이 있어야 합니다. 이런 경우, 논리적으로 완료된 카탈로그 데이터는 카탈로그 데이터베이스 자원 그룹이라 불립니다.


이 안내서의 이전 장에서 설명된 WebSphere Commerce 데이터베이스 자원은 그룹으로 배열될 수 있습니다. 그룹은 개별적으로 로드될 수 있는 데이터의 논리적으로 완전한 세트입니다. 각 데이터베이스 자원 그룹은 WebSphere Commerce 데이터베이스 테이블로 구성되며 377 페이지의 부록 E 『데이터베이스 자원 그룹』에서 설명된 것과 같이 외부 종속성을 가집니다. 테이블 목록은 WebSphere Commerce 건본 상점을 기반으로 하나 목록은 일반 상점으로 적용 가능합니다. 각 데이터베이스 자원 그룹에 대한

테이블 목록이 고갈되지 않고 일반 지시사항으로써 제공되어야 합니다. 상점의 특정 요구에 따라 일부 테이블을 포함 또는 배제할 필요가 있습니다.

## 데이터베이스 자원 로드 순서

데이터베이스 자원 그룹을 성공적으로 로드하기 위해 따라야 하는 특정 순서가 있습니다. 각 그룹은 구조적으로 완료되도록 고려되어야 하며 다른 데이터베이스 자원 그룹으로부터 독립적입니다. 그러나 데이터베이스 자원 그룹 내에 foreign key 관계가 있습니다. 이러한 관계(다른 그룹의 데이터와의 관계)는 데이터베이스 자원 그룹의 외부 종속성이라고 합니다.

데이터베이스 자원 그룹의 외부 종속성은 그룹을 WebSphere Commerce 데이터베이스로 로드하기 전에 충족되어야 합니다. 제공된 데이터베이스 자원 그룹의 외부 종속성으로 정의된 모든 그룹은 먼저 로드되어야 합니다. 377 페이지의 『데이터베이스 자원 그룹 종속성』에서 외부 종속성 및 관련된 테이블 목록을 찾을 수 있습니다.

주: WebSphere Commerce 상점은 상점 소유자가 필요합니다. 기본 소유자로서 사용할 수 있는 기본 조직을 사용할 수 있습니다.  이 그룹을 로드하려면, 기본값을 사용하는 대신 새 조직을 작성하십시오.

다음 순서로 데이터베이스 자원 그룹을 로드하십시오.

1. 부트스트랩 데이터에서만 종속적인 데이터베이스 자원 그룹
  - a. 조직 데이터베이스 자원을 먼저 로드하십시오.
2. 이행 소유자에 종속적인 데이터베이스 자원 그룹
  - a. 이행 데이터베이스 자원. 조직 데이터베이스 자원 그룹을 제외한 일부 다른 데이터베이스 자원 그룹은 이 그룹에 정의된 데이터에서 직접 또는 간접 외부 종속성을 가집니다.
3. 상점 소유자 조직에서 종속적인 데이터베이스 자원 그룹
  - a. 액세스 제어 데이터베이스 자원은 상점 소유자 조직(ORGENTITY\_ID)에서 종속됩니다. 이 그룹에서 정의된 데이터에서 종속적인 다른 데이터베이스 자원 그룹이 없습니다. 이는 액세스 제어 데이터베이스 자원이 언제든지 로드될 수 있음을 의미합니다. 그러나 액세스 제어 소유자는 상점 소유자와 동일해야 합니다.
  - b. 상점 데이터베이스 자원은 상점 소유자 조직(ORGENTITY\_ID)에서 종속됩니다. 상점은 서비스 센터를 참조할 수 있습니다. 상점 소유자 조직은 서비스 센터 소유자 조직이 될 수 있습니다.
4. 상점 데이터베이스 자원에서 종속적인 데이터베이스 자원 그룹. 다음 그룹은 어떠한 순서로든지 로드될 수 있습니다.
  - a. 캠페인 데이터베이스 자원
  - b. 명령 데이터베이스 자원



- c. 통화 데이터베이스 자원
  - d. 정책 데이터베이스 자원
  - e. 운송 데이터베이스 자원
  - f. 세금 데이터베이스 자원
5. 기타 데이터베이스 자원 그룹.
- a. 카탈로그 데이터베이스 자원은 운송 및 세금 데이터베이스 자원 그룹에서 종속됩니다.
  - b. 상점 기본값 데이터베이스 자원은 운송 데이터베이스 자원 그룹에서 외부 종속성을 가집니다. 운송 데이터베이스 자원이 존재하지 않는 경우, 이 그룹은 대량 자료 반입될 필요가 없습니다.
  - c. 장기 구매 계약 데이터베이스 자원은 조직 자원에서 종속적입니다. 장기 구매 계약 데이터베이스 자원을 직접 로드할 수 없습니다. 자세한 정보는 325 페이지의 『장기 구매 계약 자원 공개』를 참조하십시오. 다른 데이터베이스 자원 그룹 이후에 장기 구매 계약을 로드해야 합니다.

377 페이지의 부록 E 『데이터베이스 자원 그룹』을 참조하여 WebSphere Commerce 견본 상점에 의해 형성된 데이터베이스 자원 그룹의 콘텐츠를 보십시오.

---

## 상점 로드

데이터베이스 자원 로드를 지원하기 위해 견본 패키지가 WebSphere Commerce 웹 사이트에서 사용 가능합니다. 이 패키지는 WebSphere Commerce 견본 상점을 기본으로 하며 아래 단계에서 발견된 파일을 포함합니다. 이 패키지를 다음에서 다운로드할 수 있습니다.

-  [http://www.ibm.com/software/webservers/commerce/wc\\_be/downloads.html](http://www.ibm.com/software/webservers/commerce/wc_be/downloads.html)
-  [http://www.ibm.com/software/webservers/commerce/wc\\_pe/downloads.html](http://www.ibm.com/software/webservers/commerce/wc_pe/downloads.html)

전체 상점에 대한 XML 데이터를 WebSphere Commerce 데이터베이스로 로드하려면, 다음을 수행하십시오.

1. 다음 정보를 검토하십시오.
  - a. 361 페이지의 부록 B 『데이터 작성』.
  - b. 377 페이지의 부록 E 『데이터베이스 자원 그룹』. 어떤 WebSphere Commerce 데이터베이스 자원 파일 및 데이터베이스 테이블이 영향을 받는지 알아야 할 필요가 있습니다.
  - c. 265 페이지의 제 27 장 『상점 데이터 로드 개요』. 로터 패키지에 대한 기본 정보를 제공합니다.

2. 상점 데이터베이스 자원의 완전한 세트를 위해 로드 프로세스를 계획하십시오. 316 페이지의 『데이터베이스 자원 그룹 로드』에서 소개된 단일 데이터베이스 자원 그룹을 로드하든지 전체 상점을 로드하든지 기본 프로세스는 동일합니다. 다음 단계에서 로드 프로세스에 대해 다음 파일을 사용하거나 작성합니다.

a. 각 그룹에 대한 하나 이상의 데이터베이스 자원 파일. 완전한 상점을 로드하는 경우, 작성된 데이터베이스 자원 파일 모두가 필요합니다. 예를 들어, *database asset.xml* 파일(*campaign.xml*, *catalog.xml* 또는 *currency.xml*) 및 사용자 상점 로케일 지원을 위해 별도의 로케일 지정 *database asset.xml* 파일이 필요합니다. 그러한 파일의 예제는 다음 디렉토리에서 WebSphere Commerce 견본 상점을 운송합니다.

-  *drive:\WebSphere\CommerceServer\samplestores\sample store name\data\*
-  *drive:\Program Files\WebSphere\CommerceServer\samplestores\sample store name\data\*
-  */usr/WebSphere/CommerceServer/samplestores/sample store name/data/*
-   */opt/WebSphere/CommerceServer/samplestores/sample store name/data/*
-  */QIBM/ProdData/WebCommerce/samplestores/sample store name/data/*

모든 데이터베이스 자원 그룹은 로케일 지정 정보가 필요합니다.







b. 상점에 대한 모든 XML 데이터베이스 자원 파일을 통합하는 새 XML 데이터베이스 파일은 XML 전체 참조 및 전체 상점에 대한 루트 요소를 포함합니다. 이것은 기본 데이터베이스 자원 그룹 XML 파일로서 참조됩니다. *store-data-assets.xml*라고 하는 견본 패키지에서 이 파일을 찾을 수 있습니다.

c. 데이터베이스 자원 그룹에서 XML 파일이 필요로하는 모든 데이터 유형을 정의하는 새 DTD 파일은 기본 데이터베이스 자원 그룹 DTD 파일로서 참조됩니다. *store-data-assets.dtd*라고 하는 견본 패키지에서 이 파일을 찾을 수 있습니다.

d. 외부 종속성을 정의하는 두 번째 DTD 파일. 기본 데이터베이스 자원 그룹 DTD 파일에 이 파일을 포함시킬 필요가 있습니다. *ForeignKeys.dtd*라고 하는 견본 패키지에서 이 파일을 찾을 수 있습니다.

e. 모든 WebSphere Commerce 테이블의 정의를 포함하는 세 번째 DTD 테이블. *wcs.dtd* 파일은 다음 디렉토리에서 로드된 WebSphere Commerce에 이미 있습니다.



-  NT `drive:\WebSphere\CommerceServer\schema\xml\`
-  2000 `drive:\Program Files\WebSphere\CommerceServer\schema\xml\`
-  AIX `/usr/WebSphere/CommerceServer/schema/xml/`
-  Solaris  Linux `/opt/WebSphere/CommerceServer/schema/xml/`
-  400 `/QIBM/ProdData/WebCommerce/schema/xml/`

기본 데이터베이스 자원 그룹 DTD 파일에 이 파일을 포함시킬 필요가 있습니다. WebSphere Commerce 스키마를 사용자 정의하지 않은 경우, 수정 없이 이 파일을 사용할 수 있습니다.

- 이 안내서에서 이전 장의 지시사항을 따라 데이터베이스 자원 XML 파일을 작성하십시오. 자원 장에서 이 태스크를 완료했을 경우, 이 XML 파일들은 이미 존재합니다. 데이터베이스 자원 파일은 파일 연결시 충돌이 발생하므로 어떤 DTD 선언 또는 파일의 시작 부분의 페이지 지시문도 포함하지 않아야 합니다. 또한 간결성을 위해 어떤 루트 요소도 작성하지 않도록 해야 합니다. 루트 요소를 가진 파일은 기본 데이터베이스 자원 그룹 XML 파일입니다.

주: 하나 이상의 언어에 대한 데이터베이스 자원 파일을 가지고 있는 경우, 각 파일은 `<?xml encoding = 로케일 고유의 인코딩>`으로 시작해야 합니다. 예를 들어, 영어 데이터베이스 자원 파일은 `<?xml encoding = "UTF-8"?`로 지정되어야 하고 프랑스 파일은 `<?xml encoding = "ISO-8859-1"?`로 지정되어야 합니다.

- 상점 데이터의 전체 세트에 대한 기본 데이터베이스 자원 그룹 XML 파일을 작성하십시오. 이 파일은 상점에 대한 다양한 데이터베이스 자원 XML 파일이 있는 참조 엔티티를 포함합니다. 외부 참조 엔티티를 사용하여 ID 분석 명령 및 로드 프로세스를 간결하게 하기 위해 XML 파일을 연결할 수 있습니다. 또한, 각 XML 파일 내에 사용된 내부 별명은 한번에 하나의 이상의 그룹을 로드할 때 그룹 내 또는 다른 그룹을 통한 다른 XML 파일에서는 외부 별명이 될 수 있습니다. XML 구문 분석기는 외부 참조 엔티티에 의해 참조되는 파일의 콘텐츠를 외부 참조 공간으로 대체합니다.

안내서와 같이 전체 상점 데이터 세트를 로드하기 위해 다음 예제를 사용하여 이 추출을 기반으로 데이터베이스 자원 그룹 파일을 작성할 수 있습니다.

```
<?xml version="1.0"?'>
<!DOCTYPE import SYSTEM "all-store-assets.dtd">
<import>
<!Fulfillment data group -->
&fulfillment.xml;

<!-- Store data group -->
&store.xml;
&en_US_store.xml;
```

```

&fr_FR_store.xml;

<!-- Tax data group -->
&tax.xml;
&en_US_tax.xml;
&fr_FR_tax.xml;
&taxfulfill.xml;

<!-- Shipping data group -->
&shipping.xml;
&en_US_shipping.xml;
&fr_FR_shipping.xml;
&shipfulfill.xml;

<!-- Catalog data group -->
&catalog.xml;
&en_US_catalog.xml;
&fr_FR_catalog.xml;
&storecatalog.xml;
&storefulfill.xml;
&offering.xml;
&store-catalog-tax.xml;
&store-catalog-shipping.xml;

<!-- Currency data group -->
¤cy.xml;
&en_US_currency.xml;
&fr_FR_currency.xml;

<!-- Campaign data group -->
&campaign.xml;
&en_US_campaign.xml;
&fr_FR_campaign.xml;

<!-- Business policy data group -->
&businesspolicy.xml;
&en_US_businesspolicy.xml;
&fr_FR_businesspolicy.xml;

<!-- Access control data group -->
&accesscontrol.xml;
&en_US_accesscontrol.xml;
&fr_FR_accesscontrol.xml;

<!-- Other data groups -->
&command.xml;
&store-default.xml;
</import>

```

여기서,

- Import은 XML 문서의 루트 요소입니다. 루트 요소는 WebSphere Commerce 에서 제공되는 wcs.dtd 파일에서 이미 정의되었고 WebSphere Commerce 데이터베이스 테이블 모두에 대한 정의를 포함합니다. 하지만 WebSphere

Commerce 스키마를 사용자 정의한 경우, 다른 루트 요소를 사용해야 합니다. 사용자 정의된 스키마에 영향을 주는 새 DTD 파일을 생성하거나 기존 wcs.dtd 파일을 갱신할 수 있습니다.

- all-store-assets.dtd는 다음 단계에서 작성하려는 기본 데이터베이스 자원 그룹 DTD 파일의 이름을 참조합니다.
- 설명 텍스트는 사용자의 상점에 대한 다른 데이터베이스 자원 그룹을 구분합니다.
- &database asset.xml;은 데이터베이스 자원 XML 파일에 대한 XML 엔티티 참조입니다. 이 경로와 위치는 데이터베이스 자원 그룹 DTD 파일에 정의되어 있습니다. 이 이름은 각 그룹에 대해 이미 작성된 데이터베이스 자원 파일을 충족시키기 위해 변경할 것입니다.
- &locale\_database asset.xml은 상점이 지원하는 각 언어에 대하여 필요합니다. 상점이 단일 언어만 사용한다면 하나의 파일만을 참조합니다. 상점이 하나 이상의 언어를 지원할 경우, 각 언어에 대한 참조가 필요합니다. 위의 추출은 상점이 영어와 프랑스어를 지원함을 나타냅니다.

5. 위의 엔티티를 정의하는 기본 데이터베이스 자원 그룹 DTD 파일 및 데이터베이스 자원에 필요한 다른 DTD 파일을 작성하십시오.

안내서와 같이 상점 데이터베이스 자원의 전체 세트에 대해 다음 예제를 사용하여 기본 데이터베이스 자원 그룹 DTD 파일을 작성할 수 있습니다.

```
<!ENTITY % wcs.dtd SYSTEM "absolute path for WebSphere Commerce wcs.dtd file">
%wcs.dtd;

<!ENTITY %NonStoreForeignKeys.dtd SYSTEM "NonStoreForeignKeys.dtd">
%NonStoreForeignKeys.dtd;
<!ENTITY fulfillment.xml SYSTEM "data/fulfillment.xml">
<!ENTITY en_US_fulfillment.xml SYSTEM "data/en_US/fulfillment.xml">
<!ENTITY fr_FR_fulfillment.xml SYSTEM "data/fr_FR/fulfillment.xml">

<!ENTITY store.xml SYSTEM "data/store.xml">
<!ENTITY en_US_store.xml SYSTEM "data/en_US/store.xml">
<!ENTITY fr_FR_store.xml SYSTEM "data/fr_FR/store.xml">

<!ENTITY tax.xml SYSTEM "data/tax.xml">
<!ENTITY en_US_tax.xml SYSTEM "data/en_US/tax.xml">
<!ENTITY fr_FR_tax.xml SYSTEM "data/fr_FR/tax.xml">
<!ENTITY taxfulfill.xml SYSTEM "data/taxfulfill.xml">

<!ENTITY shipping.xml SYSTEM "data/shipping.xml">
<!ENTITY en_US_shipping.xml SYSTEM "data/en_US/shipping.xml">
<!ENTITY fr_FR_shipping.xml SYSTEM "data/fr_FR/shipping.xml">
<!ENTITY shipfulfill.xml SYSTEM "data/shipfulfill.xml">

<!ENTITY catalog.xml SYSTEM "data/catalog.xml">
<!ENTITY en_US_catalog.xml SYSTEM "data/en_US/catalog.xml">
<!ENTITY fr_FR_catalog.xml SYSTEM "data/fr_FR/catalog.xml">
<!ENTITY store-catalog.xml SYSTEM "data/store-catalog.xml">
<!ENTITY storefulfill.xml SYSTEM "data/storefulfill.xml">
<!ENTITY offering.xml SYSTEM "data/offering.xml">
<!ENTITY store-catalog-tax.xml SYSTEM "data/store-catalog-tax.xml">
<!ENTITY store-catalog-shipping.xml SYSTEM "data/store-catalog-shipping.xml">

<!ENTITY currency.xml SYSTEM "data/currency.xml">
<!ENTITY en_US_currency.xml SYSTEM "data/en_US/currency.xml">
```

```

<!ENTITY fr_FR_currency.xml SYSTEM "data/fr_FR/currency.xml">

<!ENTITY campaign.xml SYSTEM "data/campaign.xml">
<!ENTITY en_US_campaign.xml SYSTEM "data/en_US/campaign.xml">
<!ENTITY fr_FR_campaign.xml SYSTEM "data/fr_FR/campaign.xml">

<!ENTITY businesspolicy.xml SYSTEM "data/businesspolicy.xml">
<!ENTITY en_US_businesspolicy.xml SYSTEM "data/en_US/businesspolicy.xml">
<!ENTITY fr_FR_businesspolicy.xml SYSTEM "data/fr_FR/businesspolicy.xml">

<!ENTITY accesscontrol.xml SYSTEM "data/accesscontrol.xml">
<!ENTITY en_US_accesscontrol.xml SYSTEM "data/en_US/accesscontrol.xml">
<!ENTITY fr_FR_accesscontrol.xml SYSTEM "data/fr_FR/accesscontrol.xml">

<!ENTITY command.xml SYSTEM "data/command.xml">
<!ENTITY store-defaults.xml SYSTEM "data/store-defaults.xml">

```

여기서,

- `wcs.dtd`는 데이터베이스 자원 그룹 외부에서 정의된 데이터가 있는 DTD 파일을 참조합니다. WebSphere Commerce와 함께 제공된 이 파일은 데이터베이스 자원 그룹 XML 파일에 사용된 루트 요소도 정의합니다.
- `NonStoreForeignKeys.dtd`는 루트 요소를 제외한 요소를 정의하는 DTD 파일을 참조하십시오. 이 파일은 데이터베이스 자원 그룹 밖의 외부 종속성에 대한 모든 XML 엔티티 참조 선언 및 정의를 포함합니다. 이와 같이 XML 파일은 데이터베이스 자원 그룹의 일부로서 작성되지 않는 foreign key 값으로의 참조를 가지며 이 그룹 이전에 데이터베이스로 이미 로드되어 있어야 합니다.

주: 경로가 올바르게 정의되었는지 확인하십시오. 이 예제에서 파일은 기본 데이터베이스 자원 그룹 DTD 파일과 동일한 디렉토리에 있습니다.

- `store.xml`, `en_US_store.xml` 및 `fr_FR_store.xml` 파일은 기본 데이터베이스 자원 그룹 XML 파일에 사용된 외부 참조 엔티티이며, 사용자의 상점이 영어와 불어를 지원한다고 가정합니다. 이 참조를 사용하려면, `&alias_name;` 엔티티 참조 규칙을 따르십시오.
- `database asset.xml`은 데이터베이스 자원이 로드되는 XML 파일의 이름을 참조합니다. 이 이름은 각 그룹에 대해 이미 작성된 데이터베이스 자원 파일을 충족시키기 위해 변경합니다. 로케일 지정 XML 파일은 다음 디렉토리 아래에 있습니다.




```

- ▶ NT drive:\WebSphere\CommerceServer\samplstores\
 sample store name\data\locale\

- ▶ 2000 drive:\Program
 Files\WebSphere\CommerceServer\samplstores\
 sample store name\data\locale\

- ▶ AIX /usr/WebSphere/CommerceServer/samplstores/
 sample store name/data/locale/

```

-   /opt/WebSphere/CommerceServer/samplestores/  
sample store name/data/locale/
-  /QIBM/ProdData/WebCommerce/samplestores/  
sample store name/data/locale/

- *path\_database asset.xml* 파일은 사용자 상점을 지원하는 각 언어에 대해 필요하며 위의 디렉토리 아래에 있습니다. 상점이 단일 언어만 사용한다면 하나의 파일만을 참조합니다. 상점이 하나 이상의 언어를 지원할 경우, 각 언어에 대한 로케일 고유의 파일이 필요합니다. 위의 추출은 상점이 영어와 프랑스어를 지원함을 나타냅니다.

6. 각 그룹이 외부 종속성을 가지고 있는 때, 각 데이터베이스 자원 그룹에는 도메인 또는 데이터 세트 외부에서 정의된 정보가 필요합니다. 이 데이터를 DTD 파일로 제공할 수 있습니다. 예를 들어, 상점 데이터베이스 자원 그룹에는 다음 외부 종속성이 있습니다.

```
bootstrap.LANGUAGE.LANGUAGE_ID, bootstrap.MEMBER.MEMBER_ID,
bootstrap.SETCURR.SETCURR_ID, fulfillment.FFMCENTER.FFMCENTER_ID
```

데이터베이스 자원 그룹 또는 상점 자원의 전체 세트를 로드할 때, 외부 종속성은 WebSphere Commerce 데이터베이스에서 정의되어야 합니다. 이 데이터를 사용하려면, 관련된 XML 엔티티 참조를 따르십시오. 예를 들어, *ffmcenter\_id* 엔티티에 의해 정의된 데이터를 사용하려면, XML 파일에 *&ffmcenter\_id*;를 작성해야 합니다. 안내서의 상점 데이터베이스 자원에 대한 다음 예를 사용하여 *NonStoreForeignKeys.dtd*라고 하는 이 추출에 기반을 두는 DTD 파일을 작성할 수 있습니다.

```
<!ENTITY ko_KR "-1">
<!ENTITY fr_FR "-2">
<!ENTITY de_DE "-3">
<!ENTITY it_IT "-4">
<!ENTITY es_ES "-5">
<!ENTITY pt_BR "-6">
<!ENTITY zh_CN "-7">
<!ENTITY zh_TW "-8">
<!ENTITY ko_KR "-9">
<!ENTITY ja_JP "-10">
<!ENTITY MEMBER_ID "-2000">
<!ENTITY ffmcenter_id "10001">
```



여기서,

- *MEMBER\_ID*는 상점의 소유자를 식별하는 내부 참조 번호입니다.
- *ffmcenter*는 상점 서비스 센터에 대한 참조 번호입니다. 상점이 하나 이상의 서비스 센터를 사용하기 때문에, 하나 이상의 서비스 센터가 *NonStoreForeignKeys.dtd* 파일에 정의될 수 있습니다.


- *locale*은 각 로케일(국가/지역 및 언어로 식별됨)에 대한 WebSphere Commerce 참조 번호입니다. 값을 LANGUAGE 데이터베이스 테이블에서 찾을 수 있습니다.


주: 기존의 상점 아카이브를 데이터베이스 자원 그룹으로 나누는 경우, 모든 참조 (예:@ffmcenter\_id 별명)가 해당 엔티티 참조 (예:&ffmcenter\_id;)로 교체되는지 확인하십시오.

7. 일단, 모든 데이터 파일이 작성되면 기본 데이터베이스 자원 그룹 XML 파일에 대한 IDResolve 명령을 실행하여 269 페이지의 『ID Resolve 명령』에서 설명된 것처럼 데이터를 분석하십시오.
8. 278 페이지의 『load 명령』 설명처럼 분석된 데이터 파일의 load 명령을 실행하십시오. 로드 프로세스를 검증하려면 로그 파일을 참조하십시오.

-  idresgen.db2.log 및 massload.db2.log
-  idresgen.oracle.log 및 massload.oracle.log


로그 파일은 다음 디렉토리에 있습니다.


-  drive:\WebSphere\CommerceServer\logs\
-  drive:\Program Files\WebSphere\CommerceServer\logs\
-  /usr/WebSphere/CommerceServer/logs/
-   /opt/WebSphere/CommerceServer/logs/
-  /QIBM/ProdData/WebCommerce/logs/

9.  324 페이지의 『비즈니스 계정 자원 공개』에 설명된 AccountImport 명령을 실행하십시오.
10. 적용 가능한 경우, 325 페이지의 『장기 구매 계약 자원 공개』에 설명된 장기 구매 계약을 공개하십시오.
11. 328 페이지의 『WebSphere Commerce 서버에 복사하여 상점 입구 자원 및 상점 구성 파일 공개』 태스크를 완료하십시오.

## 데이터베이스 자원 그룹 로드







데이터베이스 자원 로드를 지원하기 위해 견본 패키지가 WebSphere Commerce 웹 사이트에서 사용 가능합니다. 이 패키지는 WebSphere Commerce 견본 상점을 기본으로 하며 아래 단계에서 발견된 파일을 포함합니다. 이 패키지를 다음에서 다운로드할 수 있습니다.

-  [http://www.ibm.com/software/webservers/commerce/wc\\_be/downloads.html](http://www.ibm.com/software/webservers/commerce/wc_be/downloads.html)

-  [http://www.ibm.com/software/webservers/commerce/wc\\_pe/downloads.html](http://www.ibm.com/software/webservers/commerce/wc_pe/downloads.html)

단일 데이터베이스 자원 그룹에 대한 XML 데이터를 WebSphere Commerce 데이터베이스로 로드하려면, 다음을 수행하십시오.







1. 다음 정보를 검토하십시오.
  - a. 361 페이지의 부록 B 『데이터 작성』
  - b. 377 페이지의 부록 E 『데이터베이스 자원 그룹』. 어떤 WebSphere Commerce 자원 파일 및 데이터베이스 테이블이 영향을 받는지 알아야 할 필요가 있습니다.
  - c. 265 페이지의 제 27 장 『상점 데이터 로드 개요』. 로터 패키지에 대한 기본 정보를 제공합니다.
2. 로드 프로세스를 계획하고 로드할 데이터베이스 자원 그룹을 결정하십시오. 309 페이지의 『상점 로드』에서 소개된 것과 같이 상점 데이터베이스 자원의 전체 세트를 로드할 하든지 단일 데이터베이스 자원 그룹을 로드하든지 기본 처리는 동일합니다. 다음 단계에서 로드 프로세스에 대해 다음 파일을 사용하거나 작성합니다.
  - a. 어떤 그룹을 선택하는 지에 따라 하나 이상의 데이터베이스 자원 파일. 예를 들어, 상점 데이터베이스 그룹 자원을 로드하는 경우, 상점이 지원하는 각 로케일에 대하여 store.xml 파일 및 분리 store.xml 파일이 필요합니다. 그러한 파일의 예제는 다음 디렉토리에서 WebSphere Commerce 견본 상점을 발행합니다.

-  `drive:\WebSphere\CommerceServer\samplstores\  
sample store name\data`
-  `drive:\Program Files\WebSphere\CommerceServer\  
\samplstores\sample store name\data`
-  `/usr/WebSphere/CommerceServer/samplstores/  
sample store name/data`
-   `/opt/WebSphere/CommerceServer/samplstores/  
sample store name/data`
-  `/QIBM/ProdData/WebCommerce/samplstores/  
sample store name/data`

모든 데이터베이스 자원 그룹은 로케일 지정 정보가 필요합니다.

- b. 모든 XML 데이터베이스 자원 파일을 통합하는 새 XML 파일은 XML 전체 참조 및 데이터베이스 자원의 루트 요소를 포함합니다. 이것은 기본 데이터베이스 자원 그룹 XML 파일로서 참조됩니다. store-all-assets.xml라고 하는 견본 패키지에서 이 파일을 찾을 수 있습니다.

- c. 데이터베이스 자원 그룹에서 XML 파일이 필요로하는 모든 데이터 유형을 정의하는 새 DTD 파일은 기본 데이터베이스 자원 그룹 DTD 파일로서 참조됩니다. store-all-assets.dtd라고 하는 기본 패키지에서 이 파일을 찾을 수 있습니다.
- d. 외부 종속성을 정의하는 두 번째 DTD 파일. 기본 데이터베이스 자원 그룹 DTD 파일에 이 파일을 포함시킬 필요가 있습니다. ForeignKeys.dtd라고 하는 기본 패키지에서 이 파일을 찾을 수 있습니다.
- e. 모든 WebSphere Commerce 테이블의 정의를 포함하는 세 번째 DTD 테이블. wcs.dtd 파일은 다음 디렉토리에서 로드된 WebSphere Commerce에 이미 있습니다.

-  NT drive:\WebSphere\CommerceServer\schema\xml\
-  2000 drive:\Program Files\WebSphere\CommerceServer\schema\xml\
-  AIX /usr/WebSphere/CommerceServer/schema/xml/
-  Solaris  Linux /opt/WebSphere/CommerceServer/schema/xml/
-  400 /QIBM/ProdData/WebCommerce/schema/xml/

기본 데이터베이스 자원 그룹 DTD 파일에 이 파일을 포함시킬 필요가 있습니다. WebSphere Commerce 스키마를 사용자 정의하지 않은 경우, 수정 없이 이 파일을 사용할 수 있습니다.

3. 이 안내서에서 이전 장의 지시사항을 따라 로드할 그룹에 대한 데이터베이스 자원 XML 파일을 작성하십시오. 자원 장에서 이 태스크를 완료했을 경우, 이 XML 파일들은 이미 존재합니다. 데이터베이스 자원 파일은 파일 연결시 충돌이 발생하므로 어떤 DTD 선언 또는 파일의 시작 부분의 페이지 지시문도 포함하지 않아야 합니다. 또한 간결성을 위해 어떤 루트 요소도 작성하지 않도록 해야 합니다. 루트 요소를 가진 파일은 기본 데이터베이스 자원 그룹 XML 파일입니다.

주: 하나 이상의 언어에 대한 데이터베이스 자원 파일을 가지고 있는 경우, 각 파일은 `<?xml encoding = 로케일 고유의 인코딩>`으로 시작해야 합니다. 예를 들어, 영어 데이터베이스 자원 파일은 `<?xml encoding = "UTF-8"?>`로 지정되어야 하고 프랑스 파일은 `<?xml encoding = "ISO-8859-1"?>`로 지정되어야 합니다.

4. 로드하려는 각 그룹에 대해 기본 데이터베이스 자원 그룹 XML 파일을 작성하십시오. 이 파일은 하나 이상의 데이터베이스 자원 그룹의 다양한 XML 파일을 포함하도록 참조 엔티티를 포함합니다. 외부 참조 엔티티를 사용하여 ID 분석 명령 및 로드 프로세스를 간결하게 하기 위해 XML 파일을 연결할 수 있습니다. 또한, 각 XML 파일 내에 사용된 내부 별명은 한번에 하나의 이상의 그룹을 로드할 때



그룹 내 또는 그룹을 통한 다른 XML 데이터 파일에서는 외부 별명이 될 수 있습니다. 구문 분석기는 외부 참조 엔티티에 의해 참조되는 파일의 콘텐츠를 외부 참조 공간으로 대체합니다.

안내서와 같이 단일 상점 데이터베이스 자원 그룹을 로드하기 위해 다음 예제를 사용하여 이 추출을 기반으로 데이터베이스 자원 그룹 XML 파일을 작성할 수 있습니다.

```
<?xml version="1.0"?>
<!DOCTYPE import SYSTEM "store-assets.dtd">
<import>
&store.xml;
&en_US_store.xml;
&fr_FR_store.xml;
</import>
```

여기서,

- Import은 XML 문서의 루트 요소입니다. 루트 요소는 WebSphere Commerce 에서 제공되는 wcs.dtd 파일에서 이미 정의되었고 WebSphere Commerce 데이터베이스 테이블 모두에 대한 정의를 포함합니다. 하지만 WebSphere Commerce 스키마를 사용자 정의한 경우, 다른 루트 요소를 사용해야 합니다. 사용자 정의된 스키마에 영향을 주는 새 DTD 파일을 생성하거나 wcs.dtd 파일을 갱신할 수 있습니다.
  - store-assets.dtd는 다음 단계에서 작성하려는 기본 데이터베이스 자원 그룹 DTD 파일의 이름을 참조합니다
  - &store.xml;은 데이터베이스 자원 그룹 XML 파일로의 XML 엔티티 참조입니다. 이 경로와 위치는 데이터베이스 자원 그룹 DTD 파일에 정의되어 있습니다. 이 이름은 각 그룹에 대해 이미 작성된 자원 파일을 충족시키기 위해 변경합니다.
  - locale\_store.xml은 상점이 지원하는 각 언어에 대하여 필요합니다. 상점이 단일 언어만 사용한다면 하나의 파일만을 참조합니다. 상점이 하나 이상의 언어를 지원할 경우, 각 언어에 대한 참조가 필요합니다. 위의 추출은 상점이 영어와 프랑스어를 지원함을 나타냅니다.
5. 위의 엔티티를 정의하는 기본 데이터베이스 자원 그룹 DTD 파일 및 이 그룹에 필요한 다른 DTD 파일을 작성하십시오.

안내서와 같이 단일 상점 데이터베이스 자원 그룹을 로드하기 위해 다음 예를 사용하여 모든 데이터 그룹에 대해 기본 데이터베이스 자원 그룹을 작성할 수 있습니다.

```
<!ENTITY % wcs.dtd SYSTEM "absolute path for WebSphere Commerce wcs.dtd file">
%wcs.dtd;

<!ENTITY %NonStoreForeignKeys.dtd SYSTEM "NonStoreForeignKeys.dtd">
%NonStoreForeignKeys.dtd;
```







```
<!ENTITY store.xml SYSTEM "store.xml">
<!ENTITY en_US_store.xml SYSTEM "en_US/store.xml">
<!ENTITY fr_FR_store.xml SYSTEM "fr_FR/store.xml">
```

여기서,

- `wcs.dtd`는 데이터베이스 자원 그룹 외부에서 정의된 데이터가 있는 DTD 파일을 참조합니다. WebSphere Commerce와 함께 제공된 이 파일은 데이터베이스 자원 그룹 XML 파일에 사용된 루트 요소도 분석 및 정의합니다.
- `NonStoreForeignKeys.dtd`는 루트 요소를 제외한 요소를 정의하는 DTD 파일을 참조하십시오. 이 파일은 데이터베이스 자원 그룹 밖의 외부 종속성에 대한 모든 XML 엔티티 참조 선언 및 정의를 포함합니다. 이와 같이 XML 파일은 데이터베이스 자원 그룹의 일부로서 작성되지 않는 foreign key 값으로의 참조를 가지며 이 그룹 이전에 데이터베이스로 이미 로드되어 있어야 합니다.

주: 경로가 올바르게 정의되었는지 확인하십시오. 이 예제에서 파일은 데이터베이스 자원 그룹 DTD 파일과 동일한 디렉토리에 있습니다.

- `store.xml`, `en_US_store.xml` 및 `fr_FR_store.xml` 파일은 데이터베이스 자원 그룹 XML 파일에 사용된 외부 참조 엔티티입니다. 이 참조를 사용하려면, `&alias_name`; 엔티티 참조 규칙을 따르십시오.
- `store.xml` 데이터베이스 자원이 로드된 그룹에 대한 데이터 파일을 참조합니다. 이 이름은 각 그룹에 대해 이미 작성된 데이터베이스 자원 파일을 충족시키기 위해 변경합니다. 로케일 지정 XML 파일은 다음 디렉토리 아래에 있습니다.

-  `drive:\WebSphere\CommerceServer\samplstores\sample store name\data\locale\`
-  `drive:\Program Files\WebSphere\CommerceServer\samplstores\sample store name\data\locale\`
-  `/usr/WebSphere/CommerceServer/samplstores/sample store name/data/locale/`
-   `/opt/WebSphere/CommerceServer/samplstores/sample store name/data/locale/`
-  `/QIBM/ProdData/WebCommerce/samplstores/sample store name/data/locale/`

- `path_store.xml`은 상점이 지원하는 각 언어에 대하여 필요하며 위 디렉토리 아래에 있습니다. 상점이 단일 언어만 사용한다면 하나의 파일만을 참조합니다. 상점이 하나 이상의 언어를 지원할 경우, 각 언어에 대한 로케일 고유의 파일이 필요합니다. 위의 추출은 상점이 영어와 프랑스어를 지원함을 나타냅니다.

6. 각 그룹이 외부 종속성을 가지고 있는 때, 각 데이터베이스 자원 그룹에는 도메인 또는 데이터 세트 외부에서 정의된 정보가 필요합니다. 이 데이터를 DTD 파일로 제공할 수 있습니다. 예를 들어, 상점 데이터베이스 자원 그룹에는 다음 외부 종속성이 있습니다.

```
bootstrap.LANGUAGE.LANGUAGE_ID, bootstrap.MEMBER.MEMBER_ID,
bootstrap.SETCURR.SETCURR_ID, fulfillment.FFMCENTER.FFMCENTER_ID
```

데이터 그룹 또는 상점 데이터의 전체 세트를 로드할 때, 다음 외부 종속성은 WebSphere Commerce 데이터베이스에서 정의되어야 합니다. 이 데이터를 사용하려면, 관련된 XML 엔티티 참조를 따르십시오. 예를 들어, `ffmcenter_id` 엔티티에 의해 정의된 데이터를 사용하려면, XML 파일에 `&ffmcenter_id;`를 작성해야 합니다. 안내서의 상점 데이터베이스 자원에 대한 다음 예를 사용하여 `NonStoreForeignKeys.dtd`라고 하는 이 추출에 기반을 두는 DTD 파일을 작성할 수 있습니다.



```
<!ENTITY ko_KR "-1">
<!ENTITY fr_FR "-2">
<!ENTITY de_DE "-3">
<!ENTITY it_IT "-4">
<!ENTITY es_ES "-5">
<!ENTITY pt_BR "-6">
<!ENTITY zh_CN "-7">
<!ENTITY zh_TW "-8">
<!ENTITY ko_KR "-9">
<!ENTITY ja_JP "-10">
<!ENTITY MEMBER_ID "-2000">
<!ENTITY ffmcenter_id "10001">
```

여기서,







- `MEMBER_ID`는 상점의 소유자를 식별하는 내부 참조 번호입니다.
- `ffmcenter`는 상점 서비스 센터에 대한 참조 번호입니다. 상점이 하나 이상의 서비스 센터를 사용하기 때문에, 하나 이상의 서비스 센터가 `NonStoreForeignKeys.dtd` 파일에 정의될 수 있습니다.
- `locale`은 각 로케일(국가/지역 및 언어로 식별됨)에 대한 WebSphere Commerce 참조 번호입니다. 값을 `LANGUAGE` 데이터베이스 테이블에서 찾을 수 있습니다.

주: 기존의 상점 아카이브를 데이터베이스 자원 그룹으로 나누는 경우, 모든 참조 (예: `@ffmcenter_id` 별명)가 해당 엔티티 참조 (예: `&ffmcenter_id;`)로 교체되는지 확인하십시오.

7. 일단 모든 데이터 파일이 작성되면 데이터베이스 자원 그룹 XML 파일에 대한 `IDResolve` 명령을 실행하여 269 페이지의 『ID Resolve 명령』에서 설명된 것처럼 데이터를 분석하십시오.
8. 278 페이지의 『load 명령』 설명처럼 분석된 데이터 파일의 `load` 명령을 실행하십시오. 로드 프로세스를 검증하려면 로그 파일을 참조하십시오.

-  idresgen.db2.log 및 massload.db2.log
-  idresgen.oracle.log 및 massload.oracle.log

로그 파일은 다음 디렉토리에 있습니다.

-  `drive:\WebSphere\CommerceServer\logs\`
  -  `drive:\Program Files\WebSphere\CommerceServer\logs\`
  -  `/usr/WebSphere/CommerceServer/logs/`
  -   `/opt/WebSphere/CommerceServer/logs/`
  -  `/QIBM/ProdData/WebCommerce/logs/`
9.  324 페이지의 『비즈니스 계정 자원 공개』에 설명된 AccountImport 명령을 실행하십시오.
  10. 적용 가능한 경우, 325 페이지의 『장기 구매 계약 자원 공개』에 설명된 장기 구매 계약을 공개하십시오.
  11. 328 페이지의 『WebSphere Commerce 서버에 복사하여 상점 입구 자원 및 상점 구성 파일 공개』 태스크를 완료하십시오.

---

## 제 29 장 비즈니스 계정 및 장기 구매 계약 공개

일부 상점 데이터베이스 자원(비즈니스 계정 및 장기 구매 계약)을 로더 패키지를 사용하여 로드할 수 없습니다. 249 페이지의 제 26 장 『완성된 상점 공개』에 설명된 완성된 상점 옵션 공개의 부분으로 상점 서비스 또는 명령행을 사용하여 이 데이터베이스 자원을 공개하거나 관련 명령을 사용하여 비즈니스 계정, 장기 구매 계약을 공개할 수 있습니다. 이 명령은 다음과 같습니다.

- AccountImport -- 상점 아카이브의 businessaccount.xml 파일에서 비즈니스 계정을 작성합니다.
- ContractImportApprovedVersion -- contract.xml 파일에서 장기 구매 계약을 작성합니다. 장기 구매 계약이 활성화 상태인 경우, 명령이 작성되고 장기 구매 계약이 전개됩니다. contract.xml 파일이 하나 이상의 장기 구매 계약을 포함하고 있을지라도 명령은 한번만 호출될 필요가 있습니다.
- ProductSetPublish -- 비즈니스 계정 및 장기 구매 계약이 작성되기 전에 카탈로그로 상품 세트 데이터베이스 테이블의 상품 세트 데이터를 동기화합니다. 상점 서비스 및 명령행 공개는 AccountImport 및 ContractImportApprovedVersion 명령을 호출하는 ProductSetPublish 명령을 호출합니다.

주: 이 명령에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오.

비즈니스 계정 자원은 WebSphere Commerce로 제공된 일부 견본 상점 아카이브의 XML 파일의 양식에 포함됩니다. 하지만, 이 자원에 대한 XML 파일 작성보다는 제공된 도구를 사용하여 비즈니스 계정 자원을 작성할 것을 권장합니다. 제공된 도구를 사용하여 이 자원 작성에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오. 견본 상점 아카이브를 사용하여 제공된 관련 XML 파일 공개를 선택하는 경우 또는 사용자를 작성하는 경우, 비즈니스 계정 공개에 대한 지시사항은 다음 절에 포함됩니다.

주: 비즈니스 계정 또는 장기 구매 계약을 공개하기 위해 상점 서비스를 사용하지 않는 경우, 상점 및 카탈로그 자원이 비즈니스 계정 또는 장기 구매 계약을 공개하기 전에 공개되어야 합니다. 특히 상점을 소유한 조직의 ID와 장기 구매 계약과 연관된 모든 구매자 조직의 ID뿐 아니라 상점 및 카탈로그 식별자가 필요합니다. 장기 구매 계약의 규정이 특정 카탈로그를 지정하지 않은 경우, 비즈니스 계정 또는 장기 구매 계약을 공개하기 전에 카탈로그를 공개할 필요가 없습니다.

상점 서비스 또는 명령행 공개를 사용하여 이 자원을 공개하는 경우, 카탈로그 옵션을 선택했는지 또는 사용자 자원이 공개된 카탈로그를 가지고 있는지 확인하십시오.

오. 관련 명령을 사용하여 이 자원들을 공개하려는 경우, 데이터베이스로 위에 나열된 자원들이 이미 로드되었는지 확인하십시오.

---

## 상점 서비스 또는 명령행을 사용하여 비즈니스 계정 및 장기 구매 계약 공개

상점 서비스 또는 명령행에서 공개 유틸리티를 사용하여 비즈니스 계정 및 장기 구매 계약을 공개할 수 있습니다. 상점 서비스나 명령행을 사용하여 비즈니스 계정 및 장기 구매 계약을 공개하려면, 해당 자원이 상점 아카이브 포맷에 패키징되어 있어야 합니다. 상점 아카이브로 상점 입구 자원을 구성하는 것에 대한 자세한 정보는 237 페이지의 제 6 부 『상점 패키지』를 참조하십시오.

상점 서비스 또는 명령행 공개를 사용하려는 경우, 상점 아카이브(상점 입구 자원, 상점 데이터 자원 및 자원 번들 포함)에서 모든 유형의 자원의 공개를 선택할 수 있거나 단지 하나의 자원 유형의 공개를 선택할 수 있습니다. 상점 서비스 또는 명령행을 사용하여 자원을 공개하기 위한 자세한 단계별 지시사항에 대해서는 WebSphere Commerce 온라인 도움말을 참조하십시오.

---


## 명령을 사용하여 비즈니스 계정 및 장기 구매 계약 공개

자원 아카이브로 자원을 패키징하지 않으려는 경우, 관련 명령을 사용하여 비즈니스 계정 및 장기 구매 계약을 공개할 수 있습니다.

- AccountImport -- 상점 아카이브의 businessaccount.xml 파일에서 비즈니스 계정을 작성합니다.
- ContractImportApprovedVersion -- XML 파일에서 WebSphere Commerce 서버로 승인되거나 활성중인 장기 구매 계약을 반입합니다. 장기 구매 계약을 반입하기 전에, 명령은 반입되는 장기 구매 계약이 필요한 규정을 포함하며 올바른 장기 구매 계약인지를 확인합니다.
- ProductSetPublish -- 비즈니스 계정 및 장기 구매 계약이 작성되기 전에 카탈로그로 상품 세트 데이터베이스 테이블의 상품 세트 데이터를 동기화합니다. 상점 서비스 및 명령행 공개는 AccountImport 및 ContractImportApprovedVersion 명령을 호출하는 ProductSetPublish 명령을 호출합니다.

### 비즈니스 계정 자원 공개

비즈니스 계정 자원을 공개하려면, 다음을 수행하십시오.

1. 관리 콘솔을 사용하여 보기 레지스트리를 최신 정보로 갱신하거나 WebSphere Commerce 인스턴스를 다시 시작하십시오. 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오.
2. businessaccount.xml 파일을 다음 디렉토리로 복사하십시오.
  -  drive:\WebSphere\CommerceServer\xml\trading

- ▶ 2000 drive:\Program Files\WebSphere\CommerceServer\xml\trading
  - ▶ AIX /usr/WebSphere/CommereServer/xml/trading
  - ▶ Solaris /opt/WebSphere/CommerceServer/xml/trading
  - ▶ Linux /opt/WebSphere/CommerceServer/xml/trading
  - ▶ 400 /QIBM/UserData/WebCommerce/instances/instancename/xml/trading
3. businessaccount.xml 파일을 열고 다음 변경사항을 수행하십시오.
- &STORE\_IDENTIFIER;의 모든 발생을 상점에 대한 상점 식별자로 교체하십시오.
  - &MEMBER\_IDENTIFIER;의 모든 발생을 상점에 대한 구성원 인식 이름으로 교체하십시오.
- 주: 상점 서비스를 사용하여 작성된 상점 아카이브의 부분인 businessaccount.xml 로 작업중인 경우, 이 단계는 이미 완료됩니다.
4. 저장하고 파일을 종료하십시오.
5. 관리 콘솔을 여십시오. 관리자로 로그인하십시오.
6. 브라우저에서 다음을 입력하십시오.
- `https://hostname:8000/webapp/wcs/stores/servlet/AccountImport?fileName=businessaccount.xml`  
&URL=성공적인 완료시 경로 재지정하기 위한 URL

주: 명령 구문 및 매개변수에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오.


## 장기 구매 계약 자원 공개

장기 구매 계약 자원을 공개하려면, 다음을 수행하십시오.

1. contract.xml 파일을 다음 디렉토리로 복사하십시오.

주: 이 경로는 구성 가능합니다. 다음 경로가 기본값입니다.

- ▶ NT drive:\WebSphere\CommerceServer\xml\trading
- ▶ 2000 drive:\Program Files\WebSphere\CommerceServer\xml\trading
- ▶ AIX /usr/WebSphere/CommerceServer/xml/trading
- ▶ Solaris /opt/WebSphere/CommerceServer/xml/trading
- ▶ Linux /opt/WebSphere/CommerceServer/xml/trading

-  400 /QIBM/UserData/WebCommerce/instances/*instancename*/xml/trading
2. contract.xml 파일을 열고 다음 변경사항을 수행하십시오.
    - &STORE\_IDENTIFIER;의 모든 발생을 상점에 대한 상점 식별자로 교체하십시오.
    - &MEMBER\_IDENTIFIER;의 모든 발생을 상점에 대한 구성원 인식 이름으로 교체하십시오.
  3. 저장하고 파일을 종료하십시오.
  4. 관리 콘솔을 여십시오. 관리자로 로그인하십시오.
  5. 브라우저에서 다음을 입력하십시오.
    - `https://hostname:portnumber/webapp/wcs/tools/servlet/ContractImportApprovedVersion?fileName=contract.xml&targetStoreId=store_id&URL=ContractDisplay`
  6. 상점이 복수 contract.xml 파일(예: 로케일 지정 장기 구매 계약 파일)을 포함하는 경우, 각 contract.xml 파일에 대해 1 - 5단계를 반복하십시오.



---

## 제 30 장 상점 입구 자원 및 상점 구성 파일 공개

상점 입구 자원, HTML 및 JSP 파일, 특성 파일 또는 자원 번들 그리고 상점 페이지를 작성하는 이미지 및 그래픽 공개는 기능하는 상점 작성 프로세스의 부분입니다. 249 페이지의 제 26 장 『완성된 상점 공개』에 설명된 완성된 상점 옵션 공개의 부분으로, 상점 서비스 또는 명령행을 사용하여 이 상점 입구 자원을 공개하거나 WebSphere Commerce 서버에서 지정된 위치로 자원을 단순히 복사하여 상점 입구 자원을 공개할 수 있습니다.

견본 상점 **Business** ToolTech 및 NewFashion에 포함된 JSP 파일을 공개하고 상점을 구성하고 협업 기능을 사용하려는 경우, 해당 상점 아카이브의 부분인 상점 구성 파일을 공개해야 합니다. 두 개의 상점에는 다음 상점 구성 파일이 있습니다.

- tools\_properties.zip
- tools\_xml.zip
- runtime\_xml.zip

상점 서비스를 사용하여 또는 명령행(모든 공개 옵션 선택하여)에서 완성된 상점을 공개하는 경우, 상점 구성 파일 또한 공개됩니다. 하지만 상점 입구 자원을 WebSphere Commerce 서버에 복사함으로써 공개하려는 경우, 상점 구성 파일 또한 WebSphere Commerce 서버에 복사해야 합니다.

---

### 상점 서비스 또는 명령행을 사용하여 상점 입구 자원 및 상점 구성 파일 공개

상점 서비스를 사용하여 또는 명령행에서 공개 유틸리티를 사용하여 상점 입구 자원 및 상점 구성 파일을 구성할 수 있습니다.

주: 상점 서비스 또는 공개 유틸리티를 사용하여 상점 구성 파일을 공개하려면, 웹 자원 및 데이터 자원을 포함하는 모든 상점 자원을 공개해야 합니다. 상점 구성 파일만 공개하도록 선택할 수 없습니다.

상점 서비스 또는 명령행을 사용하여 상점 입구 자원 및 상점 구성 파일을 공개하려면, 상점 입구 자원 및 상점 구성 파일은 상점 아카이브 포맷으로 패키징되어야 합니다. 상점 아카이브로 상점 입구 자원을 구성하는 자세한 정보에 대해서는 237 페이지의 제 6 부 『상점 패키지』를 참조하십시오.

상점 서비스 또는 명령행 공개를 사용하려는 경우, 상점 아카이브(상점 입구 자원, 상점 데이터 자원 및 자원 번들 포함)에서 모든 유형의 자원의 공개를 선택하거나 단지

하나의 자원 유형의 공개를 선택할 수 있습니다. 상점 서비스 또는 명령행을 사용하여 자원을 공개하기 위한 자세한 단계별 지시사항에 대해서는 WebSphere Commerce 온라인 도움말을 참조하십시오.

---

## WebSphere Commerce 서버에 복사하여 상점 입구 자원 및 상점 구성 파일 공개

자원 아카이브로 자원을 구성하지 않으려는 경우, WebSphere Commerce 서버에 직접 상점 입구 자원을 복사해서 공개할 수 있습니다. 웹 자원(HTML, JSP 파일, 이미지 및 그래픽)은 웹 응용프로그램 문서 루트로 복사되어야 합니다. 자원 번들 또는 특성 파일은 응용프로그램 특성 경로로 복사되어야 합니다.

상점 입구 자원 및 상점 구성 파일을 WebSphere Commerce 서버에 복사하려면, 다음을 수행하십시오.

1. 파일, 이미지 및 그래픽을 포함하는 HTML 및 JSP 파일을 상점 웹 응용프로그램 문서 루트의 상점 디렉토리(*storedir*)로 복사하십시오.

- ▶ NT `drive:\WebSphere\AppServer\installedApps\WC_Enterprise_App_instancename.ear\wcstores.war\storedir`
- ▶ 2000 `drive:\Program Files\WebSphere\AppServer\installedApps\WC_Enterprise_App_instancename.ear\wcstores.war\storedir`
- ▶ AIX `/usr/WebSphere/AppServer/installedApps/WC_Enterprise_App_instancename.ear/wcstores.war/storedir`
- ▶ Solaris `/opt/WebSphere/AppServer/installedApps/WC_Enterprise_App_instancename.ear/wcstores.war/storedir`
- ▶ Linux `/opt/WebSphere/AppServer/installedApps/WC_Enterprise_App_instancename.ear/wcstores.war/storedir`
- ▶ 400 `/QIBM/UserData/WebASAdv4/WASinstancename/installedApps/WC_Enterprise_App_instancename.ear/wcstores.war/storedir`

여기서, *storedir*은 STORE 데이터베이스 테이블에서 DIRECTORY 열의 값입니다. 이 값이 존재하지 않는 경우, 다음을 수행하여 값을 추가할 수 있습니다. 상점에서 디렉토리 선택=공개 파일 자원에 대한 관련 디렉토리 추가

2. 자원 번들 및 특성 파일을 응용프로그램 특성 경로로 복사하십시오.

- ▶ NT `drive:\WebSphere\AppServer\installedApps\WC_Enterprise_App_instancename.ear\wcstores.war\WEB-INF\classes\storedir`







- ▶ 2000 drive:\Program Files\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_instancename.ear\wcstores.war\WEB-INF\classes\storedir
- ▶ AIX /usr/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_instancename.ear/wcstores.war/WEB-INF/classes/storedir
- ▶ Solaris /opt/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_instancename.ear/wcstores.war/WEB-INF/classes/storedir
- ▶ Linux /opt/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_instancename.ear/wcstores.war/WEB-INF/classes/storedir
- ▶ 400 /QIBM/UserData/WebASAdv4/WASinstancename/installedApps/WC\_Enterprise\_App\_instancename.ear/wcstores.war/WEB-INF/classes/storedir

3. 상점 구성 파일을 WebSphere Commerce 구성 파일 instance\_name.xml에서 정의된 위치로 복사하십시오. 이 파일은 다음 디렉토리에 있습니다.

- ▶ NT drive:\WebSphere\CommerceServer\instances\instancename\xml
- ▶ 2000 drive:\Program Files\WebSphere\CommerceServer\instances\instancename\xml
- ▶ AIX /usr/WebSphere/CommerceServer/instances/instancename/xml
- ▶ Solaris /opt/WebSphere/CommerceServer/instances/instancename/xml
- ▶ Linux /opt/WebSphere/CommerceServer/instances/instancename/xml
- ▶ 400 /QIBM/UserData/WebCommerce/instances/instancename/xml

상점 구성 파일은 다음 위치로 복사됩니다.

- runtime\_xml.zip 파일이 StoresXMLPath로 복사됩니다. 이 경로는 WebSphere Commerce 구성 파일 instance\_name.xml에 정의됩니다.
- tools\_properties.zip 파일이 ToolsStoresPropertiesPath로 복사됩니다. 이 경로는 WebSphere Commerce 구성 파일 instance\_name.xml에 정의됩니다.

- tools\_xml.zip 파일이 ToolsStoresXMLPath로 복사됩니다. 이 경로는 WebSphere Commerce 구성 파일 instance\_name.xml에 정의됩니다.
4. 다음 방법 중 하나를 사용하여 상점을 실행하십시오.
- StoreCatalogDisplay 명령 사용: StoreCatalogDisplay? storeId=storeId&catalogId=catalogId&langId=langId  
여기서,
    - storeId는 STORE 데이터베이스 테이블의 STORE\_ID 열에 위치한 값입니다.
    - catalogId는 CATALOG 데이터베이스 테이블의 CATALOG\_ID 열에 위치한 값입니다.
    - langId는 주어진 로케일에 대한 LANGUAGE 데이터베이스 테이블의 LANGUAGE\_ID 열 값입니다. 기본 WebSphere Commerce 값의 목록은 LANGUAGE 데이터베이스 테이블을 참조하십시오.
  - 상점이 WebSphere Commerce 견본 상점을 기반으로 하는 경우, 상점 URL을 모아 다음 디렉토리 아래의 index.jsp 파일을 편집하여 구성하십시오.
    -  NT drive:\WebSphere\CommerceServer\wc.ear\wcstores.war\storedir
    -  2000 drive:\Program Files\WebSphere\CommerceServer\wc.ear\wcstores.war\storedir
    -  AIX /usr/WebSphere/CommerceServer/wc.ear/wcstores.war/storedir
    -  Solaris  Linux /opt/WebSphere/CommerceServer/wc.ear/wcstores.war/storedir
    -  400 /QIBM/ProdData/WebCommerce/wc.ear/wcstores.war/storedir

다음 매개변수에 대한 올바른 값을 추가하십시오.

- hostname은 사용자 WebSphere Commerce 시스템의 완전한 이름입니다.
- storeId는 STORE 데이터베이스 테이블의 STORE\_ID 열에 위치한 값입니다.
- catalogId는 CATALOG 데이터베이스 테이블의 CATALOG\_ID 열에 위치한 값입니다.
- langId는 주어진 로케일에 대한 LANGUAGE 데이터베이스 테이블의 LANGUAGE\_ID 열 값입니다. 기본 WebSphere Commerce 값의 목록은 LANGUAGE 데이터베이스 테이블을 참조하십시오.

브라우저에서 상점을 보려면 `http://host name/webapp/wcs/stores/  
servlet/storedir/index.jsp` URL을 실행하십시오.



---

## 제 8 부 상점에 WebSphere Commerce 특징 추가

WebSphere Commerce에서 사용 가능한 일부 기능을 상점에 추가하려면, 일부는 수동으로 단계를 완료해야 합니다. 다음 장에서는 상점에 다음 기능들을 추가하는 것에 대해 설명합니다.

- 335 페이지의 제 31 장 『상점에 고객 지원 추가』
- 351 페이지의 제 32 장 『상점에 e-Marketing 지점 추가』





---

## 제 31 장 상점에 고객 지원 추가

WebSphere Commerce의 고객 지원 기능은 Lotus® Sametime™ 서버를 사용한 동기 텍스트 인터페이스를 통한 실시간 고객 서비스 지원을 제공합니다. 상점에서 고객 지원이 사용 가능하면 고객이 상점에 들어와 링크를 눌러 고객 서비스 영업대표(CSR)에 연결할 수 있습니다. 그렇게 하면, 고객은 인터넷을 통해 CSR과 상담할 수 있습니다.

주: 이 장은 상점에서 고객 지원을 사용하는 방법을 다룹니다. 하지만, 상점에서 고객 지원을 사용할 수 있기 전에, 먼저 Sametime 서버를 설치해야 하고 이를 구성하여 WebSphere Commerce와 작업해야 합니다. 자세한 정보는 *WebSphere Commerce 추가 소프트웨어 안내서*를 참조하십시오. 또한 관리 콘솔에 CSR을 등록하여 고객 지원에 사용 가능해야 합니다. 고객 지원의 전반적인 개념, CSR이 고객 지원을 사용하는 방법 뿐만 아니라, 이 태스크에서 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오.

주:

**Business** ToolTech 및 NewFashion 견본 상점 중 하나에 기반을 둔 상점을 작성하면, 상점 서비스를 통해 쉽고 빠르게 고객 지원을 사용할 수 있습니다. 상점 서비스를 사용하여 상점을 공개한 다음, 상점을 선택하고 구성을 선택하여 고객 지원 기능을 사용 가능하게 하십시오. 자세한 지시사항은 WebSphere Commerce 온라인 도움말을 참조하십시오.

하지만 견본 상점을 기반으로 상점을 작성하지 않았다면, 상점에서 고객 지원을 사용하기 위해 몇 가지 작업을 해야 합니다. 이 장의 나머지 부분에서는 견본에 기반을 두지 않은 상점에서의 고객 지원 사용을 위한 필요 개념 및 단계들을 논의합니다.

주: 견본 상점 **Business** ToolTech 및 NewFashion은 고객 지원이 어떻게 구현되어야 하는지를 보여주고, 고객 지원을 상점에서 사용할 수 있게 할 때 사용할 수 있는 코드를 제공합니다. 이 장에서는 이 두 상점에서의 예를 인용하여 상점에서 고객 지원을 사용 가능하게 하는 방법을 보여줄 것입니다. 이 장을 읽을 때, 견본 상점(WebSphere Commerce 상점 웹 사이트)의 최신 버전이 있음을 확인하십시오.

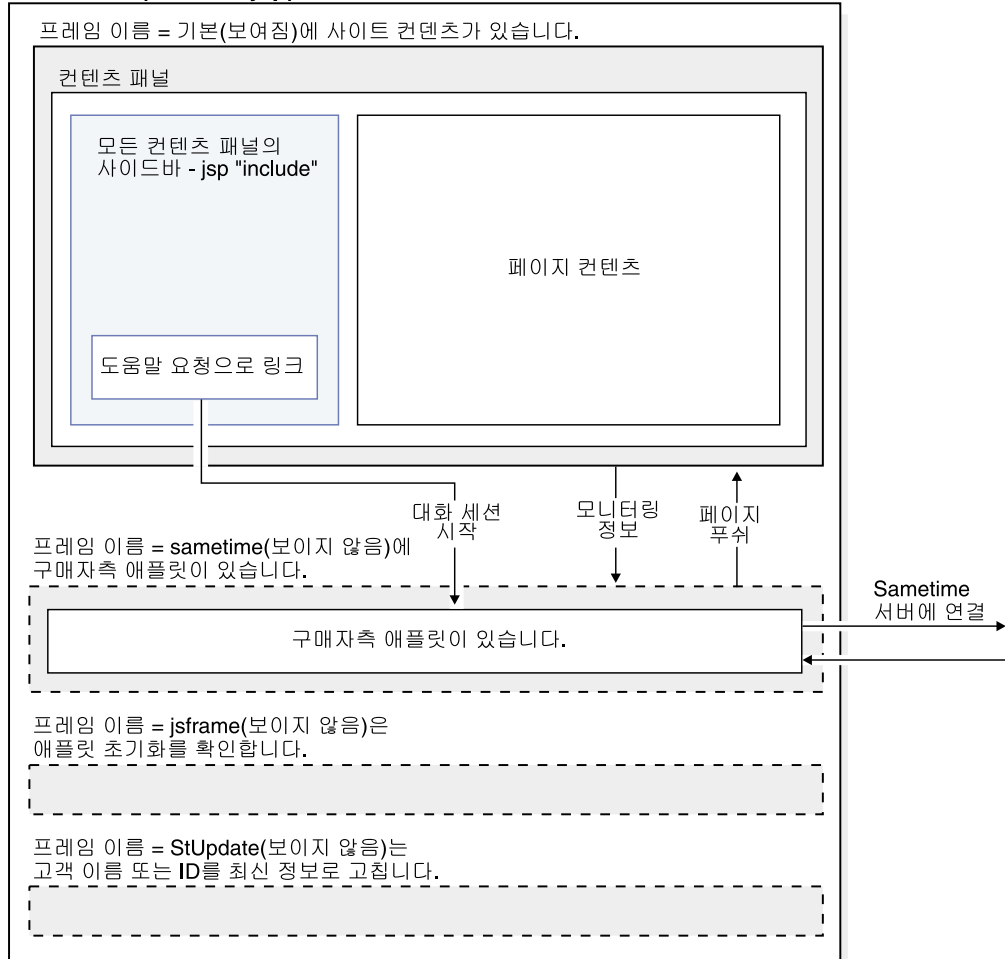
---

### 상점에서의 고객 지원 이해

고객이 고객 지원이 가능한 상점에서 고객 지원 상담원과 실시간 대화와 같은 고객 지원 링크를 선택했을 때, 대화 창을 포함한 애플릿이 실행됩니다. 이 애플릿은 사이트의 전반적인 모습이나 분위기를 방해하지 않는 숨겨진 프레임셋에서 실행됩니다. 애플릿이 실행되면 Lotus Sametime 서버에 연결됩니다.

다음 도표는 프레임셋의 구성을 보여줍니다.

### 프레임셋(in index.jsp)



프레임셋에는 다음 네 가지 프레임이 포함됩니다.

- 기본: 페이지 본문, 머리글 및 바닥글 파일, 사이드바 파일 및 상점 페이지를 작성하는 파일을 포함하는 상점에 대한 콘텐츠를 포함하는 프레임. 이 프레임의 콘텐츠는 상점의 손님에게 보여집니다. 기본 프레임은 Sametime 프레임으로의 고객 지원 및 모니터링 정보로의 링크를 포함합니다. 모니터링 정보는 338 페이지의 『고객 지원을 사용하는 고객 모니터링』에서 자세히 설명합니다.
- Sametime: 고객 지원 애플릿을 포함하는 프레임. 이 프레임은 손님에게는 보이지 않습니다. 하지만 고객이 애플릿 시작 링크를 누르면 고객은 고객 지원 창을 볼 수 있습니다. 또한 이 프레임은 페이지 푸쉬 기능을 통해 정보를 기본 프레임에 푸쉬합니다.
- jsframe: 이 프레임은 애플릿이 제대로 로드되었는지 확인합니다. 이 프레임의 콘텐츠는 고객에게 보이지 않습니다.
- StUpdate: 프레임은 고객 이름 또는 ID를 최신 정보로 고칩니다.

## 프레임셋 사용

프레임셋에서 고객 지원 애플릿을 시작하면 상점 페이지의 코드에서 애플릿 코드를 분리합니다. 위의 도표에서처럼 애플릿 코드는 Sametime 프레임에 포함되고 상점 페이지는 프레임셋의 기본 프레임에 포함됩니다. 프레임셋이 처음 시작되었을 때 애플릿은 한 번만 다운로드되므로, 애플릿 코드를 상점 페이지에서 분리하면 네트워크 부하를 줄일 수 있습니다. 고객 지원 애플릿이 프레임셋의 일부가 아니라면, 애플릿은 모든 상점 페이지에 있어야 하며 새 상점 페이지가 액세스될 때마다 다운로드되어야 합니다.

또한 프레임셋을 사용하여 Sametime 서버와의 연결을 유지보수할 수 있습니다. 애플릿이 프레임셋이 아닌 페이지의 일부라면, 고객이 새 페이지에 액세스할 때마다 새로운 Sametime 세션이 작성될 것입니다. 고객 지원 애플릿이 Sametime 서버에 익명으로 로그인하므로, 고객이 새 페이지에 액세스할 때마다 새로운 세션 작성은 상점을 통해 고객의 활동을 추적할 수 없게 합니다. 프레임셋을 사용하여 고객의 고유 Sametime 세션이 유지보수되고, 고객의 활동이 속성 변경으로써 Sametime 서버로 돌려보내집니다.

### 프레임셋 사용 이슈

프레임셋을 사용하는 것이 고객 지원을 구현하는 데 있어서 추천되는 방법이지만, 프레임셋을 사용하는 데 따른 다음과 같은 이슈들을 알고 있어야 합니다.

- 단일 시작점: 고객들은 프레임셋 내에서 상점을 찾아볼 때만 고객 지원을 사용할 수 있습니다. 마찬가지로 CSR은 프레임셋을 통해서만 사용자를 모니터링할 수 있습니다. 고객이 프레임셋을 통해서 상점을 찾아보고 있다는 것을 확인하기 위해서 고객은 단일 시작점을 통해 사이트에 액세스해야 합니다. 예를 들면, 상점의 홈페이지(전본 상점의 경우, index.jsp)를 통해서 액세스하는 것입니다. 고객이 다른 페이지(예: 카탈로그 페이지)를 통해 상점에 액세스하는 경우, 프레임셋에 있는 것이 아닙니다.
- 북마킹: 프레임셋을 사용할 때 고객은 개별 페이지가 아닌 사이트의 기본 URL만을 북마크할 수 있습니다.
- 최신 정보로 고침: 고객이 프레임셋에서 최신 정보로 고침을 눌렀을 때 프레임셋에 코드된 것처럼(예: index.jsp) 기본 프레임 주소로 돌아갑니다.
- 브라우저 창 크기 조정: 프레임셋에 있는 동안, 고객이 브라우저 창의 크기를 조정하는 경우, 브라우저는 자동으로 입력 주소를 다시 로드합니다. 입력 주소가 다시 로드되는 경우, Sametime 서버에 연결이 종료될 수 있습니다. 다른 브라우저는 이 경우 다르게 작동합니다.
- 보안: 고객이 프레임셋을 통해 사이트를 찾아보기 할 때, 프레임셋(위치 막대의 URL) 뿐만 아니라 각 개별 프레임도 보안을 해제하거나(http, 기본: 포트 80) 보안을 설정하여(https, 기본: 포트 443) 고유 연결을 유지보수합니다. 고객이 보안되지 않은 연결을 통해 상점을 찾아보고 있다면 프레임셋 내의 모든 프레임은 http

에 있습니다. 시나리오에는 SSL과 관련된 이슈가 없습니다. 하지만 고객이 보안 페이지(예: 등록 페이지)를 찾아보고 있다면 프레임세트의 기본 프레임은 https로 변경되고, 나머지 프레임은 보안이 않되는 상태(HTTP)가 됩니다. 이런 상황에서 고객은 고객 지원 애플릿을 시작할 수 없습니다. 애플릿(보안, 포트 443)이 브라우저(HTTP, 포트 80)의 위치 막대의 URL과 다른 서버로부터 오는 것으로 나타나므로 브라우저는 애플릿 시작을 허용하지 않습니다. 애플릿은 전체 프레임세트가 다시 보안될 때까지, 즉 위치 막대의 보안 막대의 URL이 HTTPS를 지정할 때까지 시작되지 않습니다. 전체 프레임세트를 다시 보안하는 데는 다음 옵션들이 있습니다.

- 전체 프레임세트를 보안 연결로 방향 재지정하십시오. 하지만, 이렇게 하는 도중에 애플릿이 다른 새로운 Sametime 연결로 전환하면 현재 진행중인 모든 대화는 종료됩니다.
- 사이트에 들어갈 때 전체 프레임세트를 보안 연결로 방향 재지정하십시오. 이로 인해 성능은 약간 떨어지지만 고객 세션에 보안 및 찾아보기 프라이버시가 추가됩니다.

안전한 연결로 프레임세트를 재지정하는 한 가지 방법은 index.html 파일에 다음 코드를 추가하는 것입니다.

```

-
<html>
<head>
<META HTTP-EQUIV=Refresh CONTENT="0;URL=https://hostname/webapp/wcs/stores/
servlet/NewFashion/index.jsp>
</head>
<html>
</head>

```

## 고객 지원을 사용하는 고객 모니터링

고객 지원을 사용하면 상점의 CSR에 해당하는 고객을 모니터링할 수 있습니다.


- 고객 이름 및 ID 확보
- 고객이 찾아보고 있는 페이지 결정
- 장비구나의 항목 추적

사용자 정의된 코드는 이 정보를 얻기 위해 상점 페이지에 추가됩니다. 다음 절에서는 이 모니터링 기능이 전본 상점에서 어떻게 구현되는지에 관하여 논의합니다.

### 고객 이름 및 ID 확보

고객 지원 애플릿이 시작되고 CSR이 로그인하면, CSR은 이름 및 구매자 ID를 통해 누가 애플릿을 사용하고 있는지 식별할 수 있습니다. 전본 상점은 사용자의 이름 및 구매자 ID를 결정할 수 있는 고객 지원 애플릿에 작동하는 특별한 코드를 포함합니다. 이 코드는 고객이 게스트 고객인지, 장비구내 항목을 가지고 있는 게스트 고객인지 또는 등록된 고객인지를 결정하고, 고객에게 이름 또는 ID를 지정하고 그 이름을 고객 지원 애플릿으로 되돌려 보냅니다. 이 이름은 CSR로 표시됩니다. 예를 들어, 고객이 장비구

나에 아무 것도 없는 게스트 고객이면 이 고객은 생성된 구매자 ID -1002로 지정됩니다. 고객이 장바구니에 항목이 있는 게스트 고객이면 구매자 ID가 표시되고, 고객이 등록된 고객이면 이름과 성이 표시됩니다.

전본 상점은 다음 코드를 StUpdate 프레임을 최신 정보로 고치는 상점 헤더 파일에 추가하여 고객의 이름 및 ID를 확보합니다. 이 코드는 NewFashion의 header.jsp 파일에 포함되고  ToolTech의 NavHeader.jsp 파일에 포함됩니다.


주: 고객이 상점의 새 페이지를 찾아볼 때마다 고객의 이름 또는 ID가 최신 정보로 고쳐집니다.

```
<script language="javascript">
 if (typeof top.updateStInfo == 'function')
 top.updateStInfo();
</script>
```

이전 코드는 StUpdate 프레임에서 다음을 최신 정보로 고칩니다.

```
//set Customer Name for LiveHelp if user is registered.

if (userRegistrationDataBean.findUser()) {
if (userRegistrationDataBean.getLastName() != null & &
userRegistrationDataBean.getLastName().length() > 0) {
 if(cmdcontext != null) {
 Long uid = cmdcontext.getUserId();
 String customerName = "";
 if (locale.toString().equals("ja_JP")||locale.toString().equals("ko_KR")
||locale.toString().equals("zh_CN")||locale.toString().equals("zh_TW"))
 {
 customerName = " " + userRegistrationDataBean.getLastName() + " "
+ userRegistrationDataBean.getFirstName();
 }
 else {
 customerName = " " + userRegistrationDataBean.getFirstName() + " "
+ userRegistrationDataBean.getLastName();
 }
 }
}
else {
 customerName=userRegistrationDataBean.getUserId();
 if (customerName.equals("-1002"))
 customerName="";
 customer_name=customer_name.trim();
}
}
```

전본 상점의 로그인 페이지에는 고객 이름을 생성된 ID로 설정하고 장바구니에 있는 항목 번호를 0으로 재설정하는 더 많은 사용자 정의 코드가 있습니다. NewFashion의 로그인 페이지는 LoginForm.jsp입니다.  ToolTech에서는 Logoff.jsp입니다. 사용자 정의 코드는 다음과 같습니다.


```
<HTML>
<HEAD>
<SCRIPT language="javascript">
 if (typeof parent.setCustomerName == 'function')
 parent.setCustomerName (parent.WCSGUESTID, '')
 if (typeof parent.setShoppingCartItems == 'function')
```

```

 parent.setShoppingCartItems(0);
</SCRIPT>
</HEAD>
</HTML>

```

## 고객이 찾아보고 있는 페이지 판별

고객 지원은 CSR을 통해 고객이 상점에서 어떤 페이지를 찾아보고 있는지를 판별할 수 있게 합니다. 견본 상점은 다음 코드를 헤더 파일에 추가하여 고객이 어느 페이지에 있는지를 판별합니다(NewFashion의 header.jsp 와  ToolTech의 NavHeader.jsp).

```

<%
//Determine Page Type for LiveHelp
String headerType = (String) request.getAttribute("liveHelpPageType");
if (headerType==null)
headerType = "";

%>

<script language="javascript">
<%
 String pname = request.getRequestURI();
 int indpn = pname.lastIndexOf('/');
 indpn = pname.lastIndexOf('/', indpn-1);
 if(indpn != -1)
 pname = pname.substring(indpn+1);

 //Determine if this is a personal page or not
 if (headerType.equals("personal")) {
%>
if (typeof parent.setPageParams == 'function')
 parent.setPageParams('PERSONAL_URL', '<%=pname%>');
<% } else { %>
if (typeof parent.setPageParams == 'function')
parent.setPageParams(location.href, '<%=pname%>');
<% } %>
</script>

```

페이지가 헤더 파일을 사용하지 않으면 파일 StHeader1.jsp가 페이지에 포함됩니다. StHeader1.jsp는 상점의 헤더 파일에 추가된 것과 동일한 코드를 포함합니다.

고객 프라이버시를 유지보수하기 위해 CSR은 특정 페이지에 액세스할 수 없어야 합니다. 예를 들어, CSR은 캠페인 페이지나 장기 구매 계약에 의해 결정된 가격을 포함하는 페이지 또는 주소록 페이지와 같은 사용자 ID를 포함하는 페이지에 액세스할 수 없어야 합니다. 이 페이지는 개인용으로 표시됩니다. 견본 상점에서 다음 페이지는 개인용으로 표시됩니다.

- NewFashion
  - AddressBookForm.jsp
  - AllocationCheck.jsp
  - edit\_registration.jsp

- emptyshoppingcart.jsp
- interestItemDisplay.jsp
- myaccount.jsp
- orderItemDisplay.jsp
- OrderDisplayPending.jsp
- ResultList.jsp
- shoppingcart.jsp
- TrackOrderStatus.jsp
- Business ToolTech
  - Address.jsp
  - Addressbook.jsp
  - AddToExistReqList.jsp
  - AdvancedSearch.jsp
  - AllocationCheck.jsp
  - CatalogMainDisplay.jsp
  - CatalogItemDisplay.jsp
  - CatalogTopCategoriesDisplay.jsp
  - Confirmation.jsp
  - OrderDisplayPending.jsp
  - OrderItemDisplay.jsp
  - OrderDetail.jsp
  - QuickOrder.jsp
  - RequisitionListCreate.jsp
  - RequisitionListDetailDisplay.jsp
  - RequisitionListDisplay.jsp
  - RequisitionListUpdate.jsp
  - Result List.jsp
  - Shipping.jsp
  - shoppingcart.jsp
  - TrackOrderStatus.jsp
  - UserAccount.jsp
  - UserRegistrationUpdate.jsp

(CSR이 액세스할 수 없는) 개인용으로 페이지를 표시하기 위해 견본 상점은 헤더가 포함되기 바로 전에 아래 코드를 페이지에 포함시킵니다.

```

<%
// Set header type needed for this JSP for Customer Care. This must
// be set before Header.jsp
request.setAttribute("liveHelpPageType", "personal");
%>

<%
String incfile;
incfile = includeDir + "Header.jsp";
%>
<jsp:include="<%=incfile%>"flush="true"/>

```

주: CSR이 표시된 개인용 페이지의 콘텐츠를 볼 수 없지만, 해당 페이지의 URL은 볼 수 있습니다.

### 장바구니에서 항목 수 추적

고객 지원은 CSR을 통해 언제라도 고객의 장바구니에 얼마나 많은 항목들이 있는지 추적할 수 있습니다. 견본 상점은 다음 위치에서 장바구니에 있는 항목 수를 얻습니다.

- 장바구니 페이지(NewFashion: shoppingcart.jsp)  
Business ToolTech:ShoppingCart.jsp)
- 비어있는 장바구니 페이지(NewFashion: emptyshopcart.jsp)  
Business ToolTech:EmptyOrder.jsp)
- 주문 확인 페이지(NewFashion: confirmation.jsp)  
Business ToolTech:confirmation.jsp)
- 로그아웃 페이지(Business ToolTech:Logoff.jsp)

주: 장바구니가 개인용 페이지로 지정된다 해도 CSR은 장바구니에서 항목 수를 추적할 수 있습니다. CSR은 이 방법을 사용하여 장바구니가 포함하고 있는 것을 볼 수 없고 항목의 수만 볼 수 있습니다. 하지만, CSR은 장바구니 보기 버튼을 사용하여 장바구니의 콘텐츠를 볼 수 있습니다. 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오.

견본 상점은 다음 코드를 위의 페이지에 추가하여 장바구니에 있는 항목 수를 결정합니다.

- 첫 번째, int 변수가 정의됩니다.

```
int liveHelpShoppingCartItems= 0;
```
- 두 번째, 다음 코드 행은 장바구니에 추가된 orderitem이 있을 때마다 그 수량을 liveHelpShoppingCartItems에 추가하는 데 사용됩니다.

```
liveHelpShoppingCartItems+= orderItem.getQuantityInEJBType().intValue();
```
- 세 번째, 다음 코드를 페이지의 마지막 부분에 추가하여 고객 이름을 게스트 구매자 ID로 설정하고 고객 장바구니에 있는 항목 수를 확보합니다.



```

<script language="javascript">
if (typeof parent.setShoppingCartItems == 'function')
parent.setShoppingCartItems(<%=liveHelpShoppingCartItems%>);
</script>

```

다음 코드는 빈 장바구니 페이지 및 주문 확인 페이지에서 사용되며 장바구니에 있는 항목 수를 0으로 재설정합니다.

```

<script language="javascript">
if (typeof parent.setShoppingCartItems == 'function')
parent.setShoppingCartItems(0);
</script>

```

---

## 상점에 고객 지원 추가

고객 지원을 견본을 기반으로 하지 않는 상점에 추가하려면 다음을 수행하십시오.

### 부분 1: 설치 전제 조건

상점에서 작동하는 고객 지원에 대한 주문에 대해서는 다음을 수행해야 합니다.

- Sametime 서버를 설치하십시오. 자세한 정보는 *WebSphere Commerce 추가 소프트웨어 안내서*를 참조하십시오.
- WebSphere Commerce Sametime 통합 패키지를 설치하십시오. 자세한 정보는 *WebSphere Commerce 추가 소프트웨어 안내서*를 참조하십시오.
- WebSphere Commerce 인스턴스를 중지한 후, 구성 관리자에서 Sametime을 사용 가능하게 하고 인스턴스를 다시 시작하십시오. 자세한 정보는 *WebSphere Commerce 추가 소프트웨어 안내서*를 참조하십시오.
- CSR을 작성하고 관리 콘솔을 사용하여 고객 지원에 대한 CSR을 등록하십시오. 자세한 정보는 *WebSphere Commerce 온라인 도움말*을 참조하십시오.







### 부분 2: 견본 상점으로부터 고객 지원 통합 파일 복사

견본 상점 NewFashion 및 ToolTech는 고객 지원을 상점으로 통합하는 데 사용되는 다음 파일들을 포함합니다.

- *Sametime.js*: 모든 프레임에서 포함되는 JavaScript 기능을 포함합니다. 이 파일로부터의 함수는 기본 프레임(예: `parent.setCustomerName`)에 있는 페이지로부터 상위 접두부로 호출됩니다.
- *StBlank.jsp*: 비어있는 JSP 파일.
- *StFrame.jsp*: JavaScript 함수를 포함하고 상점 입구에 대한 고객 지원 애플릿을 임베드합니다.
- *StReadyJS.jsp*: 애플릿이 정상적으로 로드된 것을 표시합니다.
- *StHeader1.jsp*: 이 헤더를 포함하는 페이지가 개인 페이지인지를 표시하는 애플릿으로 매개변수를 전달하는 헤더 파일.

- StUpdate.jsp: 고객 이름 정보 및 ID를 갱신합니다.

견본 상점에서 사용자 상점으로 Sametime 통합 파일을 복사하려면 다음을 수행하십시오.

1. NewFashion 상점 또는 ToolTech 상점에 대한 상점 아카이브 파일을 찾으십시오. 상점 아카이브 파일은 다음 디렉토리에 있습니다.
  -  NT drive:\WebSphere\CommerceServer\samplstores
  -  2000 drive:\Program Files\WebSphere\CommerceServer\samplstores
  -  AIX /usr/WebSphere/CommerceServer/samplstores
  -  Solaris /opt/WebSphere/CommerceServer/samplstores
  -  Linux /opt/WebSphere/CommerceServer/samplstores
  -  400 /qibm/proddata/WebCommerce/samplstores
2. ToolTech 또는 NewFashion 폴더를 열고 ToolTech 또는 NewFashion 상점 아카이브를 선택하십시오.
3. WinZip 또는 비슷한 도구를 사용하여 상점 아카이브 파일을 여십시오.
4. webapp.zip 파일을 찾으십시오. WinZip 또는 비슷한 도구를 사용하여 파일을 여십시오.
5. 다음 파일들을 선택하십시오.
  - Sametime.js
  - StBlank.jsp
  - StFrame.jsp
  - StReadyJS.jsp
  - StHeader1.jsp
  - StUpdate.jsp
6. 상점에 대한 웹 자원을 포함하고 있는 디렉토리로 파일을 추출하십시오.

주: StHeader1.jsp는 견본 상점 아카이브의 포함 디렉토리에 있습니다. 상점 페이지에 포함시킨 파일에 대한 별도의 디렉토리가 있다면, 그 디렉토리에 StHeader1.jsp를 저장하십시오. 디렉토리가 없다면, 다른 상점 파일과 동일한 디렉토리에 저장하십시오.

### 부분 3: 상점에 프레임셋 추가

337 페이지의 『프레임셋 사용』에서 논의한 것처럼, 고객 지원 애플릿은 프레임셋에서 실행됩니다. 프레임셋은 상점 홈페이지 또는 고객이 상점으로 들어갈 때 가장 빈번히 거쳐갈 만한 페이지에 추가되어야 합니다. 상점에 프레임셋을 추가하려면 다음을 수행하십시오.

- 어떤 페이지가 상점에 대한 시작점인지 결정하십시오.
- ToolTech 또는 NewFashion에 대한 상점 아카이브 파일을 여십시오. 상점 아카이브 파일은 다음 디렉토리에 있습니다.
  - ▶ NT drive:\WebSphere\CommerceServer\samplstores
  - ▶ 2000 drive:\Program Files\WebSphere\CommerceServer\samplstores
  - ▶ AIX /usr/WebSphere/CommerceServer/samplstores
  - ▶ Solaris /opt/WebSphere/CommerceServer/samplstores
  - ▶ Linux /opt/WebSphere/CommerceServer/samplstores
  - ▶ 400 /qibm/proddata/WebCommerce/samplstores
- WinZip 또는 비슷한 도구를 사용하여 상점 아카이브 파일을 여십시오.
- webapp.zip 파일을 찾으십시오. WinZip 또는 비슷한 도구를 사용하여 파일을 여십시오.
- index.jsp 파일을 여십시오.
- 다음 코드를 복사하십시오.

```

<script src="<%= "Sametime.js"%></script>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"DTD/xhtml1-transitional.dtd">
<html>
<head>
<title></title>
<script language="javascript">
var MainPageURL="";
if (MainPageURL=="")
MainPageURL="/webapp/wcs/stores/servlet/Logoff?storeId=<%=storeId%>
&langId=<%=langId%>&URL=LogonForm?
storeId=<%=storeId%>&catalogId=<%=catalogId%>";
function loadFrame()
{
main.document.location.href=MainPageURL;
}
</script>
</head>
<%=
String sBasePath="/webapp/wcs/stores/servlet/";
String sSametimeUrl="StFrame.jsp?storeId="+storeId;
String sBlankUrl="StBlank.jsp?storeId="+storeId;
String sUpdateUrl="StUpdate.jsp?storeId="+storeId;
%>
<FRAMESET border=0 frameBorder=0 ROWS="100%,1,1,1,1" onLoad="loadFrame();">
<FRAME NAME="main"
SRC="javascript:top.loadFrame();" MARGINWIDTH=0 SCROLLING="Auto"
FRAMEBORDER="no" noresize>
<FRAME NAME="JSFrame"
SRC="<%=sBlankUrl%>" MARGINWIDTH=0 SCROLLING="no"
FRAMEBORDER="no" noresize>
<FRAME NAME="sametime" SRC="<%=sSametimeUrl%>" MARGINWIDTH=0 SCROLLING="no"
FRAMEBORDER="no" noresize>
<FRAME NAME="StUpdate" SRC="<%=sUpdateUrl%>" MARGINWIDTH=0 SCROLLING="no"
FRAMEBORDER="no" noresize>
</FRAMESET>
</html>

```

주: 위의 예 ToolTech 견본 상점에서 "기본" 프레임에 대한 소스(SRC)는 다른 페이지에 대한 명령입니다. 상점이 설정되어 있는 방법에 따라 SRC는 명령, JSP 파일 또는 HTML 파일이 될 수 있습니다.

7. 6단계에서 복사한 코드를 상점의 시작점으로 지정한 페이지로 붙여넣으십시오. 기본 프레임에 대한 소스(SRC)에 필요한 변경을 수행하십시오.
8. 파일을 저장하십시오.

#### 부분 4: 고객 이름 및 ID 확보를 위한 코드 추가

CSR에 대한 고객 지원을 사용하는 고객의 구매자 ID 이름을 표시하기 위해 다음을 수행하십시오.

1. 338 페이지의 『고객 이름 및 ID 확보』 정보를 검토하십시오.
2. 어디에서 고객의 이름 또는 구매자 ID를 얻으려고 하는지를 결정하십시오. 예를 들면, 다음과 같습니다.
  - 상점으로의 시작점
  - 신규 등록 또는 등록 갱신
  - 로그아웃
  - 헤더
3. 어디에서 고객 이름 또는 구매자 ID를 얻으려고 하는지를 결정한 다음, 페이지가 상점에 대한 기본 헤더 파일을 포함하는지를 결정하십시오. 포함하지 않는다면, 필요한 코드를 페이지 자체에 추가하거나 헤더 파일을 포함할 것인지를 결정하십시오.
4. NewFashion 또는 ToolTech 상점 아카이브에 있는 webapp.zip으로부터 다음 파일들 중 하나를 여십시오.
  - NewFashion: header.jsp
  - ToolTech: NavHeader.jsp

주: 두 파일 모두 webapp.zip 파일에 있는 포함 디렉토리에 있습니다.

5. 다음 코드를 복사하십시오.

```
<script language="javascript">
 if (typeof top.updateStInfo == 'function')
 top.updateStInfo()
</script>
```

6. 5단계에서 복사한 코드를 상점에 대한 헤더 파일이나 적합한 페이지에 직접 붙여넣으십시오.

주: 원한다면, 위의 코드를 상점 헤더 파일이나 파일에 직접 복사하는 대신, 파일 StHeader1.jsp를 적합한 페이지에 포함시킬 수 있습니다.

7. 파일을 저장하십시오.

8. (선택적) 로그아웃 페이지에서 고객 ID(예: 고객 ID ~ -1002)를 재설정하거나 장바구니에 있는 항목 수를 0으로 재설정하려면 다음 코드를 로그아웃 파일에 추가하십시오.

```
<HTML>
<HEAD>
<SCRIPT language="javascript">
 if (typeof parent.setCustomerName == 'function')
 parent.setCustomerName (parent.WCSGUESTID, '')
 if (typeof parent.setShoppingCartItems == 'function')
 parent.setShoppingCartItems(0);
</SCRIPT>
</HEAD>
</HTML>
```

## 부분 5: 고객이 찾아보고 있는 페이지 판별을 위한 코드 추가

고객이 찾아보고 있는 페이지 판별을 위해 다음을 수행하십시오.

1. 상점의 헤더 파일에 StHeader1.jsp 파일을 포함시키십시오. 예를 들면, 다음과 같습니다.

```
<%@ include file="StHeader1.jsp" %>
```

2. 다음 코드를 개인용으로 표시되어야 하는 페이지에 추가하십시오.(따라서 CSR에서 액세스가 사용 불가능합니다)

```
<%
// Set header type needed for this JSP for Customer Care. This must
// be set before Header.jsp
request.setAttribute("liveHelpPageType", "personal");
%>
```

```
<%
String incfile;
incfile = includeDir + "Header.jsp";
%>
<jsp:include="<%=incfile%>" flush="true"/>
```

3. 헤더를 사용하지 않지만 개인 페이지라고 표시되어야 할 페이지에 다음 코드를 추가하십시오.

```
<%
// Set header type needed for this JSP for Customer Care. This must
// be set before StHeader1.jsp
request.setAttribute("liveHelpPageType", "personal");

String incfile;
incfile = includeDir + "StHeader1.jsp";
%>
<jsp:include page="<%=incfile%>" flush="true"/>
```

## 부분 6: 장바구니에 있는 항목 추적을 위한 코드 추가

CSR이 고객의 장바구니에 있는 항목을 추적할 수 있게 하려면 다음을 수행하십시오.

1. 342 페이지의 『장바구니에서 항목 수 추적』 정보를 검토하십시오.

2. 어디에서 장바구니 항목을 추적할지 결정하십시오. 예를 들면, 다음과 같습니다.

- 장바구니 페이지
- 빈 장바구니 페이지
- 주문 확인 페이지
- 로그아웃 페이지

3. 장바구니 항목을 추적하려는 페이지에서 다음을 수행하십시오.

a. int 변수를 정의하십시오.

```
int liveHelpShoppingCartItems= 0;
```

b. 주문 항목이 장바구니에 추가되었을 때 수량을 shoppingCartItems에 추가하려면 다음 코드 행을 추가하십시오.

```
liveHelpShoppingCartItems+= orderItem.getQuantityInEJBType().intValue();
```

c. 페이지의 끝부분에 다음 코드를 추가하여 게스트 구매자 ID(-1002)를 실제 구매자 ID로 설정하고 장바구니에서 항목 수를 얻으십시오.

```
<script language="javascript">
if (typeof parent.setCustomerName == 'function')
parent.setCustomerName(<%=cmdcontext.getUserId()%>, parent.CustomerName);
if (typeof parent.setShoppingCartItems == 'function')
parent.setShoppingCartItems(<%=liveHelpShoppingCartItems%>);
</script>
```

d. 빈 장바구니 페이지 및 주문 확인 페이지에서 데이터를 추적하려면, 그 페이지들에 아래 코드를 추가하여 장바구니 값을 0으로 설정하십시오.

```
<script language="javascript">
if (typeof parent.setShoppingCartItems == 'function')
parent.setShoppingCartItems(0);
</script>
```

## 부분 7: 고객 지원으로의 링크 추가

고객이 상점에서 고객 지원에 액세스하게 하려면 다음을 수행하십시오.

1. 어디에 고객 지원으로의 링크를 위치시킬지를 결정하십시오. 예를 들어, 탐색 모음에 링크를 위치시켜 고객이 언제든지 사용할 수 있게 하거나 상점의 특정 페이지에 위치시킬 수도 있습니다.
2. 다음 코드를 링크를 포함할 페이지에 복사하십시오.

```

<%=infashiontext.getString("LiveHelp")%>
```


## 부분 8: 고객에게 표시하는 메시지 변경

메시지가 CSR에 처음에 연결될 때, 고객에게 표시하는 메시지는 "안녕하세요, 무엇을 도와드릴까요?"입니다. 또는 "영업 시간은 오전 9시부터 오후 9시까지입니다"이고 Sametime 서버의 특성 파일에 저장됩니다. 특성 파일은 다음과 같이 두 가지 유형의 파일로 나뉩니다(Customer.properties 및 Agent.properties).

Customer.properties 파일에는 고객에게 표시하는 메시지가 포함되는 반면,

Agent.properties 파일에는 CSR에 표시하는 정보가 포함됩니다. 이 파일 두 개 모두 로케일 지정 파일이 있습니다(예: Customer\_de\_DE.properties 및 Agent\_de\_DE.properties). 각 로케일은 WebSphere Commerce의 사용자 인스턴스에 설치됩니다.

이 파일에서 메시지를 변경하려면, 다음을 수행하십시오.

1. Sametime 서버에 특성 파일을 찾으십시오. 기본적으로, 이 특성 파일은 다음 디렉토리에 있습니다.
  -  drive:\Sametime\Data\domino\html\wc\properties
2. 필요한 변경을 수행하십시오.
3. 파일을 저장하고 종료하십시오.





---

## 제 32 장 상점에 e-Marketing 지점 추가

e-Marketing 지점은 캠페인 행사 표시에 대한 마케팅 콘텐츠를 개인별 판매 설정한 상점 페이지에 공간을 예약합니다. 고객이 페이지를 요구한 경우, 페이지에 나타나는 모든 e-Marketing 지점은 이 지점과 연관된 규칙 기반 코드를 처리하기 위해 규칙 서버와 통신합니다. 각 e-Marketing 지점은 이와 연관된 하나 이상의 캠페인 행사를 가집니다. 캠페인 및 캠페인 행사에 대한 추가 정보는 129 페이지의 제 12 장 『캠페인 자원』 및 WebSphere Commerce 온라인 도움말을 참조하십시오.

상점 페이지에 캠페인 행사를 올바르게 표시하려면, e-Marketing 지점이 JSP 파일에 추가되어야 하며 WebSphere Commerce 액셀러레이터를 사용하여 데이터베이스에 등록해야 합니다. 이 장에서는 상점의 JSP 파일에 e-Marketing 지점을 추가하는 방법을 설명합니다. WebSphere Commerce 액셀러레이터를 사용하는 데이터베이스로의 e-Marketing 지점 등록에 대한 추가 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오.

---

### e-Marketing 지점

다음은 e-Marketing 지점의 예입니다.

```
<!-- =====
/**-----
/** 여기에 포함된 견본은 "현상태대로" 제공됩니다.
/**
/** 이것은 IBM이 제공하는 견본 예제로서 모든 조건하에서
/** 테스트되지 않았습니다. 그러므로 IBM은 신뢰성,
/** 내구성 또는 기능성을 보증할 수 없습니다.
/**
/** 이 견본은 가능한 확실하게 개념을 설명하기 위해
/** 개인, 회사, 브랜드 및 제품의 이름을
/** 포함할 수 있습니다.
/** 이러한 모든 이름은 가상의 이름이며,
/** 실제 사람 또는 비즈니스 엔터프라이즈에 의해 사용되는 이름 및 주소에 대한
/** 유사성은 전적으로 우연입니다.
/**-----
/**
=====-->
<%
/**
* 시작 - 다음 코드는 하나의 페이지에 한번만 존재해야 합니다. 이것은
* 명령 컨텍스트 및 상점 데이터 bean을 초기화합니다.
*/

// 상점 디렉토리를 확보하기 위해 상점 bean을 작성하십시오.
String collateralPath = "/webapp/wcs/stores/";
com.ibm.commerce.command.CommandContext emsCommandContext =
(com.ibm.commerce.command.CommandContext) request.getAttribute(
ECCConstants.EC_COMMANDCONTEXT);
com.ibm.commerce.common.beans.StoreDataBean storeDataBean =
new com.ibm.commerce.common.beans.StoreDataBean();
storeDataBean.setStoreId(emsCommandContext.getStoreId().toString());
com.ibm.commerce.beans.DataBeanManager.activate(storeDataBean, request);
if (storeDataBean.getDirectory() != null) {
collateralPath += storeDataBean.getDirectory() + "/";
```

```

}
%>
<!-- =====
// 다음 HTML 형식이 캠페인 통계를 캡처하는 ClickInfo 명령으로
// e-Marketing 지점에서 요청을 제출했으며 URL 매개변수가 지정한 위치로
// 경로 재지정합니다.
===== -->
<form name="storeEmsForm" method="POST" action="/webapp/wcs/stores/servlet/ClickInfo">
<input type="hidden" name="evtype">
<input type="hidden" name="mpe_id">
<input type="hidden" name="intv_id">
<input type="hidden" name="URL">
</form>
<%
/**
 * 끝 - 다음 코드는 하나의 페이지에 한번만 존재해야 합니다. 이것은
 * 명령 컨텍스트 및 상점 데이터 bean을 초기화합니다.
 */
%>
<%
/**
 * 시작 - 다음 코드는 복수 e-Marketing 지점을 해당 페이지로 놓기 위해
사용될 수 있습니다.
 * 사용하기 전에 적절한 EMarketingSpot 인스턴스 이름 및 e-Marketing
지점 이름을
 * 사용자 정의하십시오. 하나 이상의 지점이 필요한 경우, 코드를 중복하고
같은 지점 이름은
 * 사용하지 마십시오.
 */

// e-Marketing 지점을 작성하십시오.
com.ibm.commerce.marketing.beans.EMarketingSpot eMarketingSpot =
new com.ibm.commerce.marketing.beans.EMarketingSpot();

// 중요 - 올바른 이름을 여기에 설정하십시오.
eMarketingSpot.setName("eMarketingSpotName");

// 이 e-Marketing 지점을 통해 표시되는 제품/카테고리/광고 사본의 최대 수가
//여기에 설정될 수 있습니다.
eMarketingSpot.setMaximumNumberOfCatalogEntries(20);
eMarketingSpot.setMaximumNumberOfCategories(20);
eMarketingSpot.setMaximumNumberOfCollateral(20);

// bean의 인스턴스를 생성하십시오.
com.ibm.commerce.beans.DataBeanManager.activate(eMarketingSpot,
request);
%>

<%
// 다음 블록은 이 e-Marketing 지점과 연관되는 광고를 표시하는데
// 사용됩니다. 광고로 정의된 URL 링크가 위에 첨부된 HTML 양식의
// 제출을 통해 참조될 수 있습니다.

if (eMarketingSpot.getCollateral() != null && eMarketingSpot.getCollateral().
length > 0) {
%>
<TABLE>
<% for (int i=0; i eMarketingSpot.getCollateral().length; i++) { %>
<TR>
<% if (eMarketingSpot.getCollateral()[i].getTypeName().equals("Image")) { %>
<TD>
A HREF="javascript:document.storeEmsForm.evtype.value='CpgnClick'
;document.storeEmsForm.mpe_id.value=' <%= eMarketingSpot.getId() %>
';document.storeEmsForm.intv_id.value='<%= eMarketingSpot.getCollateral
()[i].getInitiativeId() %>';document.storeEmsForm.URL.value='
<%= eMarketingSpot.getCollateral()[i].getUrlLink() %>';
document.storeEmsForm.submit();">
IMG SRC=" <%= collateralPath + eMarketingSpot.getCollateral()[i].getLocation() %>">

</TD>
<TD>
<%= eMarketingSpot.getCollateral()[i].getMarketingText() %>
</TD>
<% } else if (eMarketingSpot.getCollateral()[i].getTypeName().equals("Flash")) { %>

```



```

getThumbnail() %>" ALT="<%= eMarketingSpot.getCatalogEntries()[i].getDescription
(emsCommandContext.getLanguageId()).getShortDescription()%" BORDER=0 WIDTH=60>

</TD>
<TD><%= eMarketingSpot.getCatalogEntries()[i].getDescription
(emsCommandContext.getLanguageId()).getShortDescription() %> </TD>
</TR>
<% } %>
</TABLE>
<% } %>

<% /**
 * END - 다음 코드는 복수 e-Marketing 지점을 해당 페이지로 놓는 데 사용됩니다.
 * 사용하기 전에 적절한 e-Marketing 지점 이름을 사용자 정의하십시오.
 * 하나 이상의 지점이 필요한 경우, 코드를 중복하고 같은 지점 이름은 사용하지 마십시오.
 */
%>

```

앞의 e-Marketing 지점은 세 가지 유형의 캠페인 행사를 지원합니다.

- 상품 추천
- 카테고리 추천 상품
- 인지 광고

주: 이 행사들 각각에 대한 자세한 정보는 129 페이지의 제 12 장 『캠페인 자원』을 참조하십시오.

## e-MarketingSpot bean

e-Marketing 지점은 e-MarketingSpot bean을 사용하여 지점에 현재 스케줄된 캠페인 행사의 결과를 리턴합니다. bean의 다른 특성 사용은 e-Marketing 지점 및 해당되는 캠페인 행사를 사용자 정의하도록 허용합니다. e-MarketingSpot bean 및 이의 등록 정보에 대한 추가 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오.

---

## 상점 페이지에 e-Marketing 지점 추가

상점 페이지에 e-Marketing 지점을 추가하려면 다음을 수행하십시오.

1. 지점이 표시될 JSP 파일을 결정하십시오. 이 지점이 여러 JSP 파일에 추가될 수 있습니다.
2. 지점을 위치 지정할 JSP 파일의 위치를 결정하십시오.
3. 351 페이지의 『e-Marketing 지점』에 견본 e-Marketing 지점을 복사하십시오.
4. JSP 파일의 원하는 위치로 견본 e-Marketing 지점을 붙여넣으십시오.
5. JSP 파일의 레이아웃을 맞추기 위해 견본 e-Marketing 지점을 사용자 정의하십시오.
6. e-Marketing 지점 코드 내에 e-Marketing 지점에 이름을 제공하십시오.

주: e-Marketing 지점은 HomePageAd 또는 CheckoutPageRecommendation과 같은 위치가 있는 설명적인 이름을 지정해야 합니다. 이것은 나타나는 위치와 콘텐츠 내용에 대한 혼동을 줄이는데 도움이 됩니다. 필요한 경우, 동일한 페이지

에 나타나는 두 개의 e-Marketing 지점 사이에 차이를 나타내기 위해 이름에 숫자를 추가할 수 있습니다. e-Marketing 지점 이름은 올바른 Java 식별자여야 합니다. WebSphere Commerce 액셀러레이터를 사용하여 데이터베이스에 e-Marketing 지점을 등록할 때 동일한 이름을 사용해야 합니다.

7. JSP 파일당 하나 이상의 e-Marketing 지점이 필요한 경우, 2 - 6단계를 반복하십시오. JSP 파일에 다른 e-Marketing 지점을 추가할 때, 351 페이지의 『e-Marketing 지점』에서 제공된 견본 e-Marketing 지점의 두 번째 절 모두를 복사하여 JSP 파일에 두 번째 전체 절을 붙여넣었는지 다시 확인하십시오. 그런 후, JSP 파일에 e-Marketing 지점의 이름을 변경하십시오.
8. WebSphere Commerce 액셀러레이터를 사용하여 데이터베이스에 e-Marketing 지점을 등록하십시오. 자세한 지시사항은 WebSphere Commerce 온라인 도움말을 참조하십시오.

**주:**

- a. 상점 ID, 카탈로그 ID 또는 언어 ID를 "langId=<%= languageId %>" 규칙을 사용하여 URL에 추가할 경우, e-Marketing 지점이 임베드된 JSP가 사용 가능한 적절한 ID를 작성해야 하는 것에 유의하십시오. 또한 ID는 명령 컨텍스트를 통해 검색될 수 있습니다(예: `getCommandContext().getLanguageId()`).
- b. NewFashion 견본 상점의 구조의 결과에 따라 NewFashion 견본 상점에 기반을 두는 상점의 항목이 아니라 상품만이 e-Marketing 지점을 통해 추천될 수 있습니다.
- c. URL 매개변수, `CatalogDisplay`는 "?" 대신에 "&"로 시작되어야 합니다. 이는 코드가 직접적으로 명령을 참조하지 않기 때문입니다.



---

## 제 9 부 부록





## 부록 A. UML 범례

통합된 모델 언어는 소프트웨어 디자인의 서로 다른 구성요소를 나타내기 위한 표준 그래픽 언어입니다. 다음 예는 UML의 가장 공통적인 구성요소들 중 일부입니다. 공식적인 스펙에 대한 자세한 정보는 <http://www.rational.com> and <http://www.omg.org> 를 참조하십시오.

UML 도표는 다음과 같은 항목으로 구성됩니다.

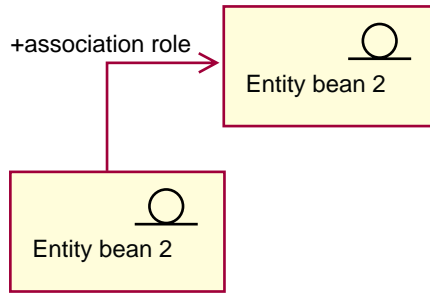
- 상자: 상자는 오브젝트의 클래스를 나타냅니다. 클래스 이름이 상자의 맨 위에 나타납니다. 속성은, 보일 경우, 클래스 이름 아래에 나타납니다. 클래스 이름과 속성은 행으로 분리되어 있습니다.
- 행: 행은 두 개의 클래스의 오브젝트간의 관계를 나타냅니다. 행의 한쪽 끝에 있는 클래스의 오브젝트를 다른 클래스의 오브젝트와 "연관"시킬 수 있습니다.
- 단색 다이아몬드: 행의 끝에 있는 단색 다이아몬드는 값별 내포를 나타냅니다. 행의 다른쪽 끝에 있는 클래스의 오브젝트는 다이아몬드가 터치하는 클래스의 하나 그리고 단 하나의 오브젝트의 파트입니다.
- 오픈 다이아몬드: 행의 끝에 있는 오픈 다이아몬드는 참조별 내포를 나타냅니다. 행의 다이아몬드 끝에 있는 오브젝트를 행의 다른쪽 끝에 있는 클래스의 오브젝트 그룹으로 생각할 수도 있습니다.
- 기수: 기수는 관계 행 끝에 나타나 기수 제한을 나타냅니다. 다음 테이블에는 기수 제한이 요약되어 있습니다.

기수	관계 유형
1	단 하나
0..1	0 또는 1
0..n	0개 이상
1..n	한 개 이상

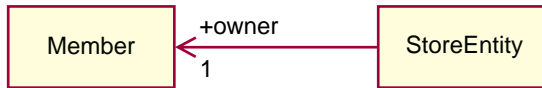
기수 제한이 보이지 않을 경우, 단색 다이아몬드가 관계 행 끝에 나타나지 않으면 기수가 0..n이라고 가정합니다. 이런 경우, 기수는 1입니다.

- 플러스(+) 부호: 관계 행 끝에 나타난 플러스(+) 부호는 행 끝에 있는 클래스의 오브젝트가 관계에서 어떤 역할을 행함을 나타냅니다. 플러스(+) 기호 뒤에 있는 텍스트는 관계에서 오브젝트의 역할을 나타냅니다.
- 화살표: 관계 행 끝에 있는 화살표는 두 오브젝트 간의 관계 방향이 화살표 방향임을 나타냅니다. 관계 행에 화살표가 없으면 오브젝트간의 관계 방향이 정상적으로 양쪽 방향임을 나타냅니다.

다음 도표는 위의 개념을 보여줍니다.



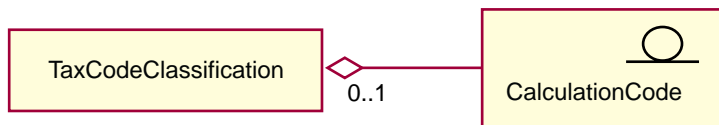
이 도표는 Enterprise Java Bean을 나타내는 장식적이고 형식적인 기호를 가진 두 개의 엔티티 bean을 보여줍니다. 첫 번째 bean에서 두 번째 엔티티 bean까지 단방향 연관이 있습니다. 플러스(+) 부호는 Entity bean 2가 어떤 연관 역할을 하는지 설명하는 텍스트 뒤에 나옵니다.



이 도표에서 StoreEntity는 구성원인 단 하나의 소유자를 갖습니다. 구성원은 0개 이상의 StoreEntity를 소유할 수 있습니다. 플러스(+) 부호는 구성원이 관계에서 어떤 역할을 행함을 나타냅니다. 이 경우, 구성원은 StoreEntity의 소유자입니다. 회살표는 구성원이 StoreEntity를 소유할 것을 요구하지 않으면서 모든 소유자의 StoreEntity를 요구해서 일반적으로 StoreEntity의 소유자를 찾을 수 있음을 나타냅니다.



이 도표에서, OrderItem은 항상 단 하나의 주문 파트입니다. 하나의 주문은 0개 이상의 OrderItem을 갖습니다.



이 도표는 CalculationCode가 0개 이상의 TaxCodeClassification 그룹이며, TaxCodeClassification 그룹은 0개 이상의 CalculationCode임을 나타냅니다.

---

## 부록 B. 데이터 작성

XML 파일의 양식으로 상점 데이터를 작성하기 전에, 다음을 수행하십시오.

- 작성 중인 정보의 순서를 결정하십시오. 각 상점 데이터 장의 정보는 데이터 작성 순서를 권고하지만, XML 파일 작성 시 상위 테이블에 대한 정보가 하위 테이블에 대한 정보에 우선해야 함을 기억하십시오.
- 상점을 사용하려는 방식을 결정하십시오. 새 상점을 작성하기 위한 기준으로 복사되고 사용되는 견본 상점 아카이브 파일(.sar)을 작성 중인 경우, 견본 상점을 작성하지 않는 경우와는 약간 다르게 데이터를 작성할 필요가 있습니다. 자세한 정보는 『견본 상점의 데이터 작성』을 참조하십시오.

---

### 견본 상점의 데이터 작성

견본 상점 아카이브에 있는 데이터는 로더 패키지에 적합하게 만들어진 XML 파일 양식입니다. 상점 아카이브 XML 파일은 휴대용으로 계획되고 데이터베이스의 특정 인스턴스에 고유한 생성된 기본 키를 포함하지 않아야 합니다. 대신 공개 시 ID 분석기가 해석하는 내부 별명을 사용합니다. 이런 규칙을 사용하면 견본 상점 아카이브를 여러 번 복사하고 공개할 수 있습니다.








XML 파일 양식으로 상점에 대한 상점 데이터를 작성할 때, 여러 상점을 생성하는데 사용되는 상점 아카이브를 작성하거나 휴대용 상점 아카이브를 작성하려고 하지 않는다면 이 규칙을 사용할 필요가 없습니다. 즉, 다른 WebSphere Commerce 인스턴스로 공개될 수 있는 것은 상점 아카이브입니다.

결과적으로 견본 상점 아카이브는 다음 규칙을 사용합니다.

- `member_id="& MEMBER_ID;" & XXX;` 규칙에서 처럼 `&`은 DTD 매크로(XML에서 엔티티로 알려진)입니다.

주: 견본 상점 아카이브 작성 시 DTD 매크로로 `MEMBER_ID`를 정의해야 합니다. WebSphere Commerce는 다음 파일의 매크로 세트를 정의합니다.

-  `drive:\WebSphere\CommerceServer\xml\sar\DBLoadMacros.dtd`
-  `drive:\Program Files\WebSphere\CommerceServer\xml\sar\DBLoadMacros.dtd`
-  `/usr/WebSphere/CommerceServer/xml/sar/DBLoadMacros.dtd`
-  `/opt/WebSphere/CommerceServer/xml/sar/DBLoadMacros.dtd`
-  `/opt/WebSphere/CommerceServer/xml/sar/DBLoadMacros.dtd`

- `400` /qibm/proddata/WebCommerce/xml/sar/DBLoadMacros.dtd

ko\_KR 및 es\_ES와 같은 매크로는 적절한 언어 ID에 설정됩니다. 예:

```
<!ENTITY ko_KR "-1">
```

상점 서비스의 도구를 사용하여 정보를 지정합니다. 예를 들어, 사용자는 상점 서비스의 상점 아카이브 작성 페이지에서 MEMBER\_ID를 선택합니다. MEMBER\_ID 매크로는 상점을 소유하는 구성원의 ID에 대한 위치 보유자입니다. 상점 아카이브를 작성할 때, 상점 소유자가 될 구성원을 선택합니다. MEMBER\_ID 매크로는 구성원 ID에 설정됩니다. 예를 들어, 구성원 ID -2000을 선택하면, MEMBER\_ID는 다음과 같이 -2000에 설정됩니다.

```
<!ENTITY MEMBER_ID "-2000">
```

- ffmcenter\_id="@ffmcenter\_id\_1"에서 처럼 @입니다. @ 기호 사용은 내부 별명 분석으로 알려져 있습니다. 로더 패키지 유틸리티인 ID 분석기는 필요로 하는 XML 요소의 식별자를 생성합니다. 기술 ID 분석기 중 하나는 내부 별명 분석입니다. 내부 별명 분석을 사용할 때, 별명은 XML 문서의 기본 키(식별자)를 대신합니다. 그런 다음 이 별명은 XML 파일의 다른 곳에서 사용되어 해당 요소를 참조합니다. 이를 사용하면 XML 파일을 빌드하는 데 필요한 고유 색인을 알 필요가 없습니다. 상점 서비스에서 공개 동안 또는 로더 패키지 사용 동안, ID 분석기가 @ 기호를 고유한 값으로 바꿉니다. XML 파일의 다음 예를 참조하십시오.

- 사전 ID 분석기

```
<catalog
 catalog_id="@catalog_id_1"
 member_id("&MEMBER_ID;"
 identifier=InFashion"
 description="InFashion Catalog"/>
```

- 우편 ID 분석기

```
<catalog
 catalog_id="10001"
 member_id="-2000"
 identifier=InFashion"
 description="InFashion Catalog"/>
```

여기서, 10001은 ID 분석기가 지정한 고유 ID이고, -2000은 상점 서비스에서 사용자가 선택한 구성원 ID입니다. XML 파일은 로더 패키지를 사용하여 로드됩니다. ID 분석기를 통해 파일을 실행하면, XML 파일의 단일 세트에서 여러 상점을 작성할 수 있습니다.

## 상점 서비스 및 견본 상점

상점 아카이브에서 새로 만들기 및 상점 아카이브 옵션 작성은 위에서 설명한 규칙에 따릅니다. 상점 서비스를 통해 다른 상점을 작성하는 템플릿으로 상점을 사용하려는 경우, 데이터 자원을 작성할 때 이 규칙을 따라야 합니다.

그러나 견본 상점 아카이브를 작성하지 않으려는 경우, 상점 서비스의 도구를 사용하여 이 규칙을 따르지 않는 상점 아카이브를 여전히 편집 또는 공개할 수 있습니다.



---

## 부록 C. sarinfo.xml

각 상점 아카이브는 sarinfo.xml 파일을 포함해야 합니다. 설명자로 알려진 파일에는 파일 자원 ZIP 파일 상점 데이터베이스 XML 파일의 이름 및 이 파일들이 공개된 순서를 포함하여, 상점 아카이브가 공개되었을 때 사용된 상점 아카이브에 대한 정보가 들어 있습니다. 상점 아카이브가 여러 언어로 된 파일을 포함하는 경우, sarinfo.xml 파일은 해당 정보도 포함하며, 각 언어 파일이 공개된 순서를 판별합니다.

주: 일부 데이터 자원이 다른 자원 이전에 공개되어야 하므로 데이터 자원이 공개된 순서는 중요합니다. 결과적으로, sarinfo.xml 파일에 지정된 대로 사용자 자원의 순서는 기본 상점에 대한 sarinfo.xml 파일에 지정된 자원의 순서와 일치해야 합니다. 기본 상점 아카이브는 다음 디렉토리에 있습니다.

- ▶ NT drive:\WebSphere\CommerceServer\samplstores
- ▶ 2000 drive:\Program Files\WebSphere\CommerceServer\samplstores
- ▶ AIX /usr/WebSphere/CommerceServer/samplstores
- ▶ Solaris /opt/WebSphere/CommerceServer/samplstores
- ▶ Linux /opt/WebSphere/CommerceServer/samplstores
- ▶ 400 /qibm/proddata/WebCommerce/samplstores

상점 아카이브의 콘텐츠를 보려면, ZIP 프로그램을 사용하여 압축을 푸십시오. sarinfo.xml은 SAR-INF 디렉토리에 있습니다.

---

### sarinfo.xml 예제

다음 예는 ToolTech sarinfo.xml 파일입니다. 요소, 속성 및 속성값에 대한 자세한 정보는 아래 정보를 참조하십시오. 상점 아카이브의 XML 스펙에 대한 자세한 정보는 다음 디렉토리에서 sarinfo.dtd를 참조하십시오.

- ▶ NT drive:\WebSphere\CommerceServer\xml\sar
- ▶ 2000 drive:\Program Files\WebSphere\CommerceServer\xml\sar
- ▶ AIX /usr/WebSphere/CommerceServer/xml/sar
- ▶ Solaris /opt/WebSphere/CommerceServer/xml/sar
- ▶ Linux /opt/WebSphere/CommerceServer/xml/sar
- ▶ 400 /qibm/proddata/WebCommerce/xml/sar

```

<?xml version = "1.0"?>
<!DOCTYPE sarinfo SYSTEM "sarinfo.dtd">
<sarinfo complete-store="yes" multi-language="yes" version="1.0">

<store-info asset-name="store"/>

<file name="webapp.zip" type="zip">

<asset fragmented="no" name="webapp">
<file name="webapp.zip" type="zip">
<display-name>My Web App Display Name</display-name>
<description>My Web App</description>
</file>
</asset>

<asset fragmented="no" name="properties">
<file name="properties.zip" type="zip" />
</asset>

<asset fragmented="no" name="dbloadmacros">
<file name="data/DBLoadMacros.dtd" type="dtd"/>
</asset>

<asset fragmented="no" name="fulfillment">
<file name="data/fulfillment.dtd" type="dtd"/>
<file name="data/fulfillment.xml" priority="1" type="db-load"/>
<file name="data/en_US/fulfillment.xml" priority="31" type="db-load">
<locale>en_US</locale>
</file>
<file name="data/es_ES/fulfillment.xml" priority="31" type="db-load">
<locale>es_ES</locale>
</file>
</asset>

<asset fragmented="yes" name="store">
<file name="data/store.dtd" type="dtd"/>
<file name="data/store.xml" priority="2" type="db-load"/>
<file name="data/en_US/store.xml" priority="3" type="db-load">
<locale>en_US</locale>
</file>
<file name="data/es_ES/store.xml" priority="3" type="db-load">
<locale>es_ES</locale>
</file>
</asset>

<asset fragmented="yes" name="catalog">
<file name="data/catalog.dtd" type="dtd"/>
<file name="data/catalog.xml" priority="4" type="db-load"/>
<file name="data/en_US/catalog.xml" priority="5" type="db-load">
<locale>en_US</locale>
</file>
<file name="data/es_ES/catalog.xml" priority="5" type="db-load">
<locale>es_ES</locale>
</asset>

<asset fragmented="yes" name="tax">
<file name="data/tax.dtd" type="dtd"/>

```



```

<file name="data/tax.xml" priority="6" type="db-load"/>
<file name="data/en_US/tax.xml" priority="7" type="db-load">
<locale>en_US</locale>
</file>
<file name="data/es_ES/tax.xml" priority="7" type="db-load">
<locale>es_ES</locale>
</file>
</asset>

<asset fragmented="no" name="taxfulfill">
<file name="data/taxfulfill.dtd" type="dtd"/>
<file name="data/taxfulfill.xml" priority="8" type="db-load"/>
</asset>

<asset fragmented="yes" name="shipping">
<file name="data/shipping.dtd" type="dtd"/>
<file name="data/shipping.xml" priority="9" type="db-load"/>
file name="data/en_US/shipping.xml" priority="10" type="db-load">
<locale>en_US</locale>
/file>
file name="data/es_ES/shipping.xml" priority="10" type="db-load">
<locale>es_ES</locale>
<file>
</asset>

<asset fragmented="no" name="shippingfulfill">
<file name="data/shipfulfill.dtd" type="dtd"/>
<file name="data/shipfulfill.xml" priority="11" type="db-load"/>
</asset>

<asset fragmented="no" name="store-catalog">
<file name="data/store-catalog.dtd" type="dtd"/>
<file name="data/store-catalog.xml" priority="12" type="db-load"/>
</asset>

<asset fragmented="no" name="storefulfill">
<file name="data/storefulfill.dtd" type="dtd"/>
<file name="data/storefulfill.xml" priority="13" type="db-load"/>
</asset>

<asset fragmented="yes" name="offering">
<file name="data/offering.dtd" type="dtd"/>
<file name="data/offering.xml" priority="14" type="db-load"/>
</asset>

<asset fragmented="no" name="command">
<file name="data/command.dtd" type="dtd"/>
<file name="data/command.xml" priority="16" type="db-load"/>
</asset>

<asset fragmented="yes" name="currency">
<file name="data/currency.dtd" type="dtd"/>
<file name="data/currency.xml" priority="17" type="db-load"/>
<file name="data/en_US/currency.xml" priority="18" type="db-load">
<locale>en_US</locale>
</file>
<file name="data/es_ES/currency.xml" priority="18" type="db-load">
<locale>es_ES</locale>

```

```

</file>
</asset>

<asset fragmented="yes" name="campaign">
<file name="data/campaign.dtd" type="dtd"/>
<file name="data/campaign.xml" priority="20" type="db-load"/>

<file name="data/en_US/campaign.xml" priority="24" type="db-load">
<locale>en_US</locale>
</file>
<file name="data/es_ES/campaign.xml" priority="24" type="db-load">
<locale>es_ES</locale>
</file>
</asset>

<asset fragmented="no" name="store-catalog-tax">
<file name="data/store-catalog-tax.dtd" type="dtd"/>
<file name="data/store-catalog-tax.xml" priority="21" type="db-load"/>
</asset>

<asset fragmented="no" name="store-catalog-shipping">
<file name="data/store-catalog-shipping.dtd" type="dtd"/>
<file name="data/store-catalog-shipping.xml" priority="22" type="db-load"/>
</asset>

<asset fragmented="no" name="store-defaults">
<file name="data/store-defaults.dtd" type="dtd"/>
<file name="data/store-defaults.xml" priority="23" type="db-load"/>
</asset>

<asset fragmented="no" name="consistency_check">
<file name="data/sarrule.dtd" type="dtd"/>
<file name="data/sarrule.xml" priority="25" type="config"/>
</asset>

<asset fragmented="no" name="payment">
<file name="data/es_ES/paymentinfo.xml" type="config"/>
<file name="data/paymentinfo.dtd" type="dtd"/>
</asset>

<asset fragmented="yes" name="policy">
<file name="data/businesspolicy.dtd" type="dtd"/>
<file name="data/businesspolicy.xml" priority="26" type="db-load"/>
<file name="data/en_US/businesspolicy.xml" priority="27" type="db-load">
<locale>en_US</locale>
</file>
<file name="data/es_ES/businesspolicy.xml" priority="27" type="db-load">
<locale>es_ES</locale>
</file>
</asset>

<asset fragmented="no" name="organization">
<file name="data/organization.dtd" type="dtd"/>
<file name="data/organization.xml" priority="28" type="db-load"/>
</asset>

asset fragmented="no" name="businessaccount">

```

```

<file name="data/businessaccount.xml" type="xml"/>
/asset>
asset fragmented="yes" name="contract">
<file name="data/contract.xml" priority="1" type="xml"/>
<file name="data/en_US/contract.xml" priority="2" type="xml">
<locale>en_US</locale>
</file>
file name="data/es_ES/contract.xml" priority="2" type="xml">
<locale>es_ES</locale>
</file>
</asset>

<asset fragmented="yes" name="accesscontrol">
<file name="data/accesscontrol.dtd" type="dtd"/>
<file name="data/accesscontrol.xml" priority="29" type="db-load"/>
<file name="data/en_US/accesscontrol.xml" priority="30" type="db-load">
<locale>en_US</locale>
</file>
<file name="data/es_ES/accesscontrol.xml" priority="30" type="db-load">
<locale>es_ES</locale>
</file>
</asset>

<!-- next priority should be 32 -->

</sarinfo>

```

여기서,

sarinfo는 sarinfo.xml 파일에 있는 모든 정보입니다. 다음 도표에 있는 속성에는 상점 아카이브에 대한 일반 정보가 들어 있습니다.

속성 이름	속성값
다중 언어(필수)	다중 언어가 이 상점 아카이브에서 지원될지 여부를 결정합니다. <ul style="list-style-type: none"> <li>예</li> <li>아니오</li> </ul>
완전한 상점(필수)	상점 아카이브가 완전한 상점에 필요한 자원을 포함하는지 여부를 결정합니다. <ul style="list-style-type: none"> <li>예</li> <li>아니오</li> </ul>
버전	상점 아카이브의 버전입니다(예: 1.0, 1.1)
표시 이름	상점 아카이브의 이름입니다.
설명	상점 아카이브에 대한 간단한 설명
표준 스키마(필수)	상점 아카이브가 표준 WebSphere Commerce 데이터베이스 스키마를 따르는지 여부를 결정합니다. <ul style="list-style-type: none"> <li>예</li> <li>아니오</li> </ul>
상점 정보 자원 이름(필수)	상점 아카이브에 대한 앵커로 기능하는 자원. 상점의 모든 정보는 이 자원에 속하는 파일에 있습니다(예: 상점).

속성 이름	속성값
로케일	<p>상점 아카이브가 지원하는 로케일. 로케일 변수(아래 나열)는 언어_국가로 구성됩니다.</p> <ul style="list-style-type: none"> <li>• de_DE</li> <li>• en_US</li> <li>• es_ES</li> <li>• fr_FR</li> <li>• it_IT</li> <li>• ja_JP</li> <li>• ko_KR</li> <li>• pt_BR</li> <li>• zh_CN</li> <li>• Zh_TW</li> </ul>

asset(필수) 자원은 관련된 파일의 논리 콜렉션입니다. 예를 들어, 세금은 상점 세금에 관련된 파일 그룹의 이름입니다.

속성 이름	속성값
이름(필수)	<p>자원 유형의 이름입니다. 예를 들면, 다음과 같습니다.</p> <ul style="list-style-type: none"> <li>• 상점</li> <li>• 카탈로그</li> <li>• 지불</li> <li>• 세금</li> </ul>
단편(필수)	<p>자원 정보가 언어에 따라 다중 파일로 분할되는지 여부를 결정합니다.</p> <ul style="list-style-type: none"> <li>• 예</li> <li>• 아니오</li> </ul>

## file

속성 이름	속성값
이름(필수)	파일의 이름입니다.
유형(필수)	<p>파일 포맷 유형. 예를 들면, 다음과 같습니다.</p> <ul style="list-style-type: none"> <li>• db-load - 데이터베이스에 로그할 파일</li> <li>• dtd - 문서 유형 정의 파일</li> <li>• zip - 파일 자원의 ZIP 파일(예: webapp.zip)</li> <li>• config - 구성 파일</li> </ul>

속성 이름	속성값
우선순위	상점 아카이브에 있는 파일이 공개될 순서. 1,2,3,4 ... 주: 두 개의 파일이 동일한 우선순위를 공유하는 경우, 로드 순서는 관계가 없습니다. 파일이 특정 순서로 로드되어야 하는 경우, 다른 우선순위가 지정되어야 합니다.
표시 이름	파일의 이름입니다.
설명	참조 설명
로케일	로케일입니다. 로케일 변수(아래 나열)는 언어_국가로 구성됩니다. <ul style="list-style-type: none"> <li>• de_DE</li> <li>• en_US</li> <li>• es_ES</li> <li>• fr_FR</li> <li>• it_IT</li> <li>• ja_JP</li> <li>• ko_KR</li> <li>• pt_BR</li> <li>• zh_CN</li> <li>• Zh_TW</li> </ul>



---

## 부록 D. sarrule.xml

개별 상점 아카이브에는 상점 서비스를 사용하여 공개할 때 일관성 확인 프로그램을 실행하는 sarrule.xml 파일이 있습니다. 공개 중, 공개 유틸리티는 sarrule.xml 파일의 규칙을 사용하여 XML 파일에 나열된 웹 자원이 상점 아카이브에 있는지 확인합니다.

---

### sarrule.xml 예제

다음 sarrule.xml 파일은 ToolTech 견본 상점에서 제공됩니다.

상점 아카이브 파일은 다음 디렉토리에 있습니다.

- ▶ NT drive:\WebSphere\CommerceServer\samplstores
- ▶ 2000 drive:\Program Files\WebSphere\CommerceServer\samplstores
- ▶ AIX /usr/WebSphere/CommerceServer/samplstores
- ▶ Solaris /opt/WebSphere/CommerceServer/samplstores
- ▶ Linux /opt/WebSphere/CommerceServer/samplstores
- ▶ 400 /qibm/proddata/WebCommerce/samplstores

상점 아카이브의 sarrule.xml 파일을 보려면, ZIP 프로그램을 사용하여 이 파일의 압축을 푸십시오. sarrule.xml 파일은 data 디렉토리에 있습니다.

sarrule.dtd 파일은 다음 디렉토리에 있습니다.

- ▶ NT drive:\WebSphere\CommerceServer\xml\sar
- ▶ 2000 drive:\Program Files\WebSphere\CommerceServer\xml\sar
- ▶ AIX /usr/WebSphere/CommerceServer/xml/sar
- ▶ Solaris /opt/WebSphere/CommerceServer/xml/sar
- ▶ Linux /opt/WebSphere/CommerceServer/xml/sar
- ▶ 400 /qibm/proddata/WebCommerce/xml/sar

주: 상점 아카이브에서 견본 상점과 함께 제공되는 sarrule.xml 파일을 사용하는 것이 좋습니다. 그러나 원하는 경우 새 규칙을 기존 sarrule.xml 파일에 추가할 수 있습니다.

```
<?xml version="1.0"?>
<!DOCTYPE SAR-rules SYSTEM "sarrule.dtd">
<SAR-rules>
```

```

<asset name = "command">
<check type="webasset registration">
<rule entry="viewreg" attribute="properties" type="java.lang.String"
 removeStoreDir="false" file="webapp.zip"/>
</check>
</asset>
!--
 <asset name = "catalog">
<check type="webasset registration">
 <rule entry = "catalogdsc" attribute="thumbnail" type="java.lang.String"
 removeStoreDir="false" file="webapp.zip"/>
<rule entry = "catalogdsc" attribute="fullimage" type="java.lang.String"
 removeStoreDir="false" file="webapp.zip"/>
<rule entry = "catentdesc" attribute="thumbnail" type="java.lang.String"
 removeStoreDir="false" file="webapp.zip"/>
<rule entry = "catentdesc" attribute="fullimage" type="java.lang.String"
 removeStoreDir="false" file="webapp.zip"/>
<rule entry = "catgrpdesc" attribute="thumbnail" type="java.lang.String"
 removeStoreDir="false" file="webapp.zip"/>
<rule entry = "catgrpdesc" attribute="fullimage" type="java.lang.String"
 removeStoreDir="false" file="webapp.zip"/>
</check>
</asset>
-->
 <asset name = "store">
<check type="webasset registration">
 <rule entry = "dispentrel" attribute="pagename" type="java.lang.String"
 removeStoreDir="false" file="webapp.zip"/>
 <rule entry = "dispcgprel" attribute="pagename" type="java.lang.String"
 removeStoreDir="false" file="webapp.zip"/>
</check>
</asset>
</SAR-rules>

```

여기서,

```

<asset name = "command">
<check type="webasset registration">
<rule entry="viewreg" attribute="properties" type="java.lang.String"
 removeStoreDir="false" file="webapp.zip"/>
</check>
</asset>

```

상점 서비스의 공개 유틸리티는 command.xml 파일에 나열된 각 JSP 파일이 상점 아카이브의 웹 자원에 있는지 확인합니다.

여기서,

```

 <asset name = "catalog">
<check type="webasset registration">
 <rule entry = "catalogdsc" attribute="thumbnail" type="java.lang.String"
 removeStoreDir="false" file="webapp.zip"/>
<rule entry = "catalogdsc" attribute="fullimage" type="java.lang.String"
 removeStoreDir="false" file="webapp.zip"/>
<rule entry = "catentdesc" attribute="thumbnail" type="java.lang.String"
 removeStoreDir="false" file="webapp.zip"/>
<rule entry = "catentdesc" attribute="fullimage" type="java.lang.String"
 removeStoreDir="false" file="webapp.zip"/>
<rule entry = "catgrpdesc" attribute="thumbnail" type="java.lang.String"
 removeStoreDir="false" file="webapp.zip"/>

```



```
<rule entry = "catgrpdesc" attribute="fullimage" type="java.lang.String"
 removeStoreDir="false" file="webapp.zip"/>
</check>
</asset>
```

상점 서비스의 공개 유틸리티는 catalog.xml 파일에 나열된 각 카탈로그 자원이 상점 서비스의 웹 자원에 있는지 확인합니다.

여기서,

```
<asset name = "store">
<check type="webasset registration">
 <rule entry = "dispentrel" attribute="pagename" type="java.lang.String"
 removeStoreDir="false" file="webapp.zip"/>
 <rule entry = "dispcgprel" attribute="pagename" type="java.lang.String"
 removeStoreDir="false" file="webapp.zip"/>
</check>
</asset>
```

상점 서비스의 공개 유틸리티는 store.xml 파일에 나열된 각 상점 자원이 상점 아카이브의 웹 자원에 있는지 확인합니다.



공개 중 메모리가 문제가 되는 경우, 공개하기 전에 sarrule.xml에서 카탈로그 자원 규칙을 주석화하십시오.

---



---

## 부록 E. 데이터베이스 자원 그룹

모든 WebSphere Commerce 데이터베이스 자원은 작성 및 로드할 위해 그룹으로 분리됩니다. 이 그룹은 논리적으로 관련된 테이블 세트입니다. 특정 오브젝트가 오브젝트 사이의 관계를 로드하기 전에 존재해야하기 때문에 이 데이터베이스 자원 그룹이 구성된 순서는 데이터 로드시 중요합니다.

상점에 대해 XML 포맷에서 데이터베이스를 로드할 때, 선택된 그룹만 로드 선택할 수 있습니다. 이들 그룹은 이전 장에서 작성된 카탈로그 또는 이행과 같은 데이터베이스 자원으로 구성됩니다. 316 페이지의 『데이터베이스 자원 그룹 로드』에 표시된 것처럼 데이터 그룹을 로드하기 전에, 다음을 수행하십시오.

- 로드 중인 데이터베이스 자원 그룹을 판별하십시오. 각각의 그룹은 자원이 로드되기 전에 충족되어야 하는 종속성을 포함합니다. 『데이터베이스 자원 그룹 종속성』 정보를 검토하십시오.
- 선택된 데이터베이스 자원 그룹에 대한 XML 파일을 작성 또는 갱신하십시오. 각 자원 장의 정보는 데이터베이스 자원 작성 순서를 권고하지만, XML 파일 작성 또는 갱신시 상위 테이블에 대한 정보가 하위 테이블에 대한 정보를 우선해야 함을 명시하십시오.

---

### 데이터베이스 자원 그룹 종속성

각 데이터베이스 자원 그룹이 WebSphere Commerce 데이터베이스 테이블에서 해당 정보를 가져옵니다. 데이터베이스 자원은 이들 소유 그룹 내에서 종속성을 가집니다. 즉, 데이터베이스 자원 그룹은 다른 데이터 그룹에서 다른 XML 파일에 데이터를 가져올 수 없으며, 각 그룹은 foreign key를 제외하고는 종속적입니다. 그러나 데이터베이스 자원 그룹이 다른 그룹에서 정의된 외부 데이터를 참조할 필요가 있는 경우, 수동으로 데이터를 제공해야 합니다. 이것은 한 개 그룹에서의 데이터는 도메인 밖(즉, 다른 데이터베이스 자원 그룹)에서 정의된 데이터에 대해 외부 종속성을 가진다는 것을 의미합니다. 외부 종속성은 데이터베이스 자원 그룹이 다른 그룹 테이블의 1차 키에 대한 foreign key 관계를 가질 때 발생합니다. 데이터베이스 자원 그룹을 로드하려면, 외부 종속성을 충족시켜야 합니다. 아래 도표의 예를 사용하려면, 상점 데이터베이스 자원 그룹에 대한 외부 종속성 중 하나가 fulfillment.FFMCENTER.FFMCENTER\_ID이며, 이것은 이행 데이터베이스 자원이 상점 데이터베이스 자원 그룹을 로드하기 전에 WebSphere Commerce 데이터베이스에 이미 존재해야하는 것을 의미합니다.


로드 프로세스를 시작하기 전에 다음 도표를 고려하십시오. 데이터베이스 자원의 각 그룹은 데이터가 로드되는 다른 데이터베이스 테이블에 대해 종속적입니다.

기억해야 할 점:

- 일부 외부 종속성은 단일 그룹에 의해 충족되지 않을 수 있습니다. 모든 상점이 사용하는 사이트 와이드 또는 일반 데이터베이스 자원은 부트스트랩에서의 인스턴스 작성시 미리 대량 자료 반입되며 쉽게 액세스될 수 있습니다. 데이터베이스 자원 그룹에 포함된 테이블은 이 데이터의 유형을 참조하는 foreign key를 가집니다. 부트스트랩 데이터가 일반 및 로케일 지정 데이터로 구분됩니다. 다국어 지원 상점을 가지고 있는 경우, 일반 및 로케일 지정 부트스트랩 데이터를 선택할 필요가 있습니다. 예를 들어, 언어 및 구성원 부트스트랩 데이터가 필요합니다. 인스턴스 작성 프로세스는 상점에 대해 지원되는 WebSphere Commerce 언어로 LANGUAGE 테이블을 대량 자료 반입하며, 루트 조직(MEMBER.MEMBER\_ID=-2001) 및 기본 조직(MEMBER.MEMBER\_ID=-2000)을 작성합니다. 필요한 곳에 루트 조직을 사용해야 하나 기본 조직을 사용하는 대신 상점 소유자 조직을 작성해야 합니다. 조직 및 계층에 대한 자세한 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오.
- 외부 종속성 열 아래에 나열된 파일은 다음 이름 지정 구조를 사용합니다(예: 데이터베이스 자원 그룹.데이터베이스 테이블.데이터베이스 열). 예를 들어, store.STOREENT.STOREENT\_ID 파일을 사용할 경우, 데이터는 store 데이터베이스 자원 그룹, STOREENT 테이블 및 STOREENT\_ID 열에서 가져 옵니다. 부트스트랩으로 시작하는 파일 이름은 WebSphere Commerce 인스턴스 작성 동안 데이터가 생성되었음을 나타냅니다.
- 외부 종속성 열 아래 나열된 파일이 관련된 테이블에서 foreign key 참조를 포함합니다. 이 테이블이 먼저 대량 자료 반입되어야 합니다.
- 표시 전용 목적으로 테이블이 상품 설명과 같이 다국어 지원 정보가 있는 로케일 지정 테이블을 나타내기 위해 나누어 집니다.
- 표의 테이블은 WebSphere Commerce 견본 상점에서 데이터베이스 자원을 표시합니다. 테이블은 상점 크기, 특성 및 필요에 따라 다양합니다. 사용자의 상점 요구사항에 따라 특정 테이블 아래에 나열되지 않았을 지라도 상점 자원을 갖는 모든 데이터베이스 테이블을 포함하는지 확인하십시오.

액세스 제어 데이터베이스 자원		
외부 종속성	데이터베이스 자원 XML 파일에서 관련된 테이블	데이터베이스 자원 XML 파일에서 관련된 로케일 스펙 테이블
bootstrap.LANGUAGE.LANGUAGE_ID(루트 및 상점 소유자 조직), bootstrap.MEMBER.MEMBER_ID(루트 및 상점 소유자 조직)	accesscontrol.xml ACACTACTGP, ACACTGRP, ACACTION, ACPOLICY, ACRESCGRY, ACRESGPRES, ACRESGRP	accesscontrol.xml ACACGPDESC, ACACTDESC, ACPOLDESC, ACRSCGDES, ACRESGPDES
비즈니스 정책 데이터베이스 자원		

외부 종속성	데이터베이스 자원 <b>XML</b> 파일에서 관련된 테이블	데이터베이스 자원 <b>XML</b> 파 일에서 관련된 로케일 스펙 테이블
bootstrap.LANGUAGE.LANGUAGE_ID, bootstrap.MEMBER.MEMBER_ID, store. STOREENT.STOREENT_ID(상점 소유자 조직)	<b>businesspolicy.xml</b> POLICY, POLICYCMD	<b>businesspolicy.xml</b> POLICYDESC
캠페인 데이터베이스 자원		
외부 종속성	데이터베이스 자원 <b>XML</b> 파일에서 관련된 테이블	데이터베이스 자원 <b>XML</b> 파 일에서 관련된 로케일 스펙 테이블
store.STOREENT.STOREENT_ID	<b>campaign.xml</b> CAMPAIGN, COLLATERAL, EMSPOT, STENCALUSG	<b>campaign.xml</b> COLLDESC
카탈로그 데이터베이스 자원		
외부 종속성	데이터베이스 자원 <b>XML</b> 파일에서 관련된 테이블	데이터베이스 자원 <b>XML</b> 파 일에서 관련된 로케일 스펙 테이블

bootstrap.LANGUAGE.LANGUAGE_ID, bootstrap.MEMBER.MEMBER_ID(상점 소유자 조 직), store.STOREENT.STOREENT_ID, shipping.CALCODE.CALCODE_ID, tax.CALCODE.CALCODE_ID	<b>catalog.xml</b> BASEITEM, CATALOG, CATENTREL, CATENTRY, CATGROUP, CATGRPREL, CATTOGRP, ITEMSPC, ITEMVERSN, RA, RADETAIL, STOREITEM, STORITMFFC, VERSIONSPC <b>offering.xml</b> CATGRPTPC, MGPTRDPSCN, OFFER, OFFERPRICE, TRADEPOSCN <b>storefulfill.xml</b> INVENTORY <b>store-catalog.xml</b> DISPCGPREL, DISPENTREL, STORECAT, STORECENT, STORECGRP <b>store-catalog- shipping.xml</b> CATENTCALCD, CATENTSHIP <b>store-catalog-tax. xml</b> CATENTCALD	<b>catalog.xml</b> ATTRIBUTE, ATTRVALUE, BASEITMDSC, CATALOGDSC, CATENTDESC, CATGRPDESC, PKGATTR, PKGATTRVAL,
명령 데이터베이스 자원		
외부 종속성	데이터베이스 자원 <b>XML</b> 파일에서 관련된 테이블	데이터베이스 자원 <b>XML</b> 파 일에서 관련된 로케일 스펙 테이블
store.STOREENT.STOREENT_ID	<b>command.xml</b> CMDREG, VIEWREG	N/A
 <b>Business</b> 장기 구매 계약 데이터베이스 자원		
외부 종속성	데이터베이스 자원 <b>XML</b> 파일에서 관 련된 테이블	

store.STOREENT.STOREENT_ID		장기 구매 계약 데이터베이스 테이블은 직접 로드되지 않고 WebSphere Commerce 데이터 그룹이 아닌 다른 프로세스를 따릅니다. 자세한 정보는 325 페이지의 『장기 구매 계약 자원 공개』를 참조하십시오.
통화 데이터베이스 자원		
외부 종속성	데이터베이스 자원 XML 파일에서 관련된 테이블	데이터베이스 자원 XML 파일에서 관련된 로케일 스펙 테이블
store.STOREENT.STOREENT_ID	currency.xml CURCVLIST	currency.xml CURCONVERT, CURLIST
이행 데이터베이스 자원		
외부 종속성	데이터베이스 자원 XML 파일에서 관련된 테이블	데이터베이스 자원 XML 파일에서 관련된 로케일 스펙 테이블
bootstrap.LANGUAGE.LANGUAGE_ID, bootstrap.MEMBER.MEMBER_ID(상점 소유자 조직)	fulfillment.xml FFM CENTER, STADDRESS	fulfillment.xml FFMCENTDS
조직 데이터베이스 자원		
외부 종속성	데이터베이스 자원 XML 파일에서 관련된 테이블	데이터베이스 자원 XML 파일에서 관련된 로케일 스펙 테이블
bootstrap.LANGUAGE.LANGUAGE_ID(루트 및 상점 소유자 조직), bootstrap.MEMBER.MEMBER_ID(루트 및 상점 소유자 조직)	organization.xml ADDRESSBOOK, ADDRESS, MBRREL, MEMBER, ORGENTITY	N/A
운송 데이터베이스 자원		
외부 종속성	데이터베이스 자원 XML 파일에서 관련된 테이블	데이터베이스 자원 XML 파일에서 관련된 로케일 스펙 테이블
bootstrap.LANGUAGE.LANGUAGE_ID, bootstrap.MEMBER.MEMBER_ID(상점 소유자 조직), fulfillment.FFMCENTER.FFMCENTER_ID, store.STOREENT.STOREENT_ID	shipping.xml CALCODE, CALRULE, CRULESCALE, JURST, JURSTGPREL, JURSTGROUP, SHIPMODE, STENCALUSG shipping.xml SHPJCRULE, SHPARRANGE	shipping.xml CALCODEDSC, CALRANGE, CALRLOOKUP, CALSCALE, SHPMODEDSC
상점 데이터베이스 자원		
외부 종속성	데이터베이스 자원 XML 파일에서 관련된 테이블	데이터베이스 자원 XML 파일에서 관련된 로케일 스펙 테이블

bootstrap.LANGUAGE.LANGUAGE_ID, bootstrap.MEMBER.MEMBER_ID(상점 소유자 조직), bootstrap.SETCURR.SETCURR_ID, fulfillment.FFMCENTER.FFMCENTER_ID	<b>store.xml</b> INVADJCODE, RTNREASON, STORE, STORENT, STORELANG, VENDOR	<b>store.xml</b> FFMCENTDS, INVADJDESC, RTNRSNDESC, STADDRESS, STOREENTDS, STORLANGDS, VENDORDESC
상점 기본값 데이터베이스 자원		
외부 종속성	데이터베이스 자원 <b>XML</b> 파일에서 관련된 테이블	데이터베이스 자원 <b>XML</b> 파일에서 관련된 로케일 스펙 테이블
shipping.SHIPMODE.SHIPMODE_ID(적용 가능한 경우, 이 파일이 먼저 로드되어야 함), contract.CONTRACT.CONTRACT_ID, store.STOREENT.STOREENT_ID	<b>store-default.xml</b> STOREDEF	N/A
세금 데이터베이스 자원		
외부 종속성	데이터베이스 자원 <b>XML</b> 파일에서 관련된 테이블	데이터베이스 자원 <b>XML</b> 파일에서 관련된 로케일 스펙 테이블
bootstrap.LANGUAGE.LANGUAGE_ID, bootstrap.MEMBER.MEMBER_ID(상점 소유자 조직), store.STOREENT.STOREENT_ID, fulfillment.FFMCENTER.FFMCENTER_ID, store.STOREENT.STOREENT_ID	<b>tax.xml</b> CALCODE, CALRANGE, CALRLOOKUP, CALRULE, CALSCALE, CRULESCALE, JURST, JURSTGROUP, JURSTGPREL, STENCALUSG, TAXCGRY, TAXJCRULE <b>taxfulfill.xml</b> TAXJCRULE	<b>tax.xml</b> CALCODEDSC, CALSCALEDSC, TAXCGRYDS



---

## 부록 F. 주의사항

이 정보는 미국에서 제공되는 제품 및 서비스용으로 작성된 것입니다. IBM은 다른 국가에서 이 책에 기술된 제품, 서비스 또는 기능을 제공하지 않을 수도 있습니다. 현재 사용할 수 있는 제품 및 서비스에 대한 정보는 한국IBM 담당자에게 문의하십시오. 이 책에서 IBM 제품, 프로그램 또는 서비스를 언급했다고 해서 해당 IBM 제품, 프로그램 또는 서비스만을 사용할 수 있다는 것을 의미하지는 않습니다. IBM의 지적 재산을 침해하지 않는 한, 기능상으로 동등한 제품, 프로그램 또는 서비스를 대신 사용할 수도 있습니다. 그러나 비IBM 제품, 프로그램 또는 서비스의 운영에 대한 평가 및 검증은 사용자의 책임입니다.

IBM은 이 책에서 다루고 있는 특정 내용에 대해 특허를 보유하고 있거나 특허 출원 중일 수 있습니다. 이 책을 제공한다고 해서 특허에 대한 사용권까지 부여하는 것은 아닙니다. 사용권에 대한 의문사항은 다음으로 문의하십시오.

137-270

서울특별시 강남구 도곡동 467-12, 군인공제회관빌딩

한국 아이.비.엠 주식회사

고객만족센터

전화번호: 080-023-8080

2바이트(DBCS) 정보에 관한 사용권 문의는 한국 IBM 고객만족센터에 문의하거나 다음 주소로 서면 문의하시기 바랍니다.

IBM World Trade Asia Corporation

Licensing

2-31 Roppongi 3-chome, Minato-ku

Tokyo 106, Japan

다음 단락은 현지법과 상충하는 영국이나 기타 국가에서는 적용되지 않습니다.

IBM은 타인의 권리 비침해, 제품성 및 특정 목적에의 적합성에 대한 묵시적 보증을 포함하여(단, 이에 한하지 않음) 묵시적이든 명시적이든 어떠한 종류의 보증없이 이 책을 현상태대로 제공합니다. 일부 국가에서는 특정 거래에서 명시적 또는 묵시적 보증의 면책사항을 허용하지 않으므로, 이 사항이 적용되지 않을 수도 있습니다.

이 정보에는 기술적으로 부정확한 내용이나 인쇄상의 오류가 있을 수 있습니다. 이 정보는 주기적으로 변경되며, 이 변경사항은 최신판에 통합됩니다. IBM은 이 책에서 설명한 제품 및/또는 프로그램을 사전 통고없이 언제든지 개선 및/또는 변경할 수 있습니다.

이 정보에서 비IBM의 웹 사이트는 단지 편의상 제공된 것으로, 어떤 방식으로든 이들 웹 사이트를 옹호하고자 하는 것은 아닙니다. 해당 웹 사이트의 자료는 본 IBM 제품 자료의 일부가 아니므로 해당 웹 사이트 사용으로 인한 위험은 사용자 본인이 감수해야 합니다. 해당 웹 사이트의 자료는 본 IBM 제품 자료의 일부가 아니므로 해당 웹 사이트의 사용으로 인한 위험은 사용자 본인이 감수해야 합니다.

IBM은 귀하의 권리를 침해하지 않는 범위 내에서 적절하다고 생각하는 방식으로 귀하가 제공한 정보를 사용하거나 배포할 수 있습니다.

- (1) 독립적으로 작성된 프로그램 및 기타 프로그램(이 프로그램 포함) 간의 정보 교환
- (2) 교환된 정보의 상호 이용을 목적으로 정보를 원하는 프로그램 사용권자는 다음 주소로 문의하십시오.

135-270

서울특별시 강남구 도곡동 467-12, 군인공제회관빌딩

한국 아이.비.엠 주식회사

고객만족센터

이러한 정보는 해당 규정에 따라(예를 들어, 사용권 지불 포함) 사용할 수 있습니다.

이 정보에 기술된 사용권 프로그램 및 사용 가능한 모든 사용권 자료는 IBM이 IBM 기본 계약, IBM 프로그램 사용권 계약(IPLA) 또는 이와 동등한 계약에 따라 제공한 것입니다.

이 책에 포함된 성능 데이터는 제한된 환경에서 산출된 것입니다. 그러므로 다른 운영 환경에서의 결과와 상당히 다를 수도 있습니다. 일부 측정은 개발 단계 시스템에서 이루어진 것이므로 그 측정치가 일반적으로 사용 가능한 시스템에서도 동일하다고 보장할 수 없습니다. 또한, 일부 측정치는 보외법을 통해 이루어졌으므로 실제 결과는 다를 수 있습니다. 이 문서의 사용자는 자신의 특정 환경에 적용할 수 있는 데이터를 확인해야 합니다.

비IBM 제품에 관한 정보는 해당 제품의 공급업체, 공개 자료, 또는 기타 범용 소스로부터 얻은 것입니다. IBM에서는 이러한 제품들을 테스트하지 않았으므로 비IBM 제품의 성능 대한 정확성, 호환성 또는 배상 청구에 대해서는 확신할 수 없습니다. 비IBM 제품의 성능에 대한 의문사항은 해당 제품의 공급업체에 문의하십시오.

IBM의 향후 방향 또는 의도에 관한 모든 문장은 통고 없이 변경되거나 취소될 수 있으며, 목적과 목표만을 표현합니다.

IBM이 제안하는 소비자 가격으로 표시된 모든 IBM 가격은 현재 가격이며 통지 없이 변경될 수 있습니다. 소매업체 가격들은 다양할 수 있습니다.

이 정보는 계획 목적만을 위한 것입니다. 여기에 있는 정보는 설명된 제품이 사용 가능하게 되기 전에 변경될 수 있습니다.

이 정보에는 일상적인 비즈니스 작업에서 사용되는 데이터와 보고서의 예가 들어 있습니다. 가능한 한 완벽하게 표시하기 위해 예에는 개인, 회사, 브랜드, 제품의 이름이 포함됩니다. 이러한 모든 이름은 가상의 이름이며, 실제 비즈니스 엔터프라이즈에 의해 사용되는 이름 및 주소에 대한 유사성은 전적으로 우연입니다.

## 저작권

이 책에서는 소스 언어로 된 견본 응용프로그램이 들어 있어, 여러 운영 체제에서의 프로그래밍 기술에 대해 설명합니다. 견본 프로그램이 작성된 운영 체제에 해당하는 응용 프로그램 프로그래밍 인터페이스를 따르는 응용프로그램을 개발, 사용, 마케팅, 배포하려는 목적으로 해당 견본 프로그램을 무료로 복사, 수정, 배포할 수 있습니다. 이 책의 예들은 모든 조건에서 완전하게 테스트된 것은 아닙니다. 따라서 IBM은 이러한 프로그램들의 신뢰성, 서비스 가능성 또는 기능을 보장하거나 내포할 수 없습니다. IBM의 응용프로그램 프로그래밍 인터페이스를 따르는 응용프로그램을 개발, 사용, 마케팅, 배포하려는 목적으로 해당 견본 프로그램을 무료로 복사, 수정, 배포할 수 있습니다.

이러한 견본 프로그램의 각 사본이나 일부 또는 여기에서 파생된 저작물에는 다음과 같이 저작권 주의사항이 포함되어야 합니다.

©Copyright International Business Machines Corporation 2001. 코드의 일부는 IBM Corp. Sample Programs로부터 파생된 것입니다. ©Copyright IBM Corp. 2000, 2001. All rights reserved.

이 정보를 소프트카피로 보는 경우, 사진과 컬러 그림은 나타나지 않을 수 있습니다.

이 제품에서 제공되는 신용 카드 이미지, 상표 및 거래 이름은 신용 카드 표시 소유자가 권한 부여한 판매자만이 해당 신용 카드를 통한 지불을 승인하기 위해서만 사용해야 합니다.

---

## 상표

다음 용어는 미국 또는 기타 국가에서 사용되는 IBM Corporation의 상표 또는 등록 상표입니다.

AIX	IBM	WebSphere
AS/400	IBM Payment Manager	
DB2	iSeries	
DB2 Universal Database™	OS/400®	
eServer	VisualAge	

Microsoft®, Windows, Windows NT, Active Directory 및 Windows 로고는 미국 및 기타 국가에서 사용되는 Microsoft Corporation의 상표 또는 등록상표입니다.

Oracle은 Oracle Corporation의 등록상표이고 Oracle8은 Oracle Corporation의 상표입니다.

SET Secure Electronic Transaction™, SET™ 및 SET 로고는 SET Secure Electronic Transaction LLC가 보유한 상표입니다. SET Secure Electronic Transaction LLC의 서면 라이선스 없이 등록상표를 사용하는 것은 엄격히 금지되어 있습니다.

Solaris, Solaris Operating Environment, Java, JavaServer Pages, JavaBeans, 및 모든 Java 기반 상표와 로고는 Sun Microsystems, Inc.의 상표 또는 등록상표입니다.

UNIX®는 미국 및 기타 국가에서 사용되는 Open Group의 등록상표입니다.

기타 회사, 제품 및 서비스 이름은 다른 회사의 상표 또는 서비스표일 수 있습니다.







**IBM**